

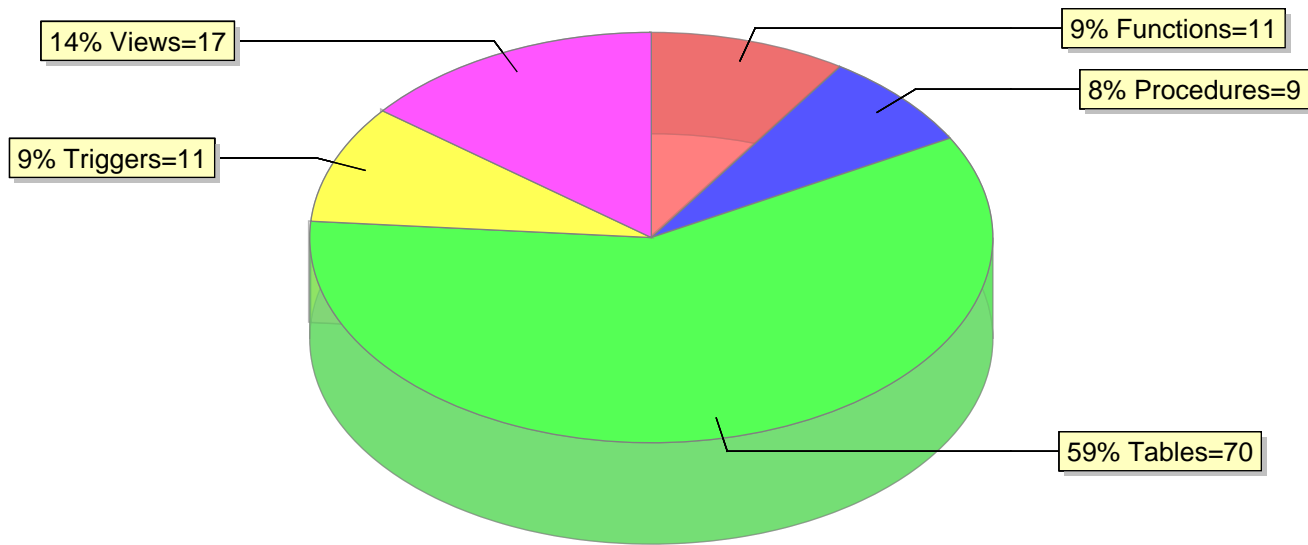
## Parameters

[Dashboards](#)

Seq	Parameter Name	Parameter Value
0	Time	Dec 24 2008 2:35:36:560PM
10	Database	AdventureWorks
20	Procedure Filter	%
30	Function Filter	%
40	Table Filter	%
50	View Filter	%
60	Trigger Filter	%
70	Index Filter	%

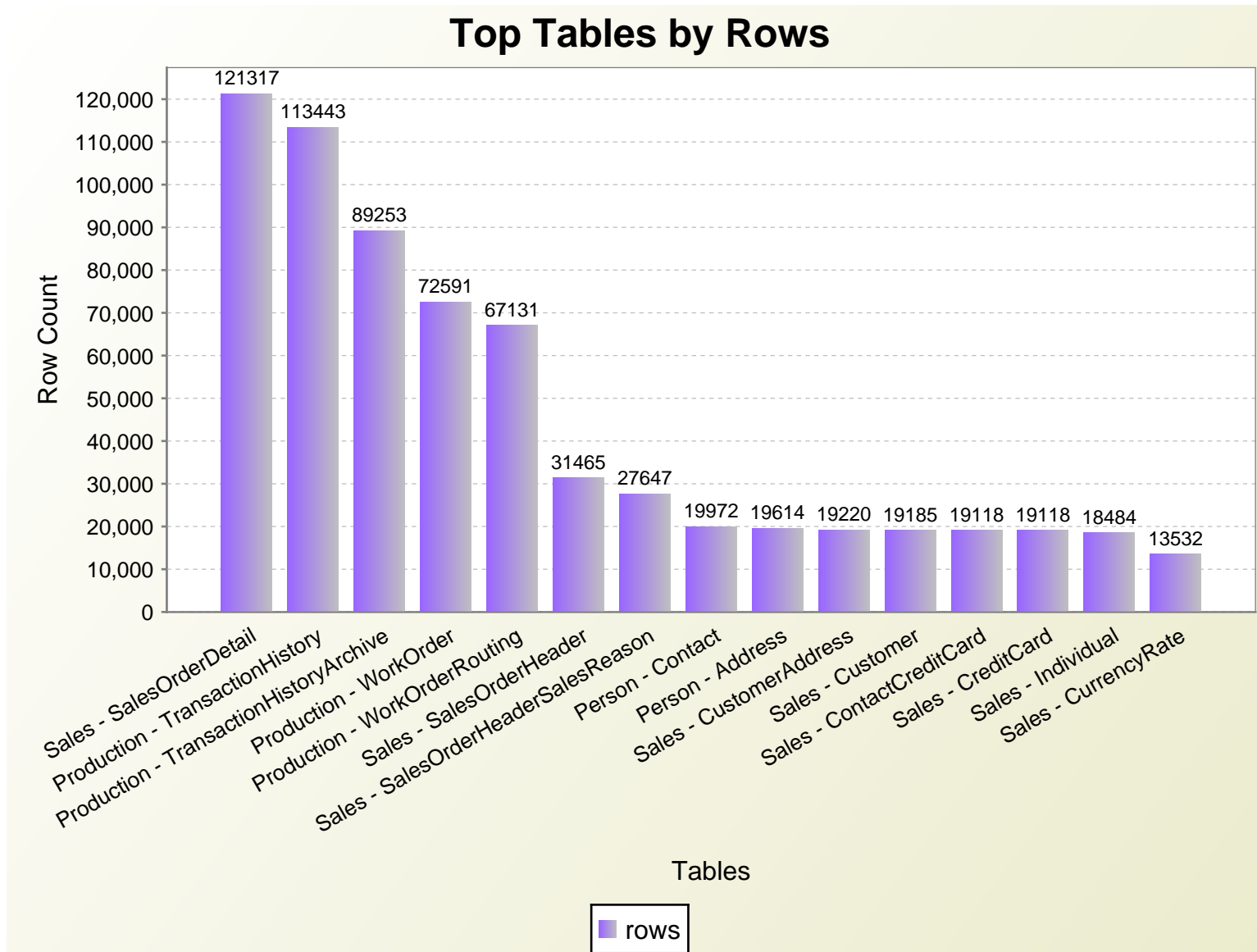
# Object Distribution

## Object Distribution

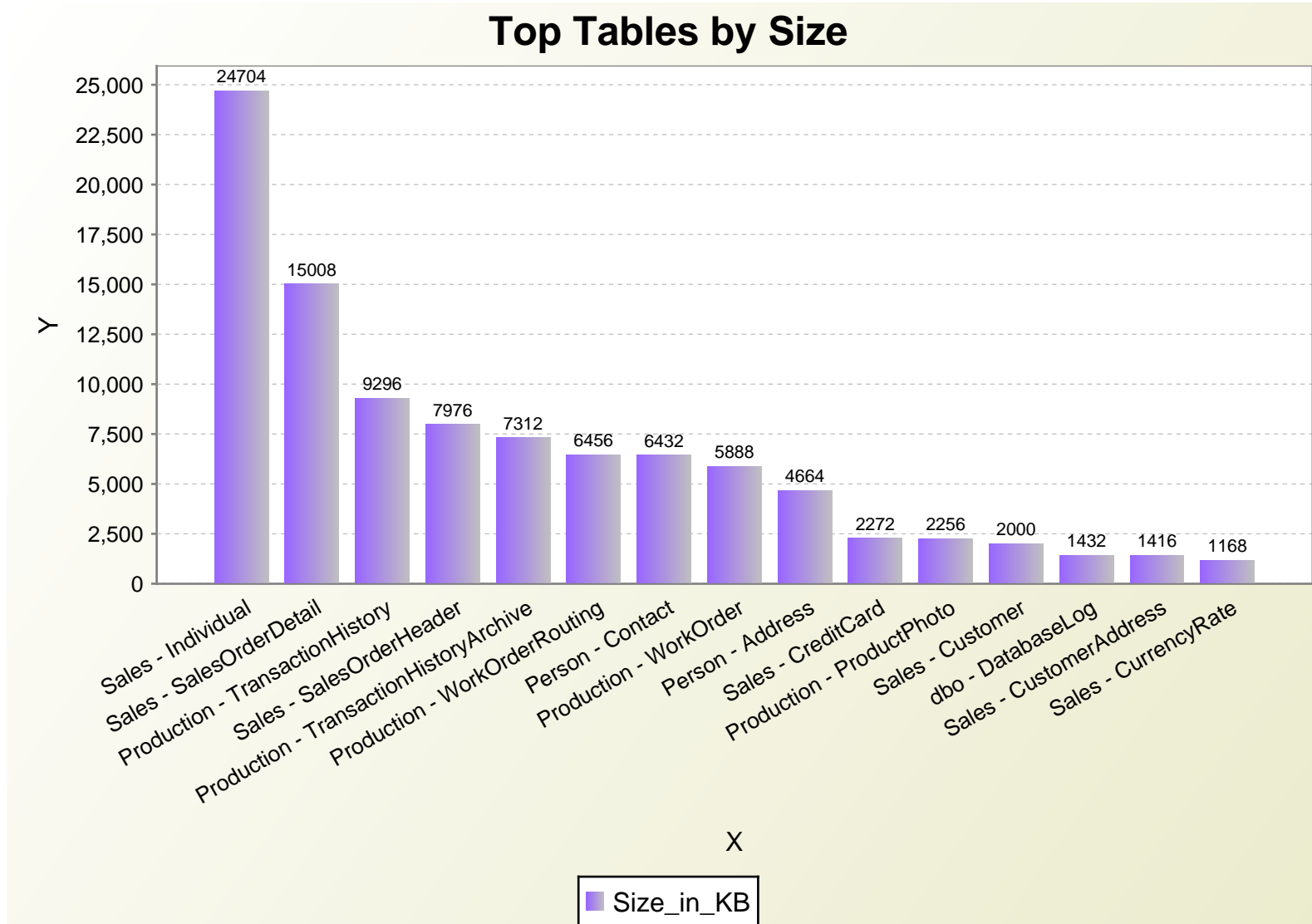


● Functions ● Procedures ● Tables ● Triggers ● Views

# Top Tables by Rows



# Top Tables by Size



## Tables

Table Name	Table Schema
AWBuildVersion	dbo
DatabaseLog	dbo
ErrorLog	dbo
Department	HumanResources
Employee	HumanResources
EmployeeAddress	HumanResources
EmployeeDepartmentHistory	HumanResources
EmployeePayHistory	HumanResources
JobCandidate	HumanResources
Shift	HumanResources
Address	Person
AddressType	Person
Contact	Person
ContactType	Person
CountryRegion	Person
StateProvince	Person
BillOfMaterials	Production
Culture	Production
Document	Production
Illustration	Production
Location	Production
Product	Production
ProductCategory	Production
ProductCostHistory	Production

Table Name	Table Schema
ProductDescription	Production
ProductDocument	Production
ProductInventory	Production
ProductListPriceHistory	Production
ProductModel	Production
ProductModelIllustration	Production
ProductModelProductDescriptionCulture	Production
ProductPhoto	Production
ProductProductPhoto	Production
ProductReview	Production
ProductSubcategory	Production
ScrapReason	Production
TransactionHistory	Production
TransactionHistoryArchive	Production
UnitMeasure	Production
WorkOrder	Production
WorkOrderRouting	Production
ProductVendor	Purchasing
PurchaseOrderDetail	Purchasing
PurchaseOrderHeader	Purchasing
ShipMethod	Purchasing
Vendor	Purchasing
VendorAddress	Purchasing
VendorContact	Purchasing
ContactCreditCard	Sales

Table Name	Table Schema
CountryRegionCurrency	Sales
CreditCard	Sales
Currency	Sales
CurrencyRate	Sales
Customer	Sales
CustomerAddress	Sales
Individual	Sales
SalesOrderDetail	Sales
SalesOrderHeader	Sales
SalesOrderHeaderSalesReason	Sales
SalesPerson	Sales
SalesPersonQuotaHistory	Sales
SalesReason	Sales
SalesTaxRate	Sales
SalesTerritory	Sales
SalesTerritoryHistory	Sales
ShoppingCartItem	Sales
SpecialOffer	Sales
SpecialOfferProduct	Sales
Store	Sales
StoreContact	Sales

## AWBuildVersion

### Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
Database Version	nvarchar	25		NO	Version number of the database in 9.yy.mm.dd.00 format.
ModifiedDate	datetime		(getdate())	NO	Date and time the record was last updated.
SystemInformationID	tinyint			NO	Primary key for AWBuildVersion records.
SystemInformationID	tinyint			NO	Clustered index created by a primary key constraint.
VersionDate	datetime			NO	Date and time the record was last updated.

### Table Indexes

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
AWBuildVersion	PK_AWBuildVersion_SystemInformationID	SystemInformationID	CLUSTERED	true	true	false	false	false	false

### Dependent Objects

---None---

## DatabaseLog

### Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
DatabaseLogID	int			NO	Primary key for DatabaseLog records.



Column Name	Data Type	Length	Column Default	Is Nullable	Description
DatabaseUser	nvarchar	128		NO	The user who implemented the DDL change.
Event	nvarchar	128		NO	The type of DDL statement that was executed.
Object	nvarchar	128		YES	The object that was changed by the DDL statement.
PostTime	datetime			NO	The date and time the DDL change occurred.
PostTime	datetime			NO	Nonclustered index created by a primary key constraint.
Schema	nvarchar	128		YES	The schema to which the changed object belongs.
TSQL	nvarchar	-1		NO	The exact Transact-SQL statement that was executed.
XmlEvent	xml	-1		NO	The raw XML data generated by database trigger.

#### Table Indexes

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
DatabaseLog	PK_DatabaseLog_DatabaseLogID	DatabaseLogID	NONCLUSTERED	true	true	false	false	false	false

#### Dependent Objects

---None---

## ErrorLog

### Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
ErrorLine	int			YES	The line number at which the error occurred.
ErrorLogID	int			NO	Primary key for ErrorLog records.
ErrorLogID	int			NO	Clustered index created by a primary key constraint.
ErrorMessage	nvarchar	4000		NO	The message text of the error that occurred.
ErrorNumber	int			NO	The error number of the error that occurred.
ErrorProcedure	nvarchar	126		YES	The name of the stored procedure or trigger where the error occurred.
ErrorSeverity	int			YES	The severity of the error that occurred.
ErrorState	int			YES	The state number of the error that occurred.
ErrorTime	datetime		(getdate())	NO	The date and time at which the error occurred.
UserName	nvarchar	128		NO	The user who executed the batch in which the error occurred.

### Table Indexes

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
ErrorLog	PK_ErrorLog_ErrorLogID	ErrorLogID	CLUSTERED	true	true	false	false	false	false

### Dependent Objects

Dependent Obj	Dependent Obj Type
uspLogError	SQL_STORED_PROCEDURE

## Department

### Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
DepartmentID	smallint			NO	Primary key for Department records.
DepartmentID	smallint			NO	Clustered index created by a primary key constraint.
GroupName	nvarchar	50		NO	Name of the group to which the department belongs.
ModifiedDate	datetime		(getdate())	NO	Date and time the record was last updated.
Name	nvarchar	50		NO	Name of the department.
Name	nvarchar	50		NO	Unique nonclustered index.

### Table Indexes

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
Department	PK_Department_DepartmentID	DepartmentID	CLUSTERED	true	true	false	false	false	false
Department	AK_Department_Name	Name	NONCLUSTERED	true	false	false	false	false	false

### Dependent Objects

Dependent Obj	Dependent Obj Type
vEmployeeDepartment	VIEW

Dependent Obj	Dependent Obj Type
vEmployeeDepartmentHistory	VIEW

## Employee

### Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
BirthDate	datetime			NO	Date of birth.
ContactID	int			NO	Identifies the employee in the Contact table. Foreign key to Contact.ContactID.
ContactID	int			NO	Unique nonclustered index.
CurrentFlag	bit		((1))	NO	0 = Inactive, 1 = Active
EmployeeID	int			NO	Primary key for Employee records.
EmployeeID	int			NO	Clustered index created by a primary key constraint.
Gender	nchar	1		NO	M = Male, F = Female
HireDate	datetime			NO	Employee hired on this date.
LoginID	nvarchar	256		NO	Network login.
LoginID	nvarchar	256		NO	Unique nonclustered index. Used to support replication samples.
ManagerID	int			YES	Manager to whom the employee is assigned. Foreign Key to Employee.M
ManagerID	int			YES	Nonclustered index.
MaritalStatus	nchar	1		NO	M = Married, S = Single
ModifiedDate	datetime		(getdate())	NO	Date and time the record was last updated.
NationalIDNumber	nvarchar	15		NO	Unique national identification number such as a social security number.
NationalIDNumber	nvarchar	15		NO	Unique nonclustered index.

Column Name	Data Type	Length	Column Default	Is Nullable	Description
rowguid	uniqueidentifier		(newid())	NO	ROWGUIDCOL number uniquely identifying the record. Used to support a merge replication sample.
SalariedFlag	bit		((1))	NO	Job classification. 0 = Hourly, not exempt from collective bargaining. 1 = Salaried, exempt from collective bargaining.
SickLeaveHours	smallint		((0))	NO	Number of available sick leave hours.
Title	nvarchar	50		NO	Work title such as Buyer or Sales Representative.
VacationHours	smallint		((0))	NO	Number of available vacation hours.

#### Table Indexes

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
Employee	PK_Employee_EmployeeID	EmployeeID	CLUSTERED	true	true	false	false	false	false
Employee	AK_Employee_LoginID	LoginID	NONCLUSTERED	true	false	false	false	false	false
Employee	AK_Employee_NationalIDNumber	NationalIDNumber	NONCLUSTERED	true	false	false	false	false	false
Employee	AK_Employee_rowguid	rowguid	NONCLUSTERED	true	false	false	false	false	false
Employee	IX_Employee_ManagerID	ManagerID	NONCLUSTERED	false	false	false	false	false	false

#### Dependent Objects

Dependent Obj	Dependent Obj Type
uspGetEmployeeManagers	SQL_STORED_PROCEDURE
uspGetManagerEmployees	SQL_STORED_PROCEDURE
uspUpdateEmployeeHireInfo	SQL_STORED_PROCEDURE
uspUpdateEmployeeLogin	SQL_STORED_PROCEDURE
uspUpdateEmployeePersonalInfo	SQL_STORED_PROCEDURE
ufnGetContactInformation	SQL_TABLE_VALUED_FUNCTION
vEmployee	VIEW
vEmployeeDepartment	VIEW
vEmployeeDepartmentHistory	VIEW
vSalesPerson	VIEW
vSalesPersonSalesByFiscalYears	VIEW

## EmployeeAddress

### Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
AddressID	int			NO	Primary key. Foreign key to Address.AddressID.
AddressID	int			NO	Unique nonclustered index. Used to support replication samples.
EmployeeID	int			NO	Primary key. Foreign key to Employee.EmployeeID.
EmployeeID	int			NO	Clustered index created by a primary key constraint.
ModifiedDate	datetime		(getdate())	NO	Date and time the record was last updated.

Column Name	Data Type	Length	Column Default	Is Nullable	Description
rowguid	uniqueidentifier		(newid())	NO	ROWGUIDCOL number uniquely identifying the record. Used to support a merge replication sample.

#### Table Indexes

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
EmployeeAddress	PK_EmployeeAddress_EmployeeID_AddressID	EmployeeID	CLUSTERED	true	true	false	false	false	false
EmployeeAddress	PK_EmployeeAddress_EmployeeID_AddressID	AddressID	CLUSTERED	true	true	false	false	false	false
EmployeeAddress	AK_EmployeeAddress_rowguid	rowguid	NONCLUSTERED	true	false	false	false	false	false

#### Dependent Objects

Dependent Obj	Dependent Obj Type
vEmployee	VIEW
vSalesPerson	VIEW

## EmployeeDepartmentHistory

### Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
DepartmentID	smallint			NO	Department in which the employee worked including currently. Foreign key to Department.DepartmentID.
DepartmentID	smallint			NO	Nonclustered index.
EmployeeID	int			NO	Employee identification number. Foreign key to Employee.EmployeeID.
EmployeeID	int			NO	Clustered index created by a primary key constraint.
EndDate	datetime			YES	Date the employee left the department. NULL = Current department.
ModifiedDate	datetime		(getdate())	NO	Date and time the record was last updated.
ShiftID	tinyint			NO	Identifies which 8-hour shift the employee works. Foreign key to Shift.Shift.ID.
ShiftID	tinyint			NO	Nonclustered index.
StartDate	datetime			NO	Date the employee started work in the department.

#### Table Indexes

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
EmployeeDepartmentHistory	PK_EmployeeDepartmentHistory_EmployeeID_StartDate_DepartmentID	EmployeeID	CLUSTERED	true	true	false	false	false	false
EmployeeDepartmentHistory	PK_EmployeeDepartmentHistory_EmployeeID_StartDate_DepartmentID	DepartmentID	CLUSTERED	true	true	false	false	false	false
EmployeeDepartmentHistory	PK_EmployeeDepartmentHistory_EmployeeID_StartDate_DepartmentID	ShiftID	CLUSTERED	true	true	false	false	false	false



Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
EmployeeDepartmentHistory	PK_EmployeeDepartmentHistory_EmployeeID_StartDate_DepartmentID	StartDate	CLUSTERED	true	true	false	false	false	false
EmployeeDepartmentHistory	IX_EmployeeDepartmentHistory_DepartmentID	DepartmentID	NONCLUSTERED	false	false	false	false	false	false
EmployeeDepartmentHistory	IX_EmployeeDepartmentHistory_ShiftID	ShiftID	NONCLUSTERED	false	false	false	false	false	false

#### Dependent Objects

Dependent Obj	Dependent Obj Type
vEmployeeDepartment	VIEW
vEmployeeDepartmentHistory	VIEW

## EmployeePayHistory

#### Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
EmployeeID	int			NO	Employee identification number. Foreign key to Employee.EmployeeID.
EmployeeID	int			NO	Clustered index created by a primary key constraint.
ModifiedDate	datetime		(getdate())	NO	Date and time the record was last updated.
PayFrequency	tinyint			NO	1 = Salary received monthly, 2 = Salary received biweekly

Column Name	Data Type	Length	Column Default	Is Nullable	Description
Rate	money			NO	Salary hourly rate.
RateChangeDate	datetime			NO	Date the change in pay is effective

#### Table Indexes

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
EmployeePayHistory	PK_EmployeePayHistory_EmployeeID_RateChangeDate	EmployeeID	CLUSTERED	true	true	false	false	false	false
EmployeePayHistory	PK_EmployeePayHistory_EmployeeID_RateChangeDate	RateChangeDate	CLUSTERED	true	true	false	false	false	false

#### Dependent Objects

Dependent Obj	Dependent Obj Type
uspUpdateEmployeeHireInfo	SQL_STORED_PROCEDURE

### JobCandidate

#### Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
EmployeeID	int			YES	Employee identification number if applicant was hired. Foreign key to Employee.EmployeeID.

Column Name	Data Type	Length	Column Default	Is Nullable	Description
EmployeeID	int			YES	Nonclustered index.
JobCandidateID	int			NO	Primary key for JobCandidate records.
JobCandidateID	int			NO	Clustered index created by a primary key constraint.
ModifiedDate	datetime		(getdate())	NO	Date and time the record was last updated.
Resume	xml	-1		YES	Résumé in XML format.

#### Table Indexes

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
JobCandidate	PK_JobCandidate_JobCandidateID	JobCandidateID	CLUSTERED	true	true	false	false	false	false
JobCandidate	IX_JobCandidate_EmployeeID	EmployeeID	NONCLUSTERED	false	false	false	false	false	false

#### Dependent Objects

Dependent Obj	Dependent Obj Type
vJobCandidate	VIEW
vJobCandidateEducation	VIEW
vJobCandidateEmployment	VIEW

## Shift

### Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
EndTime	datetime			NO	Shift end time.
ModifiedDate	datetime		(getdate())	NO	Date and time the record was last updated.
Name	nvarchar	50		NO	Shift description.
Name	nvarchar	50		NO	Unique nonclustered index.
ShiftID	tinyint			NO	Primary key for Shift records.
ShiftID	tinyint			NO	Clustered index created by a primary key constraint.
StartTime	datetime			NO	Shift start time.
StartTime	datetime			NO	Unique nonclustered index.

### Table Indexes

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
Shift	PK_Shift_ShiftID	ShiftID	CLUSTERED	true	true	false	false	false	false
Shift	AK_Shift_Name	Name	NONCLUSTERED	true	false	false	false	false	false
Shift	AK_Shift_StartTime_EndTime	StartTime	NONCLUSTERED	true	false	false	false	false	false
Shift	AK_Shift_StartTime_EndTime	EndTime	NONCLUSTERED	true	false	false	false	false	false

### Dependent Objects

Dependent Obj	Dependent Obj Type
vEmployeeDepartmentHistory	VIEW

## Address

### Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
AddressID	int			NO	Primary key for Address records.
AddressID	int			NO	Clustered index created by a primary key constraint.
AddressLine1	nvarchar	60		NO	First street address line.
AddressLine1	nvarchar	60		NO	Unique nonclustered index. Used to support replication samples.
AddressLine2	nvarchar	60		YES	Second street address line.
AddressLine2	nvarchar	60		YES	Nonclustered index.
City	nvarchar	30		NO	Name of the city.
City	nvarchar	30		NO	Nonclustered index.
ModifiedDate	datetime		(getdate())	NO	Date and time the record was last updated.
PostalCode	nvarchar	15		NO	Postal code for the street address.
rowguid	uniqueidentifier		(newid())	NO	ROWGUIDCOL number uniquely identifying the record. Used to support a merge replication sample.
StateProvinceID	int			NO	Unique identification number for the state or province. Foreign key to StateProvince table.

### Table Indexes

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
Address	PK_Address_AddressID	AddressID	CLUSTERED	true	true	false	false	false	false
Address	AK_Address_rowguid	rowguid	NONCLUSTERED	true	false	false	false	false	false
Address	IX_Address_AddressLine1_AddressLine2_City_StateProvinceID_PostalCode	AddressLine1	NONCLUSTERED	true	false	false	false	false	false
Address	IX_Address_AddressLine1_AddressLine2_City_StateProvinceID_PostalCode	AddressLine2	NONCLUSTERED	true	false	false	false	false	false
Address	IX_Address_AddressLine1_AddressLine2_City_StateProvinceID_PostalCode	City	NONCLUSTERED	true	false	false	false	false	false
Address	IX_Address_AddressLine1_AddressLine2_City_StateProvinceID_PostalCode	StateProvinceID	NONCLUSTERED	true	false	false	false	false	false
Address	IX_Address_AddressLine1_AddressLine2_City_StateProvinceID_PostalCode	PostalCode	NONCLUSTERED	true	false	false	false	false	false
Address	IX_Address_StateProvinceID	StateProvinceID	NONCLUSTERED	false	false	false	false	false	false

#### Dependent Objects

Dependent Obj	Dependent Obj Type
vEmployee	VIEW
vIndividualCustomer	VIEW

Dependent Obj	Dependent Obj Type
vSalesPerson	VIEW
vStoreWithDemographics	VIEW
vVendor	VIEW

## AddressType

### Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
AddressTypeID	int			NO	Primary key for AddressType records.
AddressTypeID	int			NO	Clustered index created by a primary key constraint.
ModifiedDate	datetime		(getdate())	NO	Date and time the record was last updated.
Name	nvarchar	50		NO	Address type description. For example, Billing, Home, or Shipping.
Name	nvarchar	50		NO	Unique nonclustered index. Used to support replication samples.
rowguid	uniqueidentifier		(newid())	NO	ROWGUIDCOL number uniquely identifying the record. Used to support a merge replication sample.
rowguid	uniqueidentifier		(newid())	NO	Unique nonclustered index.

### Table Indexes

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
AddressType	PK_AddressType_AddressTypeID	AddressTypeID	CLUSTERED	true	true	false	false	false	false

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
AddressType	AK_AddressType_rowguid	rowguid	NONCLUSTERED	true	false	false	false	false	false
AddressType	AK_AddressType_Name	Name	NONCLUSTERED	true	false	false	false	false	false

#### Dependent Objects

Dependent Obj	Dependent Obj Type
vIndividualCustomer	VIEW
vStoreWithDemographics	VIEW

## Contact

#### Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
AdditionalContactInfo	xml	-1		YES	Additional contact information about the person stored in xml format.
ContactID	int			NO	Primary key for Contact records.
ContactID	int			NO	Clustered index created by a primary key constraint.
EmailAddress	nvarchar	50		YES	E-mail address for the person.
EmailPromotion	int		((0))	NO	0 = Contact does not wish to receive e-mail promotions, 1 = Contact does wish to receive e-mail promotions from AdventureWorks, 2 = Contact does wish to receive e-mail promotions from AdventureWorks and selected partners.



Column Name	Data Type	Length	Column Default	Is Nullable	Description
FirstName	nvarchar	50		NO	First name of the person.
LastName	nvarchar	50		NO	Last name of the person.
MiddleName	nvarchar	50		YES	Middle name or middle initial of the person.
ModifiedDate	datetime		(getdate())	NO	Date and time the record was last updated.
NameStyle	bit		((0))	NO	0 = The data in FirstName and LastName are stored in western style (first name, last name) order. 1 = Eastern style (last name, first name) order.
NameStyle	bit		((0))	NO	Unique nonclustered index. Used to support replication samples.
PasswordHash	varchar	128		NO	Password for the e-mail account.
PasswordSalt	varchar	10		NO	Random value concatenated with the password string before the password is hashed.
Phone	nvarchar	25		YES	Phone number associated with the person.
rowguid	uniqueidentifier		(newid())	NO	ROWGUIDCOL number uniquely identifying the record. Used to support a merge replication sample.
Suffix	nvarchar	10		YES	Surname suffix. For example, Sr. or Jr.
Title	nvarchar	8		YES	A courtesy title. For example, Mr. or Ms.
Title	nvarchar	8		YES	Nonclustered index.

#### Table Indexes

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
Contact	PK_Contact_ContactID	ContactID	CLUSTERED	true	true	false	false	false	false

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
Contact	AK_Contact_rowguid	rowguid	NONCLUSTERED	true	false	false	false	false	false
Contact	IX_Contact_EmailAddresses	EmailAddress	NONCLUSTERED	false	false	false	false	false	false
Contact	PXML_Contact_AddContact	AdditionalContactInfo	XML	false	false	false	false	false	false

#### Dependent Objects

Dependent Obj	Dependent Obj Type
uspGetEmployeeManagers	SQL_STORED_PROCEDURE
uspGetManagerEmployees	SQL_STORED_PROCEDURE
ufnGetContactInformation	SQL_TABLE_VALUED_FUNCTION
vAdditionalContactInfo	VIEW
vEmployee	VIEW
vEmployeeDepartment	VIEW
vEmployeeDepartmentHistory	VIEW
vIndividualCustomer	VIEW
vSalesPerson	VIEW
vSalesPersonSalesByFiscalYears	VIEW
vStoreWithDemographics	VIEW
vVendor	VIEW

## ContactType

### Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
ContactTypeID	int			NO	Primary key for ContactType records.
ContactTypeID	int			NO	Clustered index created by a primary key constraint.
ModifiedDate	datetime		(getdate())	NO	Date and time the record was last updated.
Name	nvarchar	50		NO	Contact type description.
Name	nvarchar	50		NO	Unique nonclustered index.

### Table Indexes

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
ContactType	PK_ContactType_ContactTypeID	ContactTypeID	CLUSTERED	true	true	false	false	false	false
ContactType	AK_ContactType_Name	Name	NONCLUSTERED	true	false	false	false	false	false

### Dependent Objects

Dependent Obj	Dependent Obj Type
ufnGetContactInformation	SQL_TABLE_VALUED_FUNCTION
vStoreWithDemographics	VIEW
vVendor	VIEW

## CountryRegion

### Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
CountryRegionCode	nvarchar	3		NO	ISO standard code for countries and regions.
CountryRegionCode	nvarchar	3		NO	Clustered index created by a primary key constraint.
ModifiedDate	datetime		(getdate())	NO	Date and time the record was last updated.
Name	nvarchar	50		NO	Country or region name.
Name	nvarchar	50		NO	Unique nonclustered index.

### Table Indexes

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
CountryRegion	PK_CountryRegion_CountryRegionCode	CountryRegionCode	CLUSTERED	true	true	false	false	false	false
CountryRegion	AK_CountryRegion_Name	Name	NONCLUSTERED	true	false	false	false	false	false

### Dependent Objects

Dependent Obj	Dependent Obj Type
vEmployee	VIEW
vIndividualCustomer	VIEW
vSalesPerson	VIEW
vStateProvinceCountryRegion	VIEW
vStoreWithDemographics	VIEW

Dependent Obj	Dependent Obj Type
vVendor	VIEW

## StateProvince

### Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
CountryRegionCode	nvarchar	3		NO	ISO standard country or region code. Foreign key to CountryRegion.CountryRegionCode.
CountryRegionCode	nvarchar	3		NO	Unique nonclustered index.
IsOnlyStateProvinceFlag	bit		((1))	NO	0 = StateProvinceCode exists. 1 = StateProvinceCode unavailable, using CountryRegionCode.
IsOnlyStateProvinceFlag	bit		((1))	NO	Unique nonclustered index. Used to support replication samples.
ModifiedDate	datetime		(getdate())	NO	Date and time the record was last updated.
Name	nvarchar	50		NO	State or province description.
rowguid	uniqueidentifier		(newid())	NO	ROWGUIDCOL number uniquely identifying the record. Used to support a merge replication sample.
StateProvinceCode	nchar	3		NO	ISO standard state or province code.
StateProvinceCode	nchar	3		NO	Unique nonclustered index.
StateProvinceID	int			NO	Primary key for StateProvince records.
StateProvinceID	int			NO	Clustered index created by a primary key constraint.
TerritoryID	int			NO	ID of the territory in which the state or province is located. Foreign key to SalesTerritory.SalesTerritoryID.

## Table Indexes

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
StateProvince	PK_StateProvince_StateProvinceID	StateProvinceID	CLUSTERED	true	true	false	false	false	false
StateProvince	AK_StateProvince_Name	Name	NONCLUSTERED	true	false	false	false	false	false
StateProvince	AK_StateProvince_StateProvinceCode_CountryRegionCode	StateProvinceCode	NONCLUSTERED	true	false	false	false	false	false
StateProvince	AK_StateProvince_StateProvinceCode_CountryRegionCode	CountryRegionCode	NONCLUSTERED	true	false	false	false	false	false
StateProvince	AK_StateProvince_rowguid	rowguid	NONCLUSTERED	true	false	false	false	false	false

## Dependent Objects

Dependent Obj	Dependent Obj Type
vEmployee	VIEW
vIndividualCustomer	VIEW
vSalesPerson	VIEW
vStateProvinceCountryRegion	VIEW
vStoreWithDemographics	VIEW
vVendor	VIEW

## BillOfMaterials

### Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
BillOfMaterialsID	int			NO	Primary key for BillOfMaterials records.
BillOfMaterialsID	int			NO	Clustered index.
BOMLevel	smallint			NO	Indicates the depth the component is from its parent (AssemblyID).
ComponentID	int			NO	Component identification number. Foreign key to Product.ProductID.
ComponentID	int			NO	Nonclustered index.
EndDate	datetime			YES	Date the component stopped being used in the assembly item.
ModifiedDate	datetime		(getdate())	NO	Date and time the record was last updated.
PerAssemblyQty	decimal		((1.00))	NO	Quantity of the component needed to create the assembly.
ProductAssemblyID	int			YES	Parent product identification number. Foreign key to Product.ProductID.
ProductAssemblyID	int			YES	Nonclustered index created by a primary key constraint.
StartDate	datetime		(getdate())	NO	Date the component started being used in the assembly item.
UnitMeasureCode	nchar	3		NO	Standard code identifying the unit of measure for the quantity.

#### Table Indexes

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
BillOfMaterials	AK_BillOfMaterials_ProductAssemblyID_ComponentID_StartDate	ProductAssemblyID	CLUSTERED	true	false	false	false	false	false
BillOfMaterials	AK_BillOfMaterials_ProductAssemblyID_ComponentID_StartDate	ComponentID	CLUSTERED	true	false	false	false	false	false

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
BillOfMaterials	AK_BillOfMaterials_ProductAssemblyID_ComponentID_StartDate	StartDate	CLUSTERED	true	false	false	false	false	false
BillOfMaterials	PK_BillOfMaterials_BillOfMaterialsID	BillOfMaterialsID	NONCLUSTERED	true	true	false	false	false	false
BillOfMaterials	IX_BillOfMaterials_UnitMeasureCode	UnitMeasureCode	NONCLUSTERED	false	false	false	false	false	false

### Dependent Objects

Dependent Obj	Dependent Obj Type
uspGetBillOfMaterials	SQL_STORED_PROCEDURE
uspGetWhereUsedProductID	SQL_STORED_PROCEDURE

## Culture

### Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
CultureID	nchar	6		NO	Primary key for Culture records.
CultureID	nchar	6		NO	Clustered index created by a primary key constraint.
ModifiedDate	datetime		(getdate())	NO	Date and time the record was last updated.
Name	nvarchar	50		NO	Culture description.
Name	nvarchar	50		NO	Unique nonclustered index.



## Table Indexes

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
Culture	PK_Culture_CultureID	CultureID	CLUSTERED	true	true	false	false	false	false
Culture	AK_Culture_Name	Name	NONCLUSTERED	true	false	false	false	false	false

## Dependent Objects

---None---

## Document

### Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
ChangeNumber	int		((0))	NO	Engineering change approval number.
Document	varbinary	-1		YES	Complete document.
DocumentID	int			NO	Primary key for Document records.
DocumentID	int			NO	Clustered index created by a primary key constraint.
DocumentSummary	nvarchar	-1		YES	Document abstract.
FileExtension	nvarchar	8		NO	File extension indicating the document type. For example, .doc or .txt.
FileName	nvarchar	400		NO	Directory path and file name of the document

Column Name	Data Type	Length	Column Default	Is Nullable	Description
ModifiedDate	datetime		(getdate())	NO	Date and time the record was last updated.
Revision	nchar	5		NO	Revision number of the document.
Status	tinyint			NO	1 = Pending approval, 2 = Approved, 3 = Obsolete
Title	nvarchar	50		NO	Title of the document.
Title	nvarchar	50		NO	Unique nonclustered index.

#### Table Indexes

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
Document	PK_Document_DocumentID	DocumentID	CLUSTERED	true	true	false	false	false	false
Document	AK_Document_FileName_Revision	FileName	NONCLUSTERED	true	false	false	false	false	false
Document	AK_Document_FileName_Revision	Revision	NONCLUSTERED	true	false	false	false	false	false

#### Dependent Objects

---None---

### Illustration

Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
Diagram	xml	-1		YES	Illustrations used in manufacturing instructions. Stored as XML.
IllustrationID	int			NO	Primary key for Illustration records.
IllustrationID	int			NO	Clustered index created by a primary key constraint.
ModifiedDate	datetime		(getdate())	NO	Date and time the record was last updated.

#### Table Indexes

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
Illustration	PK_Illustration_IllustrationID	IllustrationID	CLUSTERED	true	true	false	false	false	false

#### Dependent Objects

---None---

### Location

#### Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
Availability	decimal		((0.00))	NO	Work capacity (in hours) of the manufacturing location.
CostRate	smallmoney		((0.00))	NO	Standard hourly cost of the manufacturing location.

Column Name	Data Type	Length	Column Default	Is Nullable	Description
LocationID	smallint			NO	Primary key for Location records.
LocationID	smallint			NO	Clustered index created by a primary key constraint.
ModifiedDate	datetime		(getdate())	NO	Date and time the record was last updated.
Name	nvarchar	50		NO	Location description.
Name	nvarchar	50		NO	Unique nonclustered index.

#### Table Indexes

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
Location	PK_Location_LocationID	LocationID	CLUSTERED	true	true	false	false	false	false
Location	AK_Location_Name	Name	NONCLUSTERED	true	false	false	false	false	false

#### Dependent Objects

---None---

## Product

#### Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
Class	nchar	2		YES	H = High, M = Medium, L = Low

Column Name	Data Type	Length	Column Default	Is Nullable	Description
Color	nvarchar	15		YES	Product color.
DaysToManufacture	int			NO	Number of days required to manufacture the product.
DiscontinuedDate	datetime			YES	Date the product was discontinued.
FinishedGoodsFlag	bit		((1))	NO	0 = Product is not a salable item. 1 = Product is salable.
ListPrice	money			NO	Selling price.
MakeFlag	bit		((1))	NO	0 = Product is purchased, 1 = Product is manufactured in-house.
MakeFlag	bit		((1))	NO	Unique nonclustered index. Used to support replication samples.
ModifiedDate	datetime		(getdate())	NO	Date and time the record was last updated.
Name	nvarchar	50		NO	Name of the product.
Name	nvarchar	50		NO	Unique nonclustered index.
ProductID	int			NO	Primary key for Product records.
ProductID	int			NO	Clustered index created by a primary key constraint.
ProductLine	nchar	2		YES	R = Road, M = Mountain, T = Touring, S = Standard
ProductModelID	int			YES	Product is a member of this product model. Foreign key to ProductModel.ProductModelID.
ProductNumber	nvarchar	25		NO	Unique product identification number.
ProductNumber	nvarchar	25		NO	Unique nonclustered index.
ProductSubcategoryID	int			YES	Product is a member of this product subcategory. Foreign key to ProductSubCategory.ProductSubCategoryID.
ReorderPoint	smallint			NO	Inventory level that triggers a purchase order or work order.
rowguid	uniqueidentifier		(newid())	NO	ROWGUIDCOL number uniquely identifying the record. Used to support a merge replication sample.

Column Name	Data Type	Length	Column Default	Is Nullable	Description
SafetyStockLevel	smallint			NO	Minimum inventory quantity.
SellEndDate	datetime			YES	Date the product was no longer available for sale.
SellStartDate	datetime			NO	Date the product was available for sale.
Size	nvarchar	5		YES	Product size.
SizeUnitMeasureCode	nchar	3		YES	Unit of measure for Size column.
StandardCost	money			NO	Standard cost of the product.
Style	nchar	2		YES	W = Womens, M = Mens, U = Universal
Weight	decimal			YES	Product weight.
WeightUnitMeasureCode	nchar	3		YES	Unit of measure for Weight column.

#### Table Indexes

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
Product	PK_Product_ProductID	ProductID	CLUSTERED	true	true	false	false	false	false
Product	AK_Product_ProductNumber	ProductNumber	NONCLUSTERED	true	false	false	false	false	false
Product	AK_Product_Name	Name	NONCLUSTERED	true	false	false	false	false	false
Product	AK_Product_rowguid	rowguid	NONCLUSTERED	true	false	false	false	false	false

#### Dependent Objects

Dependent Obj	Dependent Obj Type
ufnGetProductDealerPrice	SQL_SCALAR_FUNCTION
ufnGetProductListPrice	SQL_SCALAR_FUNCTION
ufnGetProductStandardCost	SQL_SCALAR_FUNCTION
uspGetBillOfMaterials	SQL_STORED_PROCEDURE
uspGetWhereUsedProductID	SQL_STORED_PROCEDURE
vProductAndDescription	VIEW

## ProductCategory

### Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
ModifiedDate	datetime		(getdate())	NO	Date and time the record was last updated.
Name	nvarchar	50		NO	Category description.
Name	nvarchar	50		NO	Unique nonclustered index.
ProductCategoryID	int			NO	Primary key for ProductCategory records.
ProductCategoryID	int			NO	Clustered index created by a primary key constraint.
rowguid	uniqueidentifier		(newid())	NO	ROWGUIDCOL number uniquely identifying the record. Used to support a merge replication sample.
rowguid	uniqueidentifier		(newid())	NO	Unique nonclustered index. Used to support replication samples.

### Table Indexes

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
ProductCategory	PK_ProductCategory_ProductCategoryID	ProductCategoryID	CLUSTERED	true	true	false	false	false	false
ProductCategory	AK_ProductCategory_Name	Name	NONCLUSTERED	true	false	false	false	false	false
ProductCategory	AK_ProductCategory_rowguid	rowguid	NONCLUSTERED	true	false	false	false	false	false

Dependent Objects

---None---

## ProductCostHistory

Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
EndDate	datetime			YES	Product cost end date.
ModifiedDate	datetime		(getdate())	NO	Date and time the record was last updated.
ProductID	int			NO	Product identification number. Foreign key to Product.ProductID
ProductID	int			NO	Clustered index created by a primary key constraint.
StandardCost	money			NO	Standard cost of the product.
StartDate	datetime			NO	Product cost start date.



## Table Indexes

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
ProductCostHistory	PK_ProductCostHistory_ProductID_StartDate	ProductID	CLUSTERED	true	true	false	false	false	false
ProductCostHistory	PK_ProductCostHistory_ProductID_StartDate	StartDate	CLUSTERED	true	true	false	false	false	false

## Dependent Objects

Dependent Obj	Dependent Obj Type
ufnGetProductStandardCost	SQL_SCALAR_FUNCTION

## ProductDescription

### Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
Description	nvarchar	400		NO	Description of the product.
Description	nvarchar	400		NO	Unique nonclustered index. Used to support replication samples.
ModifiedDate	datetime		(getdate())	NO	Date and time the record was last updated.
ProductDescriptionID	int			NO	Primary key for ProductDescription records.
ProductDescriptionID	int			NO	Clustered index created by a primary key constraint.
rowguid	uniqueidentifier		(newid())	NO	ROWGUIDCOL number uniquely identifying the record. Used to support a merge replication sample.

## Table Indexes

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
ProductDescription	PK_ProductDescription_ProductDescriptionID	ProductDescriptionID	CLUSTERED	true	true	false	false	false	false
ProductDescription	AK_ProductDescription_rowguid	rowguid	NONCLUSTERED	true	false	false	false	false	false

## Dependent Objects

Dependent Obj	Dependent Obj Type
vProductAndDescription	VIEW

## ProductDocument

### Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
DocumentID	int			NO	Document identification number. Foreign key to Document.DocumentID.
ModifiedDate	datetime		(getdate())	NO	Date and time the record was last updated.
ProductID	int			NO	Product identification number. Foreign key to Product.ProductID.
ProductID	int			NO	Clustered index created by a primary key constraint.

## Table Indexes

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
ProductDocument	PK_ProductDocument_ProductID_DocumentID	ProductID	CLUSTERED	true	true	false	false	false	false
ProductDocument	PK_ProductDocument_ProductID_DocumentID	DocumentID	CLUSTERED	true	true	false	false	false	false

### Dependent Objects

---None---

## ProductInventory

### Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
Bin	tinyint			NO	Storage container on a shelf in an inventory location.
LocationID	smallint			NO	Inventory location identification number. Foreign key to Location.LocationID.
ModifiedDate	datetime		(getdate())	NO	Date and time the record was last updated.
ProductID	int			NO	Product identification number. Foreign key to Product.ProductID.
ProductID	int			NO	Clustered index created by a primary key constraint.
Quantity	smallint		((0))	NO	Quantity of products in the inventory location.
rowguid	uniqueidentifier		(newid())	NO	ROWGUIDCOL number uniquely identifying the record. Used to support a merge replication sample.
Shelf	nvarchar	10		NO	Storage compartment within an inventory location.

## Table Indexes

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
ProductInventory	PK_ProductInventory_ProductID_LocationID	ProductID	CLUSTERED	true	true	false	false	false	false
ProductInventory	PK_ProductInventory_ProductID_LocationID	LocationID	CLUSTERED	true	true	false	false	false	false

## Dependent Objects

Dependent Obj	Dependent Obj Type
ufnGetStock	SQL_SCALAR_FUNCTION

## ProductListPriceHistory

### Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
EndDate	datetime			YES	List price end date
ListPrice	money			NO	Product list price.
ModifiedDate	datetime		(getdate())	NO	Date and time the record was last updated.
ProductID	int			NO	Product identification number. Foreign key to Product.ProductID
ProductID	int			NO	Clustered index created by a primary key constraint.
StartDate	datetime			NO	List price start date.

## Table Indexes

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
ProductListPriceHistory	PK_ProductListPriceHistory_ProductID_StartDate	ProductID	CLUSTERED	true	true	false	false	false	false
ProductListPriceHistory	PK_ProductListPriceHistory_ProductID_StartDate	StartDate	CLUSTERED	true	true	false	false	false	false

## Dependent Objects

Dependent Obj	Dependent Obj Type
ufnGetProductDealerPrice	SQL_SCALAR_FUNCTION
ufnGetProductListPrice	SQL_SCALAR_FUNCTION

## ProductModel

### Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
CatalogDescription	xml	-1		YES	Detailed product catalog information in xml format.
CatalogDescription	xml	-1		YES	Unique nonclustered index. Used to support replication samples.
Instructions	xml	-1		YES	Manufacturing instructions in xml format.
ModifiedDate	datetime		(getdate())	NO	Date and time the record was last updated.
Name	nvarchar	50		NO	Product model description.

Column Name	Data Type	Length	Column Default	Is Nullable	Description
Name	nvarchar	50		NO	Unique nonclustered index.
ProductModelID	int			NO	Primary key for ProductModel records.
ProductModelID	int			NO	Clustered index created by a primary key constraint.
rowguid	uniqueidentifier		(newid())	NO	ROWGUIDCOL number uniquely identifying the record. Used to support a merge replication sample.

#### Table Indexes

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
ProductModel	PK_ProductModel_ProductModelID	ProductModelID	CLUSTERED	true	true	false	false	false	false
ProductModel	AK_ProductModel_Name	Name	NONCLUSTERED	true	false	false	false	false	false
ProductModel	AK_ProductModel_rowguid	rowguid	NONCLUSTERED	true	false	false	false	false	false
ProductModel	PXML_ProductModel_CatalogDescription	CatalogDescription	XML	false	false	false	false	false	false
ProductModel	PXML_ProductModel_Instructions	Instructions	XML	false	false	false	false	false	false

#### Dependent Objects

Dependent Obj	Dependent Obj Type
vProductAndDescription	VIEW
vProductModelCatalogDescription	VIEW

Dependent Obj	Dependent Obj Type
vProductModelInstructions	VIEW

## ProductModelIllustration

### Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
IllustrationID	int			NO	Primary key. Foreign key to Illustration.IllustrationID.
ModifiedDate	datetime		(getdate())	NO	Date and time the record was last updated.
ProductModelID	int			NO	Primary key. Foreign key to ProductModel.ProductModelID.
ProductModelID	int			NO	Clustered index created by a primary key constraint.

### Table Indexes

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
ProductModelIllustration	PK_ProductModelIllustration_ProductModelID_IllustrationID	ProductModelID	CLUSTERED	true	true	false	false	false	false
ProductModelIllustration	PK_ProductModelIllustration_ProductModelID_IllustrationID	IllustrationID	CLUSTERED	true	true	false	false	false	false

## Dependent Objects

---None---

## ProductModelProductDescriptionCulture

### Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
CultureID	nchar	6		NO	Culture identification number. Foreign key to Culture.CultureID.
ModifiedDate	datetime		(getdate())	NO	Date and time the record was last updated.
ProductDescriptionID	int			NO	Primary key. Foreign key to ProductDescription.ProductDescriptionID.
ProductModelID	int			NO	Primary key. Foreign key to ProductModel.ProductModelID.
ProductModelID	int			NO	Clustered index created by a primary key constraint.

### Table Indexes

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
ProductModelProductDescriptionCulture	PK_ProductModelProductDescriptionCulture_ProductModelID_ProductDescriptionID_CultureID	ProductModelID	CLUSTERED	true	true	false	false	false	false
ProductModelProductDescriptionCulture	PK_ProductModelProductDescriptionCulture_ProductModelID_ProductDescriptionID_CultureID	ProductDescriptionID	CLUSTERED	true	true	false	false	false	false



Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
ProductModelProductDescriptionCulture	PK_ProductModelProductDescriptionCulture_ProductModelID_ProductDescriptionID_CultureID	CultureID	CLUSTERED	true	true	false	false	false	false

#### Dependent Objects

Dependent Obj	Dependent Obj Type
vProductAndDescription	VIEW

## ProductPhoto

#### Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
LargePhoto	varbinary	-1		YES	Large image of the product.
LargePhotoFileName	nvarchar	50		YES	Large image file name.
ModifiedDate	datetime		(getdate())	NO	Date and time the record was last updated.
ProductPhotoID	int			NO	Primary key for ProductPhoto records.
ProductPhotoID	int			NO	Clustered index created by a primary key constraint.
ThumbNailPhoto	varbinary	-1		YES	Small image of the product.
ThumbNailPhotoFileName	nvarchar	50		YES	Small image file name.

## Table Indexes

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
ProductPhoto	PK_ProductPhoto_ProductPhotoID	ProductPhotoID	CLUSTERED	true	true	false	false	false	false

## Dependent Objects

---None---

## ProductProductPhoto

### Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
ModifiedDate	datetime		(getdate())	NO	Date and time the record was last updated.
Primary	bit		((0))	NO	0 = Photo is not the principal image. 1 = Photo is the principal image.
ProductID	int			NO	Product identification number. Foreign key to Product.ProductID.
ProductPhotoID	int			NO	Product photo identification number. Foreign key to ProductPhoto.ProductPhotoID.
ProductPhotoID	int			NO	Nonclustered index created by a primary key constraint.

## Table Indexes

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
ProductPhoto	PK_ProductPhoto_ProductID_ProductPhotoID	ProductID	NONCLUSTERED	true	true	false	false	false	false
ProductPhoto	PK_ProductPhoto_ProductPhotoID_ProductPhotoID	ProductPhotoID	NONCLUSTERED	true	true	false	false	false	false

### Dependent Objects

---None---

## ProductReview

### Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
Comments	nvarchar	3850		YES	Reviewer's comments
EmailAddress	nvarchar	50		NO	Reviewer's e-mail address.
ModifiedDate	datetime		(getdate())	NO	Date and time the record was last updated.
ProductID	int			NO	Product identification number. Foreign key to Product.ProductID.
ProductID	int			NO	Nonclustered index.
ProductReviewID	int			NO	Primary key for ProductReview records.
ProductReviewID	int			NO	Clustered index created by a primary key constraint.
Rating	int			NO	Product rating given by the reviewer. Scale is 1 to 5 with 5 as the highest rating.
ReviewDate	datetime		(getdate())	NO	Date review was submitted.

Column Name	Data Type	Length	Column Default	Is Nullable	Description
ReviewerName	nvarchar	50		NO	Name of the reviewer.

#### Table Indexes

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
ProductReview	PK_ProductReview_ProductReviewID	ProductReviewID	CLUSTERED	true	true	false	false	false	false
ProductReview	IX_ProductReview_ProductID_Name	ProductID	NONCLUSTERED	false	false	false	false	false	false
ProductReview	IX_ProductReview_ProductID_Name	ReviewerName	NONCLUSTERED	false	false	false	false	false	false
ProductReview	IX_ProductReview_ProductID_Name	Comments	NONCLUSTERED	false	false	false	false	false	false

#### Dependent Objects

---None---

### ProductSubcategory

#### Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
ModifiedDate	datetime		(getdate())	NO	Date and time the record was last updated.
Name	nvarchar	50		NO	Subcategory description.

Column Name	Data Type	Length	Column Default	Is Nullable	Description
Name	nvarchar	50		NO	Unique nonclustered index. Used to support replication samples.
ProductCategoryID	int			NO	Product category identification number. Foreign key to ProductCategory.ProductCategoryID.
ProductCategoryID	int			NO	Unique nonclustered index.
ProductSubcategoryID	int			NO	Primary key for ProductSubcategory records.
ProductSubcategoryID	int			NO	Clustered index created by a primary key constraint.
rowguid	uniqueidentifier		(newid())	NO	ROWGUIDCOL number uniquely identifying the record. Used to support a merge replication sample.

#### Table Indexes

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
ProductSubcategory	PK_ProductSubcategory_ProductSubcategoryID	ProductSubcategoryID	CLUSTERED	true	true	false	false	false	false
ProductSubcategory	AK_ProductSubcategory_Name	Name	NONCLUSTERED	true	false	false	false	false	false
ProductSubcategory	AK_ProductSubcategory_rowguid	rowguid	NONCLUSTERED	true	false	false	false	false	false

#### Dependent Objects

---None---

## ScrapReason

### Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
ModifiedDate	datetime		(getdate())	NO	Date and time the record was last updated.
Name	nvarchar	50		NO	Failure description.
Name	nvarchar	50		NO	Unique nonclustered index.
ScrapReasonID	smallint			NO	Primary key for ScrapReason records.
ScrapReasonID	smallint			NO	Clustered index created by a primary key constraint.

### Table Indexes

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
ScrapReason	PK_ScrapReason_ScrapReasonID	ScrapReasonID	CLUSTERED	true	true	false	false	false	false
ScrapReason	AK_ScrapReason_Name	Name	NONCLUSTERED	true	false	false	false	false	false

### Dependent Objects

---None---



Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
TransactionHistory	IX_TransactionHistory_ReferenceOrderID_ReferenceOrderLineID	ReferenceOrderID	NONCLUSTERED	false	false	false	false	false	false
TransactionHistory	IX_TransactionHistory_ReferenceOrderID_ReferenceOrderLineID	ReferenceOrderLineID	NONCLUSTERED	false	false	false	false	false	false

#### Dependent Objects

Dependent Obj	Dependent Obj Type
iduSalesOrderDetail	SQL_TRIGGER
iPurchaseOrderDetail	SQL_TRIGGER
iWorkOrder	SQL_TRIGGER
uPurchaseOrderDetail	SQL_TRIGGER
uWorkOrder	SQL_TRIGGER

## TransactionHistoryArchive

#### Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
ActualCost	money			NO	Product cost.
ModifiedDate	datetime		(getdate())	NO	Date and time the record was last updated.
ProductID	int			NO	Product identification number. Foreign key to Product.ProductID.
ProductID	int			NO	Nonclustered index.





## Dependent Objects

---None---

## UnitMeasure

### Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
ModifiedDate	datetime		(getdate())	NO	Date and time the record was last updated.
Name	nvarchar	50		NO	Unit of measure description.
Name	nvarchar	50		NO	Unique nonclustered index.
UnitMeasureCode	nchar	3		NO	Primary key.
UnitMeasureCode	nchar	3		NO	Clustered index created by a primary key constraint.

### Table Indexes

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
UnitMeasure	PK_UnitMeasure_UnitMeasureCode	UnitMeasureCode	CLUSTERED	true	true	false	false	false	false
UnitMeasure	AK_UnitMeasure_Name	Name	NONCLUSTERED	true	false	false	false	false	false

## Dependent Objects

---None---

## WorkOrder

### Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
DueDate	datetime			NO	Work order due date.
EndDate	datetime			YES	Work order end date.
ModifiedDate	datetime		(getdate())	NO	Date and time the record was last updated.
OrderQty	int			NO	Product quantity to build.
OrderQty	int			NO	Nonclustered index.
ProductID	int			NO	Product identification number. Foreign key to Product.ProductID.
ProductID	int			NO	Nonclustered index.
ScrappedQty	smallint			NO	Quantity that failed inspection.
ScrapReasonID	smallint			YES	Reason for inspection failure.
StartDate	datetime			NO	Work order start date.
StockedQty	int			NO	Quantity built and put in inventory.
WorkOrderID	int			NO	Primary key for WorkOrder records.
WorkOrderID	int			NO	Clustered index created by a primary key constraint.

### Table Indexes

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
WorkOrder	PK_WorkOrder_WorkOrderID	WorkOrderID	CLUSTERED	true	true	false	false	false	false

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
WorkOrder	IX_WorkOrder_ScrapReasonID	ScrapReasonID	NONCLUSTERED	false	false	false	false	false	false
WorkOrder	IX_WorkOrder_ProductID	ProductID	NONCLUSTERED	false	false	false	false	false	false

### Dependent Objects

Dependent Obj	Dependent Obj Type
uWorkOrder	SQL_TRIGGER
WorkOrder	USER_TABLE

## WorkOrderRouting

### Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
ActualCost	money			YES	Actual manufacturing cost.
ActualEndDate	datetime			YES	Actual end date.
ActualResourceHrs	decimal			YES	Number of manufacturing hours used.
ActualStartDate	datetime			YES	Actual start date.
LocationID	smallint			NO	Manufacturing location where the part is processed. Foreign key to Location.LocationID.
ModifiedDate	datetime		(getdate())	NO	Date and time the record was last updated.
OperationSequence	smallint			NO	Primary key. Indicates the manufacturing process sequence.



## Dependent Objects

---None---

## ProductVendor

### Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
AverageLeadTime	int			NO	The average span of time (in days) between placing an order with the vendor and receiving the purchased product.
AverageLeadTime	int			NO	Nonclustered index.
LastReceiptCost	money			YES	The selling price when last purchased.
LastReceiptDate	datetime			YES	Date the product was last received by the vendor.
MaxOrderQty	int			NO	The minimum quantity that should be ordered.
MinOrderQty	int			NO	The maximum quantity that should be ordered.
ModifiedDate	datetime		(getdate())	NO	Date and time the record was last updated.
OnOrderQty	int			YES	The quantity currently on order.
ProductID	int			NO	Primary key. Foreign key to Product.ProductID.
ProductID	int			NO	Clustered index created by a primary key constraint.
StandardPrice	money			NO	The vendor's usual selling price.
UnitMeasureCode	nchar	3		NO	The product's unit of measure.
VendorID	int			NO	Primary key. Foreign key to Vendor.VendorID.
VendorID	int			NO	Nonclustered index.

## Table Indexes

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
ProductVendor	PK_ProductVendor_ProductID_VendorID	ProductID	CLUSTERED	true	true	false	false	false	false
ProductVendor	PK_ProductVendor_ProductID_VendorID	VendorID	CLUSTERED	true	true	false	false	false	false
ProductVendor	IX_ProductVendor_UnitMeasureCode	UnitMeasureCode	NONCLUSTERED	false	false	false	false	false	false
ProductVendor	IX_ProductVendor_VendorID	VendorID	NONCLUSTERED	false	false	false	false	false	false

## Dependent Objects

---None---

## PurchaseOrderDetail

### Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
DueDate	datetime			NO	Date the product is expected to be received.
LineTotal	money			NO	Per product subtotal. Computed as OrderQty * UnitPrice.
ModifiedDate	datetime		(getdate())	NO	Date and time the record was last updated.
OrderQty	smallint			NO	Quantity ordered.
ProductID	int			NO	Product identification number. Foreign key to Product.ProductID.
PurchaseOrderDetailID	int			NO	Primary key. One line number per purchased product.
PurchaseOrderDetailID	int			NO	Nonclustered index.

Column Name	Data Type	Length	Column Default	Is Nullable	Description
PurchaseOrderID	int			NO	Primary key. Foreign key to PurchaseOrderHeader.PurchaseOrderID.
PurchaseOrderID	int			NO	Clustered index created by a primary key constraint.
ReceivedQty	decimal			NO	Quantity actually received from the vendor.
RejectedQty	decimal			NO	Quantity rejected during inspection.
StockedQty	decimal			NO	Quantity accepted into inventory. Computed as ReceivedQty - RejectedQty.
UnitPrice	money			NO	Vendor's selling price of a single product.

#### Table Indexes

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
PurchaseOrderDetail	PK_PurchaseOrderDetail_PurchaseOrderID_PurchaseOrderDetailID	PurchaseOrderID	CLUSTERED	true	true	false	false	false	false
PurchaseOrderDetail	PK_PurchaseOrderDetail_PurchaseOrderID_PurchaseOrderDetailID	PurchaseOrderDetailID	CLUSTERED	true	true	false	false	false	false
PurchaseOrderDetail	IX_PurchaseOrderDetail_ProductID	ProductID	NONCLUSTERED	false	false	false	false	false	false

#### Dependent Objects

Dependent Obj	Dependent Obj Type
iPurchaseOrderDetail	SQL_TRIGGER
uPurchaseOrderDetail	SQL_TRIGGER



Dependent Obj	Dependent Obj Type
PurchaseOrderDetail	USER_TABLE

## PurchaseOrderHeader

### Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
EmployeeID	int			NO	Employee who created the purchase order. Foreign key to Employee.EmployeeID.
Freight	money		((0.00))	NO	Shipping cost.
ModifiedDate	datetime		(getdate())	NO	Date and time the record was last updated.
OrderDate	datetime		(getdate())	NO	Purchase order creation date.
PurchaseOrderID	int			NO	Primary key.
PurchaseOrderID	int			NO	Clustered index created by a primary key constraint.
RevisionNumber	tinyint		((0))	NO	Incremental number to track changes to the purchase order over time.
RevisionNumber	tinyint		((0))	NO	Nonclustered index.
ShipDate	datetime			YES	Estimated shipment date from the vendor.
ShipMethodID	int			NO	Shipping method. Foreign key to ShipMethod.ShipMethodID.
Status	tinyint		((1))	NO	Order current status. 1 = Pending; 2 = Approved; 3 = Rejected; 4 = Complete
Status	tinyint		((1))	NO	Nonclustered index.
SubTotal	money		((0.00))	NO	Purchase order subtotal. Computed as SUM(PurchaseOrderDetail.LineTotal)for the appropriate PurchaseOrderID.
TaxAmt	money		((0.00))	NO	Tax amount.
TotalDue	money			NO	Total due to vendor. Computed as Subtotal + TaxAmt + Freight.

Column Name	Data Type	Length	Column Default	Is Nullable	Description
VendorID	int			NO	Vendor with whom the purchase order is placed. Foreign key to Vendor.VendorID.

#### Table Indexes

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
PurchaseOrderHeader	PK_PurchaseOrderHeader_PurchaseOrderID	PurchaseOrderID	CLUSTERED	true	true	false	false	false	false
PurchaseOrderHeader	IX_PurchaseOrderHeader_VendorID	VendorID	NONCLUSTERED	false	false	false	false	false	false
PurchaseOrderHeader	IX_PurchaseOrderHeader_EmployeeID	EmployeeID	NONCLUSTERED	false	false	false	false	false	false

#### Dependent Objects

Dependent Obj	Dependent Obj Type
iPurchaseOrderDetail	SQL_TRIGGER
uPurchaseOrderDetail	SQL_TRIGGER
uPurchaseOrderHeader	SQL_TRIGGER
PurchaseOrderHeader	USER_TABLE

## ShipMethod

### Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
ModifiedDate	datetime		(getdate())	NO	Date and time the record was last updated.
Name	nvarchar	50		NO	Shipping company name.
Name	nvarchar	50		NO	Unique nonclustered index.
rowguid	uniqueidentifier		(newid())	NO	ROWGUIDCOL number uniquely identifying the record. Used to support a merge replication sample.
ShipBase	money		((0.00))	NO	Minimum shipping charge.
ShipBase	money		((0.00))	NO	Unique nonclustered index. Used to support replication samples.
ShipMethodID	int			NO	Primary key for ShipMethod records.
ShipMethodID	int			NO	Clustered index created by a primary key constraint.
ShipRate	money		((0.00))	NO	Shipping charge per pound.

#### Table Indexes

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
ShipMethod	PK_ShipMethod_ShipMethodID	ShipMethodID	CLUSTERED	true	true	false	false	false	false
ShipMethod	AK_ShipMethod_Name	Name	NONCLUSTERED	true	false	false	false	false	false
ShipMethod	AK_ShipMethod_rowguid	rowguid	NONCLUSTERED	true	false	false	false	false	false

#### Dependent Objects

---None---

## Vendor

### Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
AccountNumber	nvarchar	15		NO	Vendor account (identification) number.
AccountNumber	nvarchar	15		NO	Unique nonclustered index.
ActiveFlag	bit		((1))	NO	0 = Vendor no longer used. 1 = Vendor is actively used.
CreditRating	tinyint			NO	1 = Superior, 2 = Excellent, 3 = Above average, 4 = Average, 5 = Below average
ModifiedDate	datetime		(getdate())	NO	Date and time the record was last updated.
Name	nvarchar	50		NO	Company name.
PreferredVendorStatus	bit		((1))	NO	0 = Do not use if another vendor is available. 1 = Preferred over other vendors supplying the same product.
PurchasingWebServiceURL	nvarchar	1024		YES	Vendor URL.
VendorID	int			NO	Primary key for Vendor records.
VendorID	int			NO	Clustered index created by a primary key constraint.

### Table Indexes

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
Vendor	PK_Vendor_VendorID	VendorID	CLUSTERED	true	true	false	false	false	false

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
Vendor	AK_Vendor_AccountNumber	AccountNumber	NONCLUSTERED	true	false	false	false	false	false

#### Dependent Objects

Dependent Obj	Dependent Obj Type
vVendor	VIEW

### VendorAddress

#### Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
AddressID	int			NO	Primary key. Foreign key to Address.AddressID.
AddressID	int			NO	Nonclustered index.
AddressTypeID	int			NO	Address type. Foreign key to AddressType.AddressTypeID.
ModifiedDate	datetime		(getdate())	NO	Date and time the record was last updated.
VendorID	int			NO	Primary key. Foreign key to Vendor.VendorID.
VendorID	int			NO	Clustered index created by a primary key constraint.

#### Table Indexes

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
VendorAddresses	PK_VendorAddress_VendorID_AddressID	VendorID	CLUSTERED	true	true	false	false	false	false
VendorAddresses	PK_VendorAddress_VendorID_AddressID	AddressID	CLUSTERED	true	true	false	false	false	false
VendorAddresses	IX_VendorAddress_AddressID	AddressID	NONCLUSTERED	false	false	false	false	false	false

#### Dependent Objects

Dependent Obj	Dependent Obj Type
vVendor	VIEW

## VendorContact

#### Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
ContactID	int			NO	Contact (Vendor employee) identification number. Foreign key to Contact.ContactID.
ContactID	int			NO	Nonclustered index.
ContactTypeID	int			NO	Contact type such as sales manager, or sales agent.
ContactTypeID	int			NO	Nonclustered index.
ModifiedDate	datetime		(getdate())	NO	Date and time the record was last updated.
VendorID	int			NO	Primary key.
VendorID	int			NO	Clustered index created by a primary key constraint.

## Table Indexes

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
VendorContact	PK_VendorContact_VendorID_ContactID	VendorID	CLUSTERED	true	true	false	false	false	false
VendorContact	PK_VendorContact_VendorID_ContactID	ContactID	CLUSTERED	true	true	false	false	false	false
VendorContact	IX_VendorContact_ContactID	ContactID	NONCLUSTERED	false	false	false	false	false	false
VendorContact	IX_VendorContact_ContactTypeID	ContactTypeID	NONCLUSTERED	false	false	false	false	false	false

## Dependent Objects

Dependent Obj	Dependent Obj Type
ufnGetContactInformation	SQL_TABLE_VALUED_FUNCTION
vVendor	VIEW

## ContactCreditCard

### Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
ContactID	int			NO	Customer identification number. Foreign key to Contact.ContactID.
ContactID	int			NO	Clustered index created by a primary key constraint.

Column Name	Data Type	Length	Column Default	Is Nullable	Description
CreditCardID	int			NO	Credit card identification number. Foreign key to CreditCard.CreditCardID.
ModifiedDate	datetime		(getdate())	NO	Date and time the record was last updated.

#### Table Indexes

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
ContactCreditCard	PK_ContactCreditCard_ContactID_CreditCardID	ContactID	CLUSTERED	true	true	false	false	false	false
ContactCreditCard	PK_ContactCreditCard_ContactID_CreditCardID	CreditCardID	CLUSTERED	true	true	false	false	false	false

#### Dependent Objects

---None---

## CountryRegionCurrency

#### Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
CountryRegionCode	nvarchar	3		NO	ISO code for countries and regions. Foreign key to CountryRegion.CountryRegionCode.
CountryRegionCode	nvarchar	3		NO	Clustered index created by a primary key constraint.
CurrencyCode	nchar	3		NO	ISO standard currency code. Foreign key to Currency.CurrencyCode.



Column Name	Data Type	Length	Column Default	Is Nullable	Description
CurrencyCode	nchar	3		NO	Nonclustered index.
ModifiedDate	datetime		(getdate())	NO	Date and time the record was last updated.

#### Table Indexes

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
CountryRegionCurrency	PK_CountryRegionCurrency_CountryRegionCode_CurrencyCode	CountryRegionCode	CLUSTERED	true	true	false	false	false	false
CountryRegionCurrency	PK_CountryRegionCurrency_CountryRegionCode_CurrencyCode	CurrencyCode	CLUSTERED	true	true	false	false	false	false
CountryRegionCurrency	IX_CountryRegionCurrency_CurrencyCode	CurrencyCode	NONCLUSTERED	false	false	false	false	false	false

#### Dependent Objects

---None---

### CreditCard

#### Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
CardNumber	nvarchar	25		NO	Credit card number.
CardType	nvarchar	50		NO	Credit card name.

Column Name	Data Type	Length	Column Default	Is Nullable	Description
CardType	nvarchar	50		NO	Unique nonclustered index.
CreditCardID	int			NO	Primary key for CreditCard records.
CreditCardID	int			NO	Clustered index created by a primary key constraint.
ExpMonth	tinyint			NO	Credit card expiration month.
ExpYear	smallint			NO	Credit card expiration year.
ModifiedDate	datetime		(getdate())	NO	Date and time the record was last updated.

#### Table Indexes

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
CreditCard	PK_CreditCard_CreditCardID	CreditCardID	CLUSTERED	true	true	false	false	false	false
CreditCard	AK_CreditCard_CardNumber	CardNumber	NONCLUSTERED	true	false	false	false	false	false

#### Dependent Objects

---None---

## Currency

Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
CurrencyCode	nchar	3		NO	The ISO code for the Currency.
CurrencyCode	nchar	3		NO	Clustered index created by a primary key constraint.
ModifiedDate	datetime		(getdate())	NO	Date and time the record was last updated.
Name	nvarchar	50		NO	Currency name.
Name	nvarchar	50		NO	Unique nonclustered index.

#### Table Indexes

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
Currency	PK_Currency_CurrencyCode	CurrencyCode	CLUSTERED	true	true	false	false	false	false
Currency	AK_Currency_Name	Name	NONCLUSTERED	true	false	false	false	false	false

#### Dependent Objects

---None---

## CurrencyRate

#### Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
AverageRate	money			NO	Average exchange rate for the day.

Column Name	Data Type	Length	Column Default	Is Nullable	Description
CurrencyRateDate	datetime			NO	Date and time the exchange rate was obtained.
CurrencyRateDate	datetime			NO	Unique nonclustered index.
CurrencyRateID	int			NO	Primary key for CurrencyRate records.
CurrencyRateID	int			NO	Clustered index created by a primary key constraint.
EndOfDayRate	money			NO	Final exchange rate for the day.
FromCurrencyCode	nchar	3		NO	Exchange rate was converted from this currency code.
ModifiedDate	datetime		(getdate())	NO	Date and time the record was last updated.
ToCurrencyCode	nchar	3		NO	Exchange rate was converted to this currency code.

#### Table Indexes

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
CurrencyRate	PK_CurrencyRate_CurrencyRateID	CurrencyRateID	CLUSTERED	true	true	false	false	false	false
CurrencyRate	AK_CurrencyRate_CurrencyRateDate_FromCurrencyCode_ToCurrencyCode	CurrencyRateDate	NONCLUSTERED	true	false	false	false	false	false
CurrencyRate	AK_CurrencyRate_CurrencyRateDate_FromCurrencyCode_ToCurrencyCode	FromCurrencyCode	NONCLUSTERED	true	false	false	false	false	false
CurrencyRate	AK_CurrencyRate_CurrencyRateDate_FromCurrencyCode_ToCurrencyCode	ToCurrencyCode	NONCLUSTERED	true	false	false	false	false	false

## Dependent Objects

---None---

## Customer

### Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
AccountNumber	varchar	10		NO	Unique number identifying the customer assigned by the accounting system.
AccountNumber	varchar	10		NO	Unique nonclustered index.
CustomerID	int			NO	Primary key for Customer records.
CustomerID	int			NO	Clustered index created by a primary key constraint.
CustomerType	nchar	1		NO	Customer type: I = Individual, S = Store
ModifiedDate	datetime		(getdate())	NO	Date and time the record was last updated.
rowguid	uniqueidentifier		(newid())	NO	ROWGUIDCOL number uniquely identifying the record. Used to support a merge replication sample.
rowguid	uniqueidentifier		(newid())	NO	Nonclustered index.
TerritoryID	int			YES	ID of the territory in which the customer is located. Foreign key to SalesTerritory.SalesTerritoryID.
TerritoryID	int			YES	Unique nonclustered index. Used to support replication samples.

## Table Indexes

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
Customer	PK_Customer_CustomerID	CustomerID	CLUSTERED	true	true	false	false	false	false
Customer	AK_Customer_rowguid	rowguid	NONCLUSTERED	true	false	false	false	false	false
Customer	AK_Customer_AccountNumber	AccountNumber	NONCLUSTERED	true	false	false	false	false	false
Customer	IX_Customer_TerritoryID	TerritoryID	NONCLUSTERED	false	false	false	false	false	false

## Dependent Objects

Dependent Obj	Dependent Obj Type
ufnLeadingZeros	SQL_SCALAR_FUNCTION
Customer	USER_TABLE
vIndividualCustomer	VIEW
vStoreWithDemographics	VIEW

## CustomerAddress

### Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
AddressID	int			NO	Primary key. Foreign key to Address.AddressID.
AddressID	int			NO	Unique nonclustered index. Used to support replication samples.

Column Name	Data Type	Length	Column Default	Is Nullable	Description
AddressTypeID	int			NO	Address type. Foreign key to AddressType.AddressTypeID.
CustomerID	int			NO	Primary key. Foreign key to Customer.CustomerID.
CustomerID	int			NO	Clustered index created by a primary key constraint.
ModifiedDate	datetime		(getdate())	NO	Date and time the record was last updated.
rowguid	uniqueidentifier		(newid())	NO	ROWGUIDCOL number uniquely identifying the record. Used to support a merge replication sample.

#### Table Indexes

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
CustomerAddress	PK_CustomerAddress_CustomerID_AddressID	CustomerID	CLUSTERED	true	true	false	false	false	false
CustomerAddress	PK_CustomerAddress_CustomerID_AddressID	AddressID	CLUSTERED	true	true	false	false	false	false
CustomerAddress	AK_CustomerAddress_rowguid	rowguid	NONCLUSTERED	true	false	false	false	false	false

#### Dependent Objects

Dependent Obj	Dependent Obj Type
vIndividualCustomer	VIEW
vStoreWithDemographics	VIEW





## Dependent Objects

Dependent Obj	Dependent Obj Type
ufnGetContactInformation	SQL_TABLE_VALUED_FUNCTION
iduSalesOrderDetail	SQL_TRIGGER
iStore	SQL_TRIGGER
iuIndividual	SQL_TRIGGER
vIndividualCustomer	VIEW
vIndividualDemographics	VIEW

## SalesOrderDetail

### Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
CarrierTrackingNumber	nvarchar	25		YES	Shipment tracking number supplied by the shipper.
CarrierTrackingNumber	nvarchar	25		YES	Nonclustered index.
LineTotal	numeric			NO	Per product subtotal. Computed as UnitPrice * (1 - UnitPriceDiscount) * OrderQty.
ModifiedDate	datetime		(getdate())	NO	Date and time the record was last updated.
OrderQty	smallint			NO	Quantity ordered per product.
ProductID	int			NO	Product sold to customer. Foreign key to Product.ProductID.
rowguid	uniqueidentifier		(newid())	NO	ROWGUIDCOL number uniquely identifying the record. Used to support a merge replication sample.



## Dependent Objects

Dependent Obj	Dependent Obj Type
iduSalesOrderDetail	SQL_TRIGGER
SalesOrderDetail	USER_TABLE

## SalesOrderHeader

### Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
AccountNumber	nvarchar	15		YES	Financial accounting number reference.
BillToAddressID	int			NO	Customer billing address. Foreign key to Address.AddressID.
Comment	nvarchar	128		YES	Sales representative comments.
ContactID	int			NO	Customer contact identification number. Foreign key to Contact.ContactID.
CreditCardApprovalCode	varchar	15		YES	Approval code provided by the credit card company.
CreditCardID	int			YES	Credit card identification number. Foreign key to CreditCard.CreditCardID.
CurrencyRateID	int			YES	Currency exchange rate used. Foreign key to CurrencyRate.CurrencyRateID.
CustomerID	int			NO	Customer identification number. Foreign key to Customer.CustomerID.
DueDate	datetime			NO	Date the order is due to the customer.
Freight	money		((0.00))	NO	Shipping cost.
ModifiedDate	datetime		(getdate())	NO	Date and time the record was last updated.
OnlineOrderFlag	bit		((1))	NO	0 = Order placed by sales person. 1 = Order placed online by customer.
OrderDate	datetime		(getdate())	NO	Dates the sales order was created.
OrderDate	datetime		(getdate())	NO	Unique nonclustered index.

Column Name	Data Type	Length	Column Default	Is Nullable	Description
PurchaseOrderNumber	nvarchar	25		YES	Customer purchase order number reference.
RevisionNumber	tinyint		((0))	NO	Incremental number to track changes to the sales order over time.
RevisionNumber	tinyint		((0))	NO	Unique nonclustered index. Used to support replication samples.
rowguid	uniqueidentifier		(newid())	NO	ROWGUIDCOL number uniquely identifying the record. Used to support a merge replication sample.
SalesOrderID	int			NO	Primary key.
SalesOrderID	int			NO	Clustered index created by a primary key constraint.
SalesOrderNumber	nvarchar	25		NO	Unique sales order identification number.
SalesPersonID	int			YES	Sales person who created the sales order. Foreign key to SalesPerson.SalePersonID.
ShipDate	datetime			YES	Date the order was shipped to the customer.
ShipDate	datetime			YES	Nonclustered index.
ShipMethodID	int			NO	Shipping method. Foreign key to ShipMethod.ShipMethodID.
ShipToAddressID	int			NO	Customer shipping address. Foreign key to Address.AddressID.
Status	tinyint		((1))	NO	Order current status. 1 = In process; 2 = Approved; 3 = Backordered; 4 = Rejected; 5 = Shipped; 6 = Cancelled
Status	tinyint		((1))	NO	Nonclustered index.
SubTotal	money		((0.00))	NO	Sales subtotal. Computed as SUM(SalesOrderDetail.LineTotal)for the appropriate SalesOrderID.
TaxAmt	money		((0.00))	NO	Tax amount.
TerritoryID	int			YES	Territory in which the sale was made. Foreign key to SalesTerritory.SalesTerritoryID.
TotalDue	money			NO	Total due from customer. Computed as Subtotal + TaxAmt + Freight.

## Table Indexes

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
SalesOrderHeader	PK_SalesOrderHeader_SalesOrderID	SalesOrderID	CLUSTERED	true	true	false	false	false	false
SalesOrderHeader	AK_SalesOrderHeader_rowguid	rowguid	NONCLUSTERED	true	false	false	false	false	false
SalesOrderHeader	AK_SalesOrderHeader_SalesOrderNumber	SalesOrderNumber	NONCLUSTERED	true	false	false	false	false	false
SalesOrderHeader	IX_SalesOrderHeader_CustomerID	CustomerID	NONCLUSTERED	false	false	false	false	false	false
SalesOrderHeader	IX_SalesOrderHeader_SalesPersonID	SalesPersonID	NONCLUSTERED	false	false	false	false	false	false

## Dependent Objects

Dependent Obj	Dependent Obj Type
iduSalesOrderDetail	SQL_TRIGGER
uSalesOrderHeader	SQL_TRIGGER
SalesOrderHeader	USER_TABLE
vSalesPersonSalesByFiscalYears	VIEW

## SalesOrderHeaderSalesReason

### Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
ModifiedDate	datetime		(getdate())	NO	Date and time the record was last updated.
SalesOrderID	int			NO	Primary key. Foreign key to SalesOrderHeader.SalesOrderID.
SalesOrderID	int			NO	Clustered index created by a primary key constraint.
SalesReasonID	int			NO	Primary key. Foreign key to SalesReason.SalesReasonID.

#### Table Indexes

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
SalesOrderHeaderSalesReason	PK_SalesOrderHeaderSalesReason_SalesOrderID_SalesReasonID	SalesOrderID	CLUSTERED	true	true	false	false	false	false
SalesOrderHeaderSalesReason	PK_SalesOrderHeaderSalesReason_SalesReasonID_SalesReasonID	SalesReasonID	CLUSTERED	true	true	false	false	false	false

#### Dependent Objects

---None---

## SalesPerson

Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
Bonus	money		((0.00))	NO	Bonus due if quota is met.
CommissionPct	smallmoney		((0.00))	NO	Commision percent received per sale.
ModifiedDate	datetime		(getdate())	NO	Date and time the record was last updated.
rowguid	uniqueidentifier		(newid())	NO	ROWGUIDCOL number uniquely identifying the record. Used to support a merge replication sample.
SalesLastYear	money		((0.00))	NO	Sales total of previous year.
SalesPersonID	int			NO	Primary key for SalesPerson records.
SalesPersonID	int			NO	Clustered index created by a primary key constraint.
SalesQuota	money			YES	Projected yearly sales.
SalesYTD	money		((0.00))	NO	Sales total year to date.
TerritoryID	int			YES	Territory currently assigned to. Foreign key to SalesTerritory.SalesTerritoryID.
TerritoryID	int			YES	Unique nonclustered index. Used to support replication samples.

#### Table Indexes

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
SalesPerson	PK_SalesPerson_SalesPersonID	SalesPersonID	CLUSTERED	true	true	false	false	false	false
SalesPerson	AK_SalesPerson_rowguid	rowguid	NONCLUSTERED	true	false	false	false	false	false

## Dependent Objects

Dependent Obj	Dependent Obj Type
uSalesOrderHeader	SQL_TRIGGER
vSalesPerson	VIEW
vSalesPersonSalesByFiscalYears	VIEW

## SalesPersonQuotaHistory

### Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
ModifiedDate	datetime		(getdate())	NO	Date and time the record was last updated.
QuotaDate	datetime			NO	Sales quota date.
QuotaDate	datetime			NO	Unique nonclustered index. Used to support replication samples.
rowguid	uniqueidentifier		(newid())	NO	ROWGUIDCOL number uniquely identifying the record. Used to support a merge replication sample.
SalesPersonID	int			NO	Sales person identification number. Foreign key to SalesPerson.SalesPersonID.
SalesPersonID	int			NO	Clustered index created by a primary key constraint.
SalesQuota	money			NO	Sales quota amount.

### Table Indexes



Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
SalesPersonQuotaHistory	PK_SalesPersonQuotaHistory_SalesPersonID_QuotaDate	SalesPersonID	CLUSTERED	true	true	false	false	false	false
SalesPersonQuotaHistory	PK_SalesPersonQuotaHistory_SalesPersonID_QuotaDate	QuotaDate	CLUSTERED	true	true	false	false	false	false
SalesPersonQuotaHistory	AK_SalesPersonQuotaHistory_rowguid	rowguid	NONCLUSTERED	true	false	false	false	false	false

#### Dependent Objects

---None---

## SalesReason

#### Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
ModifiedDate	datetime		(getdate())	NO	Date and time the record was last updated.
Name	nvarchar	50		NO	Sales reason description.
ReasonType	nvarchar	50		NO	Category the sales reason belongs to.
SalesReasonID	int			NO	Primary key for SalesReason records.
SalesReasonID	int			NO	Clustered index created by a primary key constraint.

## Table Indexes

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
SalesReason	PK_SalesReason_SalesReasonID	SalesReasonID	CLUSTERED	true	true	false	false	false	false

## Dependent Objects

---None---

## SalesTaxRate

### Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
ModifiedDate	datetime		(getdate())	NO	Date and time the record was last updated.
Name	nvarchar	50		NO	Tax rate description.
rowguid	uniqueidentifier		(newid())	NO	ROWGUIDCOL number uniquely identifying the record. Used to support a merge replication sample.
SalesTaxRateID	int			NO	Primary key for SalesTaxRate records.
SalesTaxRateID	int			NO	Clustered index created by a primary key constraint.
StateProvinceID	int			NO	State, province, or country/region the sales tax applies to.
StateProvinceID	int			NO	Unique nonclustered index.
TaxRate	smallmoney		((0.00))	NO	Tax rate amount.
TaxType	tinyint			NO	1 = Tax applied to retail transactions, 2 = Tax applied to wholesale transactions, 3 = Tax applied to all sales (retail and wholesale) transactions.

Column Name	Data Type	Length	Column Default	Is Nullable	Description
TaxType	tinyint			NO	Unique nonclustered index. Used to support replication samples.

#### Table Indexes

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
SalesTaxRate	PK_SalesTaxRate_SalesTaxRateID	SalesTaxRateID	CLUSTERED	true	true	false	false	false	false
SalesTaxRate	AK_SalesTaxRate_StateProvinceID_TaxType	StateProvinceID	NONCLUSTERED	true	false	false	false	false	false
SalesTaxRate	AK_SalesTaxRate_StateProvinceID_TaxType	TaxType	NONCLUSTERED	true	false	false	false	false	false
SalesTaxRate	AK_SalesTaxRate_rowguid	rowguid	NONCLUSTERED	true	false	false	false	false	false

#### Dependent Objects

---None---

### SalesTerritory

#### Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
CostLastYear	money		((0.00))	NO	Business costs in the territory the previous year.
CostYTD	money		((0.00))	NO	Business costs in the territory year to date.

Column Name	Data Type	Length	Column Default	Is Nullable	Description
CountryRegionCode	nvarchar	3		NO	ISO standard country or region code. Foreign key to CountryRegion.CountryRegionCode.
CountryRegionCode	nvarchar	3		NO	Unique nonclustered index. Used to support replication samples.
Group	nvarchar	50		NO	Geographic area to which the sales territory belong.
ModifiedDate	datetime		(getdate())	NO	Date and time the record was last updated.
Name	nvarchar	50		NO	Sales territory description
Name	nvarchar	50		NO	Unique nonclustered index.
rowguid	uniqueidentifier		(newid())	NO	ROWGUIDCOL number uniquely identifying the record. Used to support a merge replication sample.
SalesLastYear	money		((0.00))	NO	Sales in the territory the previous year.
SalesYTD	money		((0.00))	NO	Sales in the territory year to date.
TerritoryID	int			NO	Primary key for SalesTerritory records.
TerritoryID	int			NO	Clustered index created by a primary key constraint.

#### Table Indexes

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
SalesTerritory	PK_SalesTerritory_TerritoryID	TerritoryID	CLUSTERED	true	true	false	false	false	false
SalesTerritory	AK_SalesTerritory_Name	Name	NONCLUSTERED	true	false	false	false	false	false
SalesTerritory	AK_SalesTerritory_rowguid	rowguid	NONCLUSTERED	true	false	false	false	false	false

## Dependent Objects

Dependent Obj	Dependent Obj Type
uSalesOrderHeader	SQL_TRIGGER
vSalesPerson	VIEW
vSalesPersonSalesByFiscalYears	VIEW

## SalesTerritoryHistory

### Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
EndDate	datetime			YES	Date the sales representative left work in the territory.
ModifiedDate	datetime		(getdate())	NO	Date and time the record was last updated.
rowguid	uniqueidentifier		(newid())	NO	ROWGUIDCOL number uniquely identifying the record. Used to support a merge replication sample.
SalesPersonID	int			NO	Primary key for SalesTerritoryHistory records.
SalesPersonID	int			NO	Clustered index created by a primary key constraint.
StartDate	datetime			NO	Date the sales representative started work in the territory.
TerritoryID	int			NO	Territory identification number. Foreign key to SalesTerritory.SalesTerritoryID.
TerritoryID	int			NO	Unique nonclustered index. Used to support replication samples.

## Table Indexes

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
SalesTerritoryHistory	PK_SalesTerritoryHistory_SalesPersonID_StartDate_TerritoryID	SalesPersonID	CLUSTERED	true	true	false	false	false	false
SalesTerritoryHistory	PK_SalesTerritoryHistory_SalesPersonID_StartDate_TerritoryID	TerritoryID	CLUSTERED	true	true	false	false	false	false
SalesTerritoryHistory	PK_SalesTerritoryHistory_SalesPersonID_StartDate_TerritoryID	StartDate	CLUSTERED	true	true	false	false	false	false
SalesTerritoryHistory	AK_SalesTerritoryHistory_rowguid	rowguid	NONCLUSTERED	true	false	false	false	false	false

## Dependent Objects

---None---

## ShoppingCartItem

### Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
DateCreated	datetime		(getdate())	NO	Date the time the record was created.
ModifiedDate	datetime		(getdate())	NO	Date and time the record was last updated.
ProductID	int			NO	Product ordered. Foreign key to Product.ProductID.
Quantity	int		((1))	NO	Product quantity ordered.
ShoppingCartID	nvarchar	50		NO	Shopping cart identification number.
ShoppingCartID	nvarchar	50		NO	Nonclustered index.

Column Name	Data Type	Length	Column Default	Is Nullable	Description
ShoppingCartItemID	int			NO	Primary key for ShoppingCartItem records.
ShoppingCartItemID	int			NO	Clustered index created by a primary key constraint.

#### Table Indexes

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
ShoppingCartItem	PK_ShoppingCartItem_ShopppingCartItemID	ShoppingCartItemID	CLUSTERED	true	true	false	false	false	false
ShoppingCartItem	IX_ShoppingCartItem_ShopppingCartItemID_ProductID	ShoppingCartItemID	NONCLUSTERED	false	false	false	false	false	false
ShoppingCartItem	IX_ShoppingCartItem_ShopppingCartItemID_ProductID	ProductID	NONCLUSTERED	false	false	false	false	false	false

#### Dependent Objects

---None---

### SpecialOffer

#### Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
Category	nvarchar	50		NO	Group the discount applies to such as Reseller or Customer.

Column Name	Data Type	Length	Column Default	Is Nullable	Description
Description	nvarchar	255		NO	Discount description.
Description	nvarchar	255		NO	Unique nonclustered index. Used to support replication samples.
DiscountPct	smallmoney		((0.00))	NO	Discount percentage.
EndDate	datetime			NO	Discount end date.
MaxQty	int			YES	Maximum discount percent allowed.
MinQty	int		((0))	NO	Minimum discount percent allowed.
ModifiedDate	datetime		(getdate())	NO	Date and time the record was last updated.
rowguid	uniqueidentifier		(newid())	NO	ROWGUIDCOL number uniquely identifying the record. Used to support a merge replication sample.
SpecialOfferID	int			NO	Primary key for SpecialOffer records.
SpecialOfferID	int			NO	Clustered index created by a primary key constraint.
StartDate	datetime			NO	Discount start date.
Type	nvarchar	50		NO	Discount type category.

#### Table Indexes

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
SpecialOffer	PK_SpecialOffer_SpecialOfferID	SpecialOfferID	CLUSTERED	true	true	false	false	false	false
SpecialOffer	AK_SpecialOffer_rowguid	rowguid	NONCLUSTERED	true	false	false	false	false	false



## Dependent Objects

---None---

## SpecialOfferProduct

### Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
ModifiedDate	datetime		(getdate())	NO	Date and time the record was last updated.
ProductID	int			NO	Product identification number. Foreign key to Product.ProductID.
ProductID	int			NO	Unique nonclustered index. Used to support replication samples.
rowguid	uniqueidentifier		(newid())	NO	ROWGUIDCOL number uniquely identifying the record. Used to support a merge replication sample.
rowguid	uniqueidentifier		(newid())	NO	Nonclustered index.
SpecialOfferID	int			NO	Primary key for SpecialOfferProduct records.
SpecialOfferID	int			NO	Clustered index created by a primary key constraint.

### Table Indexes

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
SpecialOfferProduct	PK_SpecialOfferProduct_SpecialOfferID_ProductID	SpecialOfferID	CLUSTERED	true	true	false	false	false	false
SpecialOfferProduct	PK_SpecialOfferProduct_SpecialOfferID_ProductID	ProductID	CLUSTERED	true	true	false	false	false	false

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
SpecialOfferProduct	AK_SpecialOfferProduct_rowguid	rowguid	NONCLUSTERED	true	false	false	false	false	false
SpecialOfferProduct	IX_SpecialOfferProduct_ProductID	ProductID	NONCLUSTERED	false	false	false	false	false	false

### Dependent Objects

---None---

## Store

### Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
CustomerID	int			NO	Primary key. Foreign key to Customer.CustomerID.
CustomerID	int			NO	Clustered index created by a primary key constraint.
Demographics	xml	-1		YES	Demographic information about the store such as the number of employees, annual sales and store type.
ModifiedDate	datetime		(getdate())	NO	Date and time the record was last updated.
Name	nvarchar	50		NO	Name of the store.
Name	nvarchar	50		NO	Unique nonclustered index. Used to support replication samples.
rowguid	uniqueidentifier		(newid())	NO	ROWGUIDCOL number uniquely identifying the record. Used to support a merge replication sample.

Column Name	Data Type	Length	Column Default	Is Nullable	Description
SalesPersonID	int			YES	ID of the sales person assigned to the customer. Foreign key to SalesPerson.SalesPersonID.
SalesPersonID	int			YES	Nonclustered index.

#### Table Indexes

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
Store	PK_Store_CustomerID	CustomerID	CLUSTERED	true	true	false	false	false	false
Store	AK_Store_rowguid	rowguid	NONCLUSTERED	true	false	false	false	false	false
Store	IX_Store_SalesPersonID	SalesPersonID	NONCLUSTERED	false	false	false	false	false	false
Store	PXML_Store_Demographics	Demographics	XML	false	false	false	false	false	false

#### Dependent Objects

Dependent Obj	Dependent Obj Type
iuIndividual	SQL_TRIGGER
vStoreWithDemographics	VIEW

## StoreContact

### Column Details



Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
StoreContact	IX_StoreContact_ContactTypeID	ContactTypeID	NONCLUSTERED	false	false	false	false	false	false

### Dependent Objects

Dependent Obj	Dependent Obj Type
ufnGetContactInformation	SQL_TABLE_VALUED_FUNCTION
vStoreWithDemographics	VIEW

### Views

View Name	View Schema
vEmployee	HumanResources
vEmployeeDepartment	HumanResources
vEmployeeDepartmentHistory	HumanResources
vJobCandidate	HumanResources
vJobCandidateEducation	HumanResources
vJobCandidateEmployment	HumanResources
vAdditionalContactInfo	Person
vStateProvinceCountryRegion	Person
vProductAndDescription	Production
vProductModelCatalogDescription	Production
vProductModelInstructions	Production

View Name	View Schema
vVendor	Purchasing
vIndividualCustomer	Sales
vIndividualDemographics	Sales
vSalesPerson	Sales
vSalesPersonSalesByFiscalYears	Sales
vStoreWithDemographics	Sales

## vEmployee

### Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
AdditionalContactInfo	xml	-1		YES	
AddressLine1	nvarchar	60		NO	
AddressLine2	nvarchar	60		YES	
City	nvarchar	30		NO	
CountryRegionName	nvarchar	50		NO	
EmailAddress	nvarchar	50		YES	
EmailPromotion	int			NO	
EmployeeID	int			NO	
FirstName	nvarchar	50		NO	
JobTitle	nvarchar	50		NO	
LastName	nvarchar	50		NO	
MiddleName	nvarchar	50		YES	
Phone	nvarchar	25		YES	

Column Name	Data Type	Length	Column Default	Is Nullable	Description
PostalCode	nvarchar	15		NO	
StateProvinceName	nvarchar	50		NO	
Suffix	nvarchar	10		YES	
Title	nvarchar	8		YES	

[View Definition](#)

## View Definition

```
CREATE VIEW [HumanResources].[vEmployee]
AS
SELECT
    e.[EmployeeID]
    ,c.[Title]
    ,c.[FirstName]
    ,c.[MiddleName]
    ,c.[LastName]
    ,c.[Suffix]
    ,e.[Title] AS [JobTitle]
    ,c.[Phone]
    ,c.[EmailAddress]
    ,c.[EmailPromotion]
    ,a.[AddressLine1]
    ,a.[AddressLine2]
    ,a.[City]
    ,sp.[Name] AS [StateProvinceName]
    ,a.[PostalCode]
    ,cr.[Name] AS [CountryRegionName]
    ,c.[AdditionalContactInfo]
FROM [HumanResources].[Employee] e
INNER JOIN [Person].[Contact] c
ON c.[ContactID] = e.[ContactID]
INNER JOIN [HumanResources].[EmployeeAddress] ea
ON e.[EmployeeID] = ea.[EmployeeID]
INNER JOIN [Person].[Address] a
ON ea.[AddressID] = a.[AddressID]
INNER JOIN [Person].[StateProvince] sp
ON sp.[StateProvinceID] = a.[StateProvinceID]
INNER JOIN [Person].[CountryRegion] cr
ON cr.[CountryRegionCode] = sp.[CountryRegionCode];
```

### Tables Used in View

Table Catalog	Table Schema	Table Name
AdventureWorks	HumanResources	Employee



Table Catalog	Table Schema	Table Name
AdventureWorks	HumanResources	EmployeeAddress
AdventureWorks	Person	Address
AdventureWorks	Person	Contact
AdventureWorks	Person	CountryRegion
AdventureWorks	Person	StateProvince

#### Dependent Objects

Dependent Obj	Dependent Obj Type
Address	USER_TABLE
Contact	USER_TABLE
CountryRegion	USER_TABLE
Employee	USER_TABLE
EmployeeAddress	USER_TABLE
StateProvince	USER_TABLE

### vEmployeeDepartment

#### Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
Department	nvarchar	50		NO	
EmployeeID	int			NO	
FirstName	nvarchar	50		NO	
GroupName	nvarchar	50		NO	

Column Name	Data Type	Length	Column Default	Is Nullable	Description
JobTitle	nvarchar	50		NO	
LastName	nvarchar	50		NO	
MiddleName	nvarchar	50		YES	
StartDate	datetime			NO	
Suffix	nvarchar	10		YES	
Title	nvarchar	8		YES	

#### View Definition

#### View Definition

```

CREATE VIEW [HumanResources].[vEmployeeDepartment]
AS
SELECT
    e.[EmployeeID]
    ,c.[Title]
    ,c.[FirstName]
    ,c.[MiddleName]
    ,c.[LastName]
    ,c.[Suffix]
    ,e.[Title] AS [JobTitle]
    ,d.[Name] AS [Department]
    ,d.[GroupName]
    ,edh.[StartDate]
FROM [HumanResources].[Employee] e
    INNER JOIN [Person].[Contact] c
        ON c.[ContactID] = e.[ContactID]
    INNER JOIN [HumanResources].[EmployeeDepartmentHistory] edh
        ON e.[EmployeeID] = edh.[EmployeeID]
    INNER JOIN [HumanResources].[Department] d
        ON edh.[DepartmentID] = d.[DepartmentID]
WHERE GETDATE() BETWEEN edh.[StartDate] AND ISNULL(edh.[EndDate], GETDATE());

```

### Tables Used in View

Table Catalog	Table Schema	Table Name
AdventureWorks	HumanResources	Department
AdventureWorks	HumanResources	Employee
AdventureWorks	HumanResources	EmployeeDepartmentHistory
AdventureWorks	Person	Contact

### Dependent Objects

Dependent Obj	Dependent Obj Type
Contact	USER_TABLE
Department	USER_TABLE
Employee	USER_TABLE
EmployeeDepartmentHistory	USER_TABLE

### vEmployeeDepartmentHistory

#### Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
Department	nvarchar	50		NO	
EmployeeID	int			NO	
EndDate	datetime			YES	

Column Name	Data Type	Length	Column Default	Is Nullable	Description
FirstName	nvarchar	50		NO	
GroupName	nvarchar	50		NO	
LastName	nvarchar	50		NO	
MiddleName	nvarchar	50		YES	
Shift	nvarchar	50		NO	
StartDate	datetime			NO	
Suffix	nvarchar	10		YES	
Title	nvarchar	8		YES	

[View Definition](#)

## View Definition

```

CREATE VIEW [HumanResources].[vEmployeeDepartmentHistory]
AS
SELECT
    e.[EmployeeID]
    ,c.[Title]
    ,c.[FirstName]
    ,c.[MiddleName]
    ,c.[LastName]
    ,c.[Suffix]
    ,s.[Name] AS [Shift]
    ,d.[Name] AS [Department]
    ,d.[GroupName]
    ,edh.[StartDate]
    ,edh.[EndDate]
FROM [HumanResources].[Employee] e
INNER JOIN [Person].[Contact] c
ON c.[ContactID] = e.[ContactID]
INNER JOIN [HumanResources].[EmployeeDepartmentHistory] edh
ON e.[EmployeeID] = edh.[EmployeeID]
INNER JOIN [HumanResources].[Department] d
ON edh.[DepartmentID] = d.[DepartmentID]
INNER JOIN [HumanResources].[Shift] s
ON s.[ShiftID] = edh.[ShiftID];

```

### Tables Used in View

Table Catalog	Table Schema	Table Name
AdventureWorks	HumanResources	Department
AdventureWorks	HumanResources	Employee
AdventureWorks	HumanResources	EmployeeDepartmentHistory
AdventureWorks	HumanResources	Shift
AdventureWorks	Person	Contact

## Dependent Objects

Dependent Obj	Dependent Obj Type
Contact	USER_TABLE
Department	USER_TABLE
Employee	USER_TABLE
EmployeeDepartmentHistory	USER_TABLE
Shift	USER_TABLE

## vJobCandidate

### Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
Addr.Loc.City	nvarchar	100		YES	
Addr.Loc.CountryRegion	nvarchar	100		YES	
Addr.Loc.State	nvarchar	100		YES	
Addr.PostalCode	nvarchar	20		YES	
Addr.Type	nvarchar	30		YES	
EMail	nvarchar	-1		YES	
EmployeeID	int			YES	
JobCandidateID	int			NO	
ModifiedDate	datetime			NO	
Name.First	nvarchar	30		YES	
Name.Last	nvarchar	30		YES	

Column Name	Data Type	Length	Column Default	Is Nullable	Description
Name.Middle	nvarchar	30		YES	
Name.Prefix	nvarchar	30		YES	
Name.Suffix	nvarchar	30		YES	
Skills	nvarchar	-1		YES	
WebSite	nvarchar	-1		YES	

[View Definition](#)

## View Definition

```
CREATE VIEW [HumanResources].[vJobCandidate]
AS
SELECT
    jc.[JobCandidateID]
    ,jc.[EmployeeID]
    ,[Resume].ref.value(N'declare default element namespace "http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/Resume";
    (/Resume/Name/Name.Prefix)[1]', 'nvarchar(30)') AS [Name.Prefix]
    ,[Resume].ref.value(N'declare default element namespace "http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/Resume";
    (/Resume/Name/Name.First)[1]', 'nvarchar(30)') AS [Name.First]
    ,[Resume].ref.value(N'declare default element namespace "http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/Resume";
    (/Resume/Name/Name.Middle)[1]', 'nvarchar(30)') AS [Name.Middle]
    ,[Resume].ref.value(N'declare default element namespace "http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/Resume";
    (/Resume/Name/Name.Last)[1]', 'nvarchar(30)') AS [Name.Last]
    ,[Resume].ref.value(N'declare default element namespace "http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/Resume";
    (/Resume/Name/Name.Suffix)[1]', 'nvarchar(30)') AS [Name.Suffix]
    ,[Resume].ref.value(N'declare default element namespace "http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/Resume";
    (/Resume/Skills)[1]', 'nvarchar(max)') AS [Skills]
    ,[Resume].ref.value(N'declare default element namespace "http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/Resume";
    (Address/Addr.Type)[1]', 'nvarchar(30)') AS [Addr.Type]
    ,[Resume].ref.value(N'declare default element namespace "http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/Resume";
    (Address/Addr.Location/Location/Loc.CountryRegion)[1]', 'nvarchar(100)') AS [Addr.Loc.CountryRegion]
    ,[Resume].ref.value(N'declare default element namespace "http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/Resume";
    (Address/Addr.Location/Location/Loc.State)[1]', 'nvarchar(100)') AS [Addr.Loc.State]
    ,[Resume].ref.value(N'declare default element namespace "http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/Resume";
    (Address/Addr.Location/Location/Loc.City)[1]', 'nvarchar(100)') AS [Addr.Loc.City]
    ,[Resume].ref.value(N'declare default element namespace "http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/Resume";
    (Address/Addr.PostalCode)[1]', 'nvarchar(20)') AS [Addr.PostalCode]
    ,[Resume].ref.value(N'declare default element namespace "http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/Resume";
    (/Resume/EMail)[1]', 'nvarchar(max)') AS [EMail]
    ,[Resume].ref.value(N'declare default element namespace "http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/Resume";
    (/Resume/WebSite)[1]', 'nvarchar(max)') AS [WebSite]
    ,jc.[ModifiedDate]
FROM [HumanResources].[JobCandidate] jc
CROSS APPLY jc.[Resume].nodes(N'declare default element namespace "http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/Resume";
/Resume') AS Resume(ref);
```



## Tables Used in View

Table Catalog	Table Schema	Table Name
AdventureWorks	HumanResources	JobCandidate

## Dependent Objects

Dependent Obj	Dependent Obj Type
JobCandidate	USER_TABLE

## vJobCandidateEducation

### Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
Edu.Degree	nvarchar	50		YES	
Edu.EndDate	datetime			YES	
Edu.GPA	nvarchar	5		YES	
Edu.GPAScale	nvarchar	5		YES	
Edu.Level	nvarchar	-1		YES	
Edu.Loc.City	nvarchar	100		YES	
Edu.Loc.CountryRegion	nvarchar	100		YES	
Edu.Loc.State	nvarchar	100		YES	
Edu.Major	nvarchar	50		YES	
Edu.Minor	nvarchar	50		YES	
Edu.School	nvarchar	100		YES	
Edu.StartDate	datetime			YES	

Column Name	Data Type	Length	Column Default	Is Nullable	Description
JobCandidateID	int			NO	

View Definition

## View Definition

```
CREATE VIEW [HumanResources].[vJobCandidateEducation]
AS
SELECT
    jc.[JobCandidateID]
    ,[Education].ref.value(N'declare default element namespace "http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/Resume";
    (Edu.Level)[1]', 'nvarchar(max)') AS [Edu.Level]
    ,CONVERT(datetime, REPLACE([Education].ref.value(N'declare default element namespace
"http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/Resume";
    (Edu.StartDate)[1]', 'nvarchar(20)') , 'Z', ''), 101) AS [Edu.StartDate]
    ,CONVERT(datetime, REPLACE([Education].ref.value(N'declare default element namespace
"http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/Resume";
    (Edu.EndDate)[1]', 'nvarchar(20)') , 'Z', ''), 101) AS [Edu.EndDate]
    ,[Education].ref.value(N'declare default element namespace "http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/Resume";
    (Edu.Degree)[1]', 'nvarchar(50)') AS [Edu.Degree]
    ,[Education].ref.value(N'declare default element namespace "http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/Resume";
    (Edu.Major)[1]', 'nvarchar(50)') AS [Edu.Major]
    ,[Education].ref.value(N'declare default element namespace "http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/Resume";
    (Edu.Minor)[1]', 'nvarchar(50)') AS [Edu.Minor]
    ,[Education].ref.value(N'declare default element namespace "http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/Resume";
    (Edu.GPA)[1]', 'nvarchar(5)') AS [Edu.GPA]
    ,[Education].ref.value(N'declare default element namespace "http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/Resume";
    (Edu.GPAScale)[1]', 'nvarchar(5)') AS [Edu.GPAScale]
    ,[Education].ref.value(N'declare default element namespace "http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/Resume";
    (Edu.School)[1]', 'nvarchar(100)') AS [Edu.School]
    ,[Education].ref.value(N'declare default element namespace "http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/Resume";
    (Edu.Location/Location/Loc.CountryRegion)[1]', 'nvarchar(100)') AS [Edu.Loc.CountryRegion]
    ,[Education].ref.value(N'declare default element namespace "http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/Resume";
    (Edu.Location/Location/Loc.State)[1]', 'nvarchar(100)') AS [Edu.Loc.State]
    ,[Education].ref.value(N'declare default element namespace "http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/Resume";
    (Edu.Location/Location/Loc.City)[1]', 'nvarchar(100)') AS [Edu.Loc.City]
FROM [HumanResources].[JobCandidate] jc
CROSS APPLY jc.[Resume].nodes(N'declare default element namespace "http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/Resume";
/Resume/Education') AS [Education](ref);
```

## Tables Used in View

Table Catalog	Table Schema	Table Name
AdventureWorks	HumanResources	JobCandidate

## Dependent Objects

Dependent Obj	Dependent Obj Type
JobCandidate	USER_TABLE

## vJobCandidateEmployment

### Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
Emp.EndDate	datetime			YES	
Emp.FunctionCategory	nvarchar	-1		YES	
Emp.IndustryCategory	nvarchar	-1		YES	
Emp.JobTitle	nvarchar	100		YES	
Emp.Loc.City	nvarchar	-1		YES	
Emp.Loc.CountryRegion	nvarchar	-1		YES	
Emp.Loc.State	nvarchar	-1		YES	
Emp.OrgName	nvarchar	100		YES	
Emp.Responsibility	nvarchar	-1		YES	
Emp.StartDate	datetime			YES	
JobCandidateID	int			NO	

## View Definition

### View Definition

```
CREATE VIEW [HumanResources].[vJobCandidateEmployment]
AS
SELECT
    jc.[JobCandidateID]
    ,CONVERT(datetime, REPLACE([Employment].ref.value(N'declare default element namespace
"http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/Resume";
    (Emp.StartDate)[1]', 'nvarchar(20)') , 'Z', ''), 101) AS [Emp.StartDate]
    ,CONVERT(datetime, REPLACE([Employment].ref.value(N'declare default element namespace
"http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/Resume";
    (Emp.EndDate)[1]', 'nvarchar(20)') , 'Z', ''), 101) AS [Emp.EndDate]
    ,[Employment].ref.value(N'declare default element namespace "http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/Resume";
    (Emp.OrgName)[1]', 'nvarchar(100)') AS [Emp.OrgName]
    ,[Employment].ref.value(N'declare default element namespace "http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/Resume";
    (Emp.JobTitle)[1]', 'nvarchar(100)') AS [Emp.JobTitle]
    ,[Employment].ref.value(N'declare default element namespace "http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/Resume";
    (Emp.Responsibility)[1]', 'nvarchar(max)') AS [Emp.Responsibility]
    ,[Employment].ref.value(N'declare default element namespace "http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/Resume";
    (Emp.FunctionCategory)[1]', 'nvarchar(max)') AS [Emp.FunctionCategory]
    ,[Employment].ref.value(N'declare default element namespace "http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/Resume";
    (Emp.IndustryCategory)[1]', 'nvarchar(max)') AS [Emp.IndustryCategory]
    ,[Employment].ref.value(N'declare default element namespace "http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/Resume";
    (Emp.Location/Location/Loc.CountryRegion)[1]', 'nvarchar(max)') AS [Emp.Loc.CountryRegion]
    ,[Employment].ref.value(N'declare default element namespace "http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/Resume";
    (Emp.Location/Location/Loc.State)[1]', 'nvarchar(max)') AS [Emp.Loc.State]
    ,[Employment].ref.value(N'declare default element namespace "http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/Resume";
    (Emp.Location/Location/Loc.City)[1]', 'nvarchar(max)') AS [Emp.Loc.City]
FROM [HumanResources].[JobCandidate] jc
CROSS APPLY jc.[Resume].nodes(N'declare default element namespace "http://schemas.microsoft.com/sqlserver/2004/07/adventure-
works/Resume";
/Resume/Employment') AS Employment(ref);
```

## Tables Used in View

Table Catalog	Table Schema	Table Name
AdventureWorks	HumanResources	JobCandidate

## Dependent Objects

Dependent Obj	Dependent Obj Type
JobCandidate	USER_TABLE

## vAdditionalContactInfo

### Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
City	nvarchar	50		YES	
ContactID	int			NO	
CountryRegion	nvarchar	50		YES	
EMailAddress	nvarchar	128		YES	
EMailSpecialInstructions	nvarchar	-1		YES	
EMailTelephoneNumber	nvarchar	50		YES	
FirstName	nvarchar	50		NO	
HomeAddressSpecialInstructions	nvarchar	-1		YES	
LastName	nvarchar	50		NO	
MiddleName	nvarchar	50		YES	
ModifiedDate	datetime			NO	
PostalCode	nvarchar	50		YES	

Column Name	Data Type	Length	Column Default	Is Nullable	Description
rowguid	uniqueidentifier			NO	
StateProvince	nvarchar	50		YES	
Street	nvarchar	50		YES	
TelephoneNumber	nvarchar	50		YES	
TelephoneSpecialInstructions	nvarchar	-1		YES	

[View Definition](#)

## View Definition

```
CREATE VIEW [Person].[vAdditionalContactInfo]
AS
SELECT
    [ContactID]
    ,[FirstName]
    ,[MiddleName]
    ,[LastName]
    ,[ContactInfo].ref.value(N'declare namespace ci="http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/ContactInfo";
    declare namespace act="http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/ContactTypes";
    (act:telephoneNumber)[1]/act:number', 'nvarchar(50)') AS [PhoneNumber]
    ,LTRIM(RTRIM([ContactInfo].ref.value(N'declare namespace ci="http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/ContactInfo";
    declare namespace act="http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/ContactTypes";
    (act:telephoneNumber/act:SpecialInstructions/text())[1]', 'nvarchar(max)'))) AS [TelephoneSpecialInstructions]
    ,[ContactInfo].ref.value(N'declare namespace ci="http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/ContactInfo";
    declare namespace act="http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/ContactTypes";
    (act:homePostalAddress/act:Street)[1]', 'nvarchar(50)') AS [Street]
    ,[ContactInfo].ref.value(N'declare namespace ci="http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/ContactInfo";
    declare namespace act="http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/ContactTypes";
    (act:homePostalAddress/act:City)[1]', 'nvarchar(50)') AS [City]
    ,[ContactInfo].ref.value(N'declare namespace ci="http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/ContactInfo";
    declare namespace act="http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/ContactTypes";
    (act:homePostalAddress/act:StateProvince)[1]', 'nvarchar(50)') AS [StateProvince]
    ,[ContactInfo].ref.value(N'declare namespace ci="http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/ContactInfo";
    declare namespace act="http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/ContactTypes";
    (act:homePostalAddress/act:PostalCode)[1]', 'nvarchar(50)') AS [PostalCode]
    ,[ContactInfo].ref.value(N'declare namespace ci="http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/ContactInfo";
    declare namespace act="http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/ContactTypes";
    (act:homePostalAddress/act:CountryRegion)[1]', 'nvarchar(50)') AS [CountryRegion]
    ,[ContactInfo].ref.value(N'declare namespace ci="http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/ContactInfo";
    declare namespace act="http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/ContactTypes";
    (act:homePostalAddress/act:SpecialInstructions/text())[1]', 'nvarchar(max)') AS [HomeAddressSpecialInstructions]
    ,[ContactInfo].ref.value(N'declare namespace ci="http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/ContactInfo";
    declare namespace act="http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/ContactTypes";
    (act:eMail/act:eMailAddress)[1]', 'nvarchar(128)') AS [EMailAddress]
    ,LTRIM(RTRIM([ContactInfo].ref.value(N'declare namespace ci="http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/ContactInfo";
    declare namespace act="http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/ContactTypes";
    (act:eMail/act:SpecialInstructions/text())[1]', 'nvarchar(max)'))) AS [EMailSpecialInstructions]
    ,[ContactInfo].ref.value(N'declare namespace ci="http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/ContactInfo";
```



### View Definition

```
declare namespace act="http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/ContactTypes";
(act:eMail/act:SpecialInstructions/act:telephoneNumber/act:number)[1], 'nvarchar(50)') AS [EMailTelephoneNumber]
,[rowguid]
,[ModifiedDate]
FROM [Person].[Contact]
OUTER APPLY [AdditionalContactInfo].nodes(
```

### Tables Used in View

Table Catalog	Table Schema	Table Name
AdventureWorks	Person	Contact

### Dependent Objects

Dependent Obj	Dependent Obj Type
Contact	USER_TABLE

### vStateProvinceCountryRegion

#### Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
CountryRegionCode	nvarchar	3		NO	
CountryRegionName	nvarchar	50		NO	
IsOnlyStateProvinceFlag	bit			NO	
StateProvinceCode	nchar	3		NO	

Column Name	Data Type	Length	Column Default	Is Nullable	Description
StateProvinceID	int			NO	Clustered index on the view vStateProvinceCountryRegion.
StateProvinceName	nvarchar	50		NO	
TerritoryID	int			NO	

### View Definition

View Definition
<pre> CREATE VIEW [Person].[vStateProvinceCountryRegion] WITH SCHEMABINDING AS SELECT     sp.[StateProvinceID]     ,sp.[StateProvinceCode]     ,sp.[IsOnlyStateProvinceFlag]     ,sp.[Name] AS [StateProvinceName]     ,sp.[TerritoryID]     ,cr.[CountryRegionCode]     ,cr.[Name] AS [CountryRegionName] FROM [Person].[StateProvince] sp     INNER JOIN [Person].[CountryRegion] cr     ON sp.[CountryRegionCode] = cr.[CountryRegionCode]; </pre>

### Tables Used in View

Table Catalog	Table Schema	Table Name
AdventureWorks	Person	CountryRegion
AdventureWorks	Person	StateProvince

## Dependent Objects

Dependent Obj	Dependent Obj Type
CountryRegion	USER_TABLE
StateProvince	USER_TABLE

## vProductAndDescription

### Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
CultureID	nchar	6		NO	
Description	nvarchar	400		NO	
Name	nvarchar	50		NO	
ProductID	int			NO	Clustered index on the view vProductAndDescription.
ProductModel	nvarchar	50		NO	

### View Definition

## View Definition

```

CREATE VIEW [Production].[vProductAndDescription]
WITH SCHEMABINDING
AS
-- View (indexed or standard) to display products and product descriptions by language.
SELECT
    p.[ProductID]
    ,p.[Name]
    ,pm.[Name] AS [ProductModel]
    ,pmx.[CultureID]
    ,pd.[Description]
FROM [Production].[Product] p
INNER JOIN [Production].[ProductModel] pm
ON p.[ProductModelID] = pm.[ProductModelID]
INNER JOIN [Production].[ProductModelProductDescriptionCulture] pmx
ON pm.[ProductModelID] = pmx.[ProductModelID]
INNER JOIN [Production].[ProductDescription] pd
ON pmx.[ProductDescriptionID] = pd.[ProductDescriptionID];
    
```

### Tables Used in View

Table Catalog	Table Schema	Table Name
AdventureWorks	Production	Product
AdventureWorks	Production	ProductDescription
AdventureWorks	Production	ProductModel
AdventureWorks	Production	ProductModelProductDescriptionCulture

### Dependent Objects

Dependent Obj	Dependent Obj Type
Product	USER_TABLE

Dependent Obj	Dependent Obj Type
ProductDescription	USER_TABLE
ProductModel	USER_TABLE
ProductModelProductDescriptionCulture	USER_TABLE

## vProductModelCatalogDescription

### Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
BikeFrame	nvarchar	-1		YES	
Color	nvarchar	256		YES	
Copyright	nvarchar	30		YES	
Crankset	nvarchar	256		YES	
MaintenanceDescription	nvarchar	256		YES	
Manufacturer	nvarchar	-1		YES	
Material	nvarchar	256		YES	
ModifiedDate	datetime			NO	
Name	nvarchar	50		NO	
NoOfYears	nvarchar	256		YES	
Pedal	nvarchar	256		YES	
PictureAngle	nvarchar	256		YES	
PictureSize	nvarchar	256		YES	
ProductLine	nvarchar	256		YES	
ProductModelID	int			NO	
ProductPhotoID	nvarchar	256		YES	

Column Name	Data Type	Length	Column Default	Is Nullable	Description
ProductURL	nvarchar	256		YES	
RiderExperience	nvarchar	1024		YES	
rowguid	uniqueidentifier			NO	
Saddle	nvarchar	256		YES	
Style	nvarchar	256		YES	
Summary	nvarchar	-1		YES	
WarrantyDescription	nvarchar	256		YES	
WarrantyPeriod	nvarchar	256		YES	
Wheel	nvarchar	256		YES	

[View Definition](#)

## View Definition

```
CREATE VIEW [Production].[vProductModelCatalogDescription]
AS
SELECT
    [ProductModelID]
    ,[Name]
    ,[CatalogDescription].value(N'declare namespace p1="http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/ProductModelDescription";
    declare namespace html="http://www.w3.org/1999/xhtml";
    (/p1:ProductDescription/p1:Summary/html:p)[1]', 'nvarchar(max)') AS [Summary]
    ,[CatalogDescription].value(N'declare namespace p1="http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/ProductModelDescription";
    (/p1:ProductDescription/p1:Manufacturer/p1:Name)[1]', 'nvarchar(max)') AS [Manufacturer]
    ,[CatalogDescription].value(N'declare namespace p1="http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/ProductModelDescription";
    (/p1:ProductDescription/p1:Manufacturer/p1:Copyright)[1]', 'nvarchar(30)') AS [Copyright]
    ,[CatalogDescription].value(N'declare namespace p1="http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/ProductModelDescription";
    (/p1:ProductDescription/p1:Manufacturer/p1:ProductURL)[1]', 'nvarchar(256)') AS [ProductURL]
    ,[CatalogDescription].value(N'declare namespace p1="http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/ProductModelDescription";
    declare namespace wm="http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/ProductModelWarrAndMain";
    (/p1:ProductDescription/p1:Features/wm:Warranty/wm:WarrantyPeriod)[1]', 'nvarchar(256)') AS [WarrantyPeriod]
    ,[CatalogDescription].value(N'declare namespace p1="http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/ProductModelDescription";
    declare namespace wm="http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/ProductModelWarrAndMain";
    (/p1:ProductDescription/p1:Features/wm:Warranty/wm:Description)[1]', 'nvarchar(256)') AS [WarrantyDescription]
    ,[CatalogDescription].value(N'declare namespace p1="http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/ProductModelDescription";
    declare namespace wm="http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/ProductModelWarrAndMain";
    (/p1:ProductDescription/p1:Features/wm:Maintenance/wm:NoOfYears)[1]', 'nvarchar(256)') AS [NoOfYears]
    ,[CatalogDescription].value(N'declare namespace p1="http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/ProductModelDescription";
    declare namespace wm="http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/ProductModelWarrAndMain";
    (/p1:ProductDescription/p1:Features/wm:Maintenance/wm:Description)[1]', 'nvarchar(256)') AS [MaintenanceDescription]
    ,[CatalogDescription].value(N'declare namespace p1="http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/ProductModelDescription";
    declare namespace wf="http://www.adventure-works.com/schemas/OtherFeatures";
    (/p1:ProductDescription/p1:Features/wf:wheel)[1]', 'nvarchar(256)') AS [Wheel]
    ,[CatalogDescription].value(N'declare namespace p1="http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/ProductModelDescription";
```

### View Definition

```
declare namespace wf="http://www.adventure-works.com/schemas/OtherFeatures";
(/p1:ProductDescription/p1:Features/wf:saddle)[1], 'nvarchar(256)') AS [Saddle]
,[CatalogDescription].value(N'declare namespace p1="http://schemas.microsoft.com/sqlserver/2004/07/adventure-
works/ProductModelDescription";
declare namespace wf="http://www.adventure-works.com/schemas/OtherFeatures";
(/p1:ProductDescription/p1:Features/wf:pedal)[1], 'nvarchar(256)') AS [Pedal]
,[CatalogDescription].value(N'declare namespace p1="http://schemas.microsoft.com/sqlserver/2004/07/adventure-
works/ProductModelDescription";
declare namespace wf="http://www.adventure-works.com/schemas/OtherFeatures";
(/p1:ProductDescription/p1:Features/wf:BikeFrame)[1], 'nvarchar(max)') AS [Bik
```

### Tables Used in View

Table Catalog	Table Schema	Table Name
AdventureWorks	Production	ProductModel

### Dependent Objects

Dependent Obj	Dependent Obj Type
ProductModel	USER_TABLE

### vProductModelInstructions

#### Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
Instructions	nvarchar	-1		YES	
LaborHours	decimal			YES	



Column Name	Data Type	Length	Column Default	Is Nullable	Description
LocationID	int			YES	
LotSize	int			YES	
MachineHours	decimal			YES	
ModifiedDate	datetime			NO	
Name	nvarchar	50		NO	
ProductModelID	int			NO	
rowguid	uniqueidentifier			NO	
SetupHours	decimal			YES	
Step	nvarchar	1024		YES	

View Definition

## View Definition

```
CREATE VIEW [Production].[vProductModelInstructions]
AS
SELECT
    [ProductModelID]
    ,[Name]
    ,[Instructions].value(N'declare default element namespace "http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/ProductModelManuInstructions";
    (/root/text())[1]', 'nvarchar(max)') AS [Instructions]
    ,[MfgInstructions].ref.value('@LocationID[1]', 'int') AS [LocationID]
    ,[MfgInstructions].ref.value('@SetupHours[1]', 'decimal(9, 4)') AS [SetupHours]
    ,[MfgInstructions].ref.value('@MachineHours[1]', 'decimal(9, 4)') AS [MachineHours]
    ,[MfgInstructions].ref.value('@LaborHours[1]', 'decimal(9, 4)') AS [LaborHours]
    ,[MfgInstructions].ref.value('@LotSize[1]', 'int') AS [LotSize]
    ,[Steps].ref.value('string(.)[1]', 'nvarchar(1024)') AS [Step]
    ,[rowguid]
    ,[ModifiedDate]
FROM [Production].[ProductModel]
CROSS APPLY [Instructions].nodes(N'declare default element namespace "http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/ProductModelManuInstructions";
    /root/Location') MfgInstructions(ref)
CROSS APPLY [MfgInstructions].ref.nodes('declare default element namespace "http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/ProductModelManuInstructions";
    step') Steps(ref);
```

### Tables Used in View

Table Catalog	Table Schema	Table Name
AdventureWorks	Production	ProductModel

### Dependent Objects

Dependent Obj	Dependent Obj Type
ProductModel	USER_TABLE

## vVendor

### Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
AddressLine1	nvarchar	60		NO	
AddressLine2	nvarchar	60		YES	
City	nvarchar	30		NO	
ContactType	nvarchar	50		NO	
CountryRegionName	nvarchar	50		NO	
EmailAddress	nvarchar	50		YES	
EmailPromotion	int			NO	
FirstName	nvarchar	50		NO	
LastName	nvarchar	50		NO	
MiddleName	nvarchar	50		YES	
Name	nvarchar	50		NO	
Phone	nvarchar	25		YES	
PostalCode	nvarchar	15		NO	
StateProvinceName	nvarchar	50		NO	
Suffix	nvarchar	10		YES	
Title	nvarchar	8		YES	
VendorID	int			NO	

## View Definition

### View Definition

```
CREATE VIEW [Purchasing].[vVendor] AS
SELECT
    v.[VendorID]
    ,v.[Name]
    ,ct.[Name] AS [ContactType]
    ,c.[Title]
    ,c.[FirstName]
    ,c.[MiddleName]
    ,c.[LastName]
    ,c.[Suffix]
    ,c.[Phone]
    ,c.[EmailAddress]
    ,c.[EmailPromotion]
    ,a.[AddressLine1]
    ,a.[AddressLine2]
    ,a.[City]
    ,[StateProvinceName] = sp.[Name]
    ,a.[PostalCode]
    ,[CountryRegionName] = cr.[Name]
FROM [Purchasing].[Vendor] v
    INNER JOIN [Purchasing].[VendorContact] vc
        ON vc.[VendorID] = v.[VendorID]
    INNER JOIN [Person].[Contact] c
        ON c.[ContactID] = vc.[ContactID]
    INNER JOIN [Person].[ContactType] ct
        ON vc.[ContactTypeID] = ct.[ContactTypeID]
    INNER JOIN [Purchasing].[VendorAddress] va
        ON va.[VendorID] = v.[VendorID]
    INNER JOIN [Person].[Address] a
        ON a.[AddressID] = va.[AddressID]
    INNER JOIN [Person].[StateProvince] sp
        ON sp.[StateProvinceID] = a.[StateProvinceID]
    INNER JOIN [Person].[CountryRegion] cr
        ON cr.[CountryRegionCode] = sp.[CountryRegionCode];
```

### Tables Used in View

Table Catalog	Table Schema	Table Name
AdventureWorks	Person	Address
AdventureWorks	Person	Contact
AdventureWorks	Person	ContactType
AdventureWorks	Person	CountryRegion
AdventureWorks	Person	StateProvince
AdventureWorks	Purchasing	Vendor
AdventureWorks	Purchasing	VendorAddress
AdventureWorks	Purchasing	VendorContact

### Dependent Objects

Dependent Obj	Dependent Obj Type
Address	USER_TABLE
Contact	USER_TABLE
ContactType	USER_TABLE
CountryRegion	USER_TABLE
StateProvince	USER_TABLE
Vendor	USER_TABLE
VendorAddress	USER_TABLE
VendorContact	USER_TABLE

## IndividualCustomer

### Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
AddressLine1	nvarchar	60		NO	
AddressLine2	nvarchar	60		YES	
AddressType	nvarchar	50		NO	
City	nvarchar	30		NO	
CountryRegionName	nvarchar	50		NO	
CustomerID	int			NO	
Demographics	xml	-1		YES	
EmailAddress	nvarchar	50		YES	
EmailPromotion	int			NO	
FirstName	nvarchar	50		NO	
LastName	nvarchar	50		NO	
MiddleName	nvarchar	50		YES	
Phone	nvarchar	25		YES	
PostalCode	nvarchar	15		NO	
StateProvinceName	nvarchar	50		NO	
Suffix	nvarchar	10		YES	
Title	nvarchar	8		YES	

## View Definition

### View Definition

```
CREATE VIEW [Sales].[vIndividualCustomer]
AS
SELECT
    i.[CustomerID]
    ,c.[Title]
    ,c.[FirstName]
    ,c.[MiddleName]
    ,c.[LastName]
    ,c.[Suffix]
    ,c.[Phone]
    ,c.[EmailAddress]
    ,c.[EmailPromotion]
    ,at.[Name] AS [AddressType]
    ,a.[AddressLine1]
    ,a.[AddressLine2]
    ,a.[City]
    ,[StateProvinceName] = sp.[Name]
    ,a.[PostalCode]
    ,[CountryRegionName] = cr.[Name]
    ,i.[Demographics]
FROM [Sales].[Individual] i
    INNER JOIN [Person].[Contact] c
        ON c.[ContactID] = i.[ContactID]
    INNER JOIN [Sales].[CustomerAddress] ca
        ON ca.[CustomerID] = i.[CustomerID]
    INNER JOIN [Person].[Address] a
        ON a.[AddressID] = ca.[AddressID]
    INNER JOIN [Person].[StateProvince] sp
        ON sp.[StateProvinceID] = a.[StateProvinceID]
    INNER JOIN [Person].[CountryRegion] cr
        ON cr.[CountryRegionCode] = sp.[CountryRegionCode]
    INNER JOIN [Person].[AddressType] at
        ON ca.[AddressTypeID] = at.[AddressTypeID]
WHERE i.[CustomerID] IN (SELECT [Sales].[Customer].[CustomerID]
    FROM [Sales].[Customer] WHERE UPPER([Sales].[Customer].[CustomerType]) = 'I');
```

## Tables Used in View

Table Catalog	Table Schema	Table Name
AdventureWorks	Person	Address
AdventureWorks	Person	AddressType
AdventureWorks	Person	Contact
AdventureWorks	Person	CountryRegion
AdventureWorks	Person	StateProvince
AdventureWorks	Sales	Customer
AdventureWorks	Sales	CustomerAddress
AdventureWorks	Sales	Individual

## Dependent Objects

Dependent Obj	Dependent Obj Type
Address	USER_TABLE
AddressType	USER_TABLE
Contact	USER_TABLE
CountryRegion	USER_TABLE
Customer	USER_TABLE
CustomerAddress	USER_TABLE
Individual	USER_TABLE
StateProvince	USER_TABLE

## IndividualDemographics

### Column Details



Column Name	Data Type	Length	Column Default	Is Nullable	Description
BirthDate	datetime			YES	
CustomerID	int			NO	
DateFirstPurchase	datetime			YES	
Education	nvarchar	30		YES	
Gender	nvarchar	1		YES	
HomeOwnerFlag	bit			YES	
MaritalStatus	nvarchar	1		YES	
NumberCarsOwned	int			YES	
NumberChildrenAtHome	int			YES	
Occupation	nvarchar	30		YES	
TotalChildren	int			YES	
TotalPurchaseYTD	money			YES	
YearlyIncome	nvarchar	30		YES	

[View Definition](#)

## View Definition

```
CREATE VIEW [Sales].[vIndividualDemographics]
AS
SELECT
    i.[CustomerID]
    ,[IndividualSurvey].[ref].[value](N'declare default element namespace "http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/IndividualSurvey";
        TotalPurchaseYTD[1]', 'money') AS [TotalPurchaseYTD]
    ,CONVERT(datetime, REPLACE([IndividualSurvey].[ref].[value](N'declare default element namespace
"http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/IndividualSurvey";
        DateFirstPurchase[1]', 'nvarchar(20)') , 'Z', ''), 101) AS [DateFirstPurchase]
    ,CONVERT(datetime, REPLACE([IndividualSurvey].[ref].[value](N'declare default element namespace
"http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/IndividualSurvey";
        BirthDate[1]', 'nvarchar(20)') , 'Z', ''), 101) AS [BirthDate]
    ,[IndividualSurvey].[ref].[value](N'declare default element namespace "http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/IndividualSurvey";
        MaritalStatus[1]', 'nvarchar(1)') AS [MaritalStatus]
    ,[IndividualSurvey].[ref].[value](N'declare default element namespace "http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/IndividualSurvey";
        YearlyIncome[1]', 'nvarchar(30)') AS [YearlyIncome]
    ,[IndividualSurvey].[ref].[value](N'declare default element namespace "http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/IndividualSurvey";
        Gender[1]', 'nvarchar(1)') AS [Gender]
    ,[IndividualSurvey].[ref].[value](N'declare default element namespace "http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/IndividualSurvey";
        TotalChildren[1]', 'integer') AS [TotalChildren]
    ,[IndividualSurvey].[ref].[value](N'declare default element namespace "http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/IndividualSurvey";
        NumberChildrenAtHome[1]', 'integer') AS [NumberChildrenAtHome]
    ,[IndividualSurvey].[ref].[value](N'declare default element namespace "http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/IndividualSurvey";
        Education[1]', 'nvarchar(30)') AS [Education]
    ,[IndividualSurvey].[ref].[value](N'declare default element namespace "http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/IndividualSurvey";
        Occupation[1]', 'nvarchar(30)') AS [Occupation]
    ,[IndividualSurvey].[ref].[value](N'declare default element namespace "http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/IndividualSurvey";
        HomeOwnerFlag[1]', 'bit') AS [HomeOwnerFlag]
    ,[IndividualSurvey].[ref].[value](N'declare default element namespace "http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/IndividualSurvey";
        NumberCarsOwned[1]', 'integer') AS [NumberCarsOwned]
```

## View Definition

```
FROM [Sales].[Individual] i
CROSS APPLY i.[Demographics].nodes(N'declare default element namespace "http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/IndividualSurvey";
  /IndividualSurvey') AS [IndividualSurvey](ref)
WHERE [Demographics] IS NOT NULL;
```

## Tables Used in View

Table Catalog	Table Schema	Table Name
AdventureWorks	Sales	Individual

## Dependent Objects

Dependent Obj	Dependent Obj Type
Individual	USER_TABLE

## vSalesPerson

### Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
AddressLine1	nvarchar	60		NO	
AddressLine2	nvarchar	60		YES	
City	nvarchar	30		NO	
CountryRegionName	nvarchar	50		NO	
EmailAddress	nvarchar	50		YES	

Column Name	Data Type	Length	Column Default	Is Nullable	Description
EmailPromotion	int			NO	
FirstName	nvarchar	50		NO	
JobTitle	nvarchar	50		NO	
LastName	nvarchar	50		NO	
MiddleName	nvarchar	50		YES	
Phone	nvarchar	25		YES	
PostalCode	nvarchar	15		NO	
SalesLastYear	money			NO	
SalesPersonID	int			NO	
SalesQuota	money			YES	
SalesYTD	money			NO	
StateProvinceName	nvarchar	50		NO	
Suffix	nvarchar	10		YES	
TerritoryGroup	nvarchar	50		YES	
TerritoryName	nvarchar	50		YES	
Title	nvarchar	8		YES	

View Definition

## View Definition

```
CREATE VIEW [Sales].[vSalesPerson]
AS
SELECT
    s.[SalesPersonID]
    ,c.[Title]
    ,c.[FirstName]
    ,c.[MiddleName]
    ,c.[LastName]
    ,c.[Suffix]
    ,[JobTitle] = e.[Title]
    ,c.[Phone]
    ,c.[EmailAddress]
    ,c.[EmailPromotion]
    ,a.[AddressLine1]
    ,a.[AddressLine2]
    ,a.[City]
    ,[StateProvinceName] = sp.[Name]
    ,a.[PostalCode]
    ,[CountryRegionName] = cr.[Name]
    ,[TerritoryName] = st.[Name]
    ,[TerritoryGroup] = st.[Group]
    ,s.[SalesQuota]
    ,s.[SalesYTD]
    ,s.[SalesLastYear]
FROM [Sales].[SalesPerson] s
    INNER JOIN [HumanResources].[Employee] e
    ON e.[EmployeeID] = s.[SalesPersonID]
    LEFT OUTER JOIN [Sales].[SalesTerritory] st
    ON st.[TerritoryID] = s.[TerritoryID]
    INNER JOIN [Person].[Contact] c
    ON c.[ContactID] = e.[ContactID]
    INNER JOIN [HumanResources].[EmployeeAddress] ea
    ON e.[EmployeeID] = ea.[EmployeeID]
    INNER JOIN [Person].[Address] a
    ON ea.[AddressID] = a.[AddressID]
    INNER JOIN [Person].[StateProvince] sp
    ON sp.[StateProvinceID] = a.[StateProvinceID]
    INNER JOIN [Person].[CountryRegion] cr
    ON cr.[CountryRegionCode] = sp.[CountryRegionCode];
```

### Tables Used in View

Table Catalog	Table Schema	Table Name
AdventureWorks	HumanResources	Employee
AdventureWorks	HumanResources	EmployeeAddress
AdventureWorks	Person	Address
AdventureWorks	Person	Contact
AdventureWorks	Person	CountryRegion
AdventureWorks	Person	StateProvince
AdventureWorks	Sales	SalesPerson
AdventureWorks	Sales	SalesTerritory

### Dependent Objects

Dependent Obj	Dependent Obj Type
Address	USER_TABLE
Contact	USER_TABLE
CountryRegion	USER_TABLE
Employee	USER_TABLE
EmployeeAddress	USER_TABLE
SalesPerson	USER_TABLE
SalesTerritory	USER_TABLE
StateProvince	USER_TABLE

## vSalesPersonSalesByFiscalYears

### Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
2002	money			YES	
2003	money			YES	
2004	money			YES	
FullName	nvarchar	152		YES	
SalesPersonID	int			YES	
SalesTerritory	nvarchar	50		NO	
Title	nvarchar	50		NO	

[View Definition](#)

## View Definition

```
CREATE VIEW [Sales].[vSalesPersonSalesByFiscalYears]
AS
SELECT
    pvt.[SalesPersonID]
    ,pvt.[FullName]
    ,pvt.[Title]
    ,pvt.[SalesTerritory]
    ,pvt.[2002]
    ,pvt.[2003]
    ,pvt.[2004]
FROM (SELECT
    soh.[SalesPersonID]
    ,c.[FirstName] + ' ' + COALESCE(c.[MiddleName], '') + ' ' + c.[LastName] AS [FullName]
    ,e.[Title]
    ,st.[Name] AS [SalesTerritory]
    ,soh.[SubTotal]
    ,YEAR(DATEADD(m, 6, soh.[OrderDate])) AS [FiscalYear]
FROM [Sales].[SalesPerson] sp
INNER JOIN [Sales].[SalesOrderHeader] soh
ON sp.[SalesPersonID] = soh.[SalesPersonID]
INNER JOIN [Sales].[SalesTerritory] st
ON sp.[TerritoryID] = st.[TerritoryID]
INNER JOIN [HumanResources].[Employee] e
ON soh.[SalesPersonID] = e.[EmployeeID]
INNER JOIN [Person].[Contact] c
ON e.[ContactID] = c.ContactID
) AS soh
PIVOT
(
    SUM([SubTotal])
    FOR [FiscalYear]
    IN ([2002], [2003], [2004])
) AS pvt;
```

Tables Used in View



Table Catalog	Table Schema	Table Name
AdventureWorks	HumanResources	Employee
AdventureWorks	Person	Contact
AdventureWorks	Sales	SalesOrderHeader
AdventureWorks	Sales	SalesPerson
AdventureWorks	Sales	SalesTerritory

#### Dependent Objects

Dependent Obj	Dependent Obj Type
Contact	USER_TABLE
Employee	USER_TABLE
SalesOrderHeader	USER_TABLE
SalesPerson	USER_TABLE
SalesTerritory	USER_TABLE

### vStoreWithDemographics

#### Column Details

Column Name	Data Type	Length	Column Default	Is Nullable	Description
AddressLine1	nvarchar	60		NO	
AddressLine2	nvarchar	60		YES	
AddressType	nvarchar	50		NO	
AnnualRevenue	money			YES	
AnnualSales	money			YES	

Column Name	Data Type	Length	Column Default	Is Nullable	Description
BankName	nvarchar	50		YES	
Brands	nvarchar	30		YES	
BusinessType	nvarchar	5		YES	
City	nvarchar	30		NO	
ContactType	nvarchar	50		NO	
CountryRegionName	nvarchar	50		NO	
CustomerID	int			NO	
EmailAddress	nvarchar	50		YES	
EmailPromotion	int			NO	
FirstName	nvarchar	50		NO	
Internet	nvarchar	30		YES	
LastName	nvarchar	50		NO	
MiddleName	nvarchar	50		YES	
Name	nvarchar	50		NO	
NumberEmployees	int			YES	
Phone	nvarchar	25		YES	
PostalCode	nvarchar	15		NO	
Specialty	nvarchar	50		YES	
SquareFeet	int			YES	
StateProvinceName	nvarchar	50		NO	
Suffix	nvarchar	10		YES	
Title	nvarchar	8		YES	
YearOpened	int			YES	

View Definition

## View Definition

```
CREATE VIEW [Sales].[vStoreWithDemographics] AS
SELECT
    s.[CustomerID]
    ,s.[Name]
    ,ct.[Name] AS [ContactType]
    ,c.[Title]
    ,c.[FirstName]
    ,c.[MiddleName]
    ,c.[LastName]
    ,c.[Suffix]
    ,c.[Phone]
    ,c.[EmailAddress]
    ,c.[EmailPromotion]
    ,at.[Name] AS [AddressType]
    ,a.[AddressLine1]
    ,a.[AddressLine2]
    ,a.[City]
    ,sp.[Name] AS [StateProvinceName]
    ,a.[PostalCode]
    ,cr.[Name] AS [CountryRegionName]
    ,s.[Demographics].value('declare default element namespace "http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/StoreSurvey";
        (/StoreSurvey/AnnualSales)[1]', 'money') AS [AnnualSales]
    ,s.[Demographics].value('declare default element namespace "http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/StoreSurvey";
        (/StoreSurvey/AnnualRevenue)[1]', 'money') AS [AnnualRevenue]
    ,s.[Demographics].value('declare default element namespace "http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/StoreSurvey";
        (/StoreSurvey/BankName)[1]', 'nvarchar(50)') AS [BankName]
    ,s.[Demographics].value('declare default element namespace "http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/StoreSurvey";
        (/StoreSurvey/BusinessType)[1]', 'nvarchar(5)') AS [BusinessType]
    ,s.[Demographics].value('declare default element namespace "http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/StoreSurvey";
        (/StoreSurvey/YearOpened)[1]', 'integer') AS [YearOpened]
    ,s.[Demographics].value('declare default element namespace "http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/StoreSurvey";
        (/StoreSurvey/Specialty)[1]', 'nvarchar(50)') AS [Specialty]
    ,s.[Demographics].value('declare default element namespace "http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/StoreSurvey";
```

## View Definition

```

    (/StoreSurvey/SquareFeet)[1]', 'integer') AS [SquareFeet]
,s.[Demographics].value('declare default element namespace "http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/StoreSurvey";
    (/StoreSurvey/Brands)[1]', 'nvarchar(30)') AS [Brands]
,s.[Demographics].value('declare default element namespace "http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/StoreSurvey";
    (/StoreSurvey/Internet)[1]', 'nvarchar(30)') AS [Internet]
,s.[Demographics].value('declare default element namespace "http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/StoreSurvey";
    (/StoreSurvey/NumberEmployees)[1]', 'integer') AS [NumberEmployees]
FROM [Sales].[Store] s
INNER JOIN [Sales].[StoreContact] sc
ON sc.[CustomerID] = s.[CustomerID]
INNER JOIN [Person].[Contact] c
ON c.[ContactID] = sc.[ContactID]
INNER JOIN [Person].[ContactType] ct
ON sc.[ContactTypeID] = ct.[ContactTypeID]
INNER JOIN [Sales].[CustomerAddress] ca
ON ca.[CustomerID] = s.[CustomerID]
INNER JOIN [Person].[Address] a
ON a.[AddressID] = ca.[AddressID]
INNER JOIN [Person].[StateProvince] sp
ON sp.[StateProvinceID] = a.[StateProvinceID]
INNER JOIN [Person].[CountryRegion] cr
ON cr.[CountryRegionCode] = sp.[CountryRegionCode]
INNER JOIN [Person].[AddressType] at
ON ca.[AddressTypeID] = at.[AddressTypeID]
WHERE s.[CustomerID] IN (SELECT [Sales].[Customer].[CustomerID]
FROM [Sales].[Customer] WHERE UPPER([Sales].[Customer].[CustomerType]) = 'S');

```

### Tables Used in View

Table Catalog	Table Schema	Table Name
AdventureWorks	Person	Address
AdventureWorks	Person	AddressType
AdventureWorks	Person	Contact

Table Catalog	Table Schema	Table Name
AdventureWorks	Person	ContactType
AdventureWorks	Person	CountryRegion
AdventureWorks	Person	StateProvince
AdventureWorks	Sales	Customer
AdventureWorks	Sales	CustomerAddress
AdventureWorks	Sales	Store
AdventureWorks	Sales	StoreContact

#### Dependent Objects

Dependent Obj	Dependent Obj Type
Address	USER_TABLE
AddressType	USER_TABLE
Contact	USER_TABLE
ContactType	USER_TABLE
CountryRegion	USER_TABLE
Customer	USER_TABLE
CustomerAddress	USER_TABLE
StateProvince	USER_TABLE
Store	USER_TABLE
StoreContact	USER_TABLE

## Procedures

Procedure Name	Procedure Schema
uspPrintError	dbo
uspLogError	dbo
uspGetBillOfMaterials	dbo
uspGetEmployeeManagers	dbo
uspGetManagerEmployees	dbo
uspGetWhereUsedProductID	dbo
uspUpdateEmployeeHireInfo	HumanResources
uspUpdateEmployeeLogin	HumanResources
uspUpdateEmployeePersonalInfo	HumanResources

### uspPrintError

Definition

## Routine Definition

```
-- uspPrintError prints error information about the error that caused
-- execution to jump to the CATCH block of a TRY...CATCH construct.
-- Should be executed from within the scope of a CATCH block otherwise
-- it will return without printing any error information.
CREATE PROCEDURE [dbo].[uspPrintError]
AS
BEGIN
    SET NOCOUNT ON;

    -- Print error information.
    PRINT 'Error ' + CONVERT(varchar(50), ERROR_NUMBER()) +
        ', Severity ' + CONVERT(varchar(5), ERROR_SEVERITY()) +
        ', State ' + CONVERT(varchar(5), ERROR_STATE()) +
        ', Procedure ' + ISNULL(ERROR_PROCEDURE(), '-') +
        ', Line ' + CONVERT(varchar(5), ERROR_LINE());
    PRINT ERROR_MESSAGE();
END;
```

## Dependent Objects

Dependent Obj	Dependent Obj Type
uspLogError	SQL_STORED_PROCEDURE
dVendor	SQL_TRIGGER
iduSalesOrderDetail	SQL_TRIGGER
iPurchaseOrderDetail	SQL_TRIGGER
iStore	SQL_TRIGGER
iWorkOrder	SQL_TRIGGER
uPurchaseOrderDetail	SQL_TRIGGER
uPurchaseOrderHeader	SQL_TRIGGER
uSalesOrderHeader	SQL_TRIGGER



Dependent Obj	Dependent Obj Type
uWorkOrder	SQL_TRIGGER

## uspLogError

Definition

## Routine Definition

```
-- uspLogError logs error information in the ErrorLog table about the
-- error that caused execution to jump to the CATCH block of a
-- TRY...CATCH construct. This should be executed from within the scope
-- of a CATCH block otherwise it will return without inserting error
-- information.
CREATE PROCEDURE [dbo].[uspLogError]
    @ErrorLogID [int] = 0 OUTPUT -- contains the ErrorLogID of the row inserted
AS
    -- by uspLogError in the ErrorLog table
BEGIN
    SET NOCOUNT ON;

    -- Output parameter value of 0 indicates that error
    -- information was not logged
    SET @ErrorLogID = 0;

    BEGIN TRY
        -- Return if there is no error information to log
        IF ERROR_NUMBER() IS NULL
            RETURN;

        -- Return if inside an uncommittable transaction.
        -- Data insertion/modification is not allowed when
        -- a transaction is in an uncommittable state.
        IF XACT_STATE() = -1
            BEGIN
                PRINT 'Cannot log error since the current transaction is in an uncommittable state. '
                    + 'Rollback the transaction before executing uspLogError in order to successfully log error information.';
                RETURN;
            END
    END

    INSERT [dbo].[ErrorLog]
    (
        [UserName],
        [ErrorNumber],
        [ErrorSeverity],
        [ErrorState],
        [ErrorProcedure],
        [ErrorLine],
        [ErrorMessage]
    )
```

## Routine Definition

```

VALUES
(
  CONVERT(sysname, CURRENT_USER),
  ERROR_NUMBER(),
  ERROR_SEVERITY(),
  ERROR_STATE(),
  ERROR_PROCEDURE(),
  ERROR_LINE(),
  ERROR_MESSAGE()
);

-- Pass back the ErrorLogID of the row inserted
SET @ErrorLogID = @@IDENTITY;
END TRY
BEGIN CATCH
  PRINT 'An error occurred in stored procedure uspLogError: ';
  EXECUTE [dbo].[uspPrintError];
  RETURN -1;
END CATCH
END;

```

### Dependent Objects

Dependent Obj	Dependent Obj Type
uspPrintError	SQL_STORED_PROCEDURE
uspUpdateEmployeeHireInfo	SQL_STORED_PROCEDURE
uspUpdateEmployeeLogin	SQL_STORED_PROCEDURE
uspUpdateEmployeePersonalInfo	SQL_STORED_PROCEDURE
dVendor	SQL_TRIGGER
iduSalesOrderDetail	SQL_TRIGGER
iPurchaseOrderDetail	SQL_TRIGGER
iStore	SQL_TRIGGER

Dependent Obj	Dependent Obj Type
iWorkOrder	SQL_TRIGGER
uPurchaseOrderDetail	SQL_TRIGGER
uPurchaseOrderHeader	SQL_TRIGGER
uSalesOrderHeader	SQL_TRIGGER
uWorkOrder	SQL_TRIGGER
ErrorLog	USER_TABLE

## **uspGetBillOfMaterials**

Definition

## Routine Definition

```
CREATE PROCEDURE [dbo].[uspGetBillOfMaterials]
    @StartProductID [int],
    @CheckDate [datetime]
AS
BEGIN
    SET NOCOUNT ON;

    -- Use recursive query to generate a multi-level Bill of Material (i.e. all level 1
    -- components of a level 0 assembly, all level 2 components of a level 1 assembly)
    -- The CheckDate eliminates any components that are no longer used in the product on this date.
    WITH [BOM_cte]([ProductAssemblyID], [ComponentID], [ComponentDesc], [PerAssemblyQty], [StandardCost], [ListPrice], [BOMLevel],
[RecursionLevel]) -- CTE name and columns
    AS (
        SELECT b.[ProductAssemblyID], b.[ComponentID], p.[Name], b.[PerAssemblyQty], p.[StandardCost], p.[ListPrice], b.[BOMLevel], 0 -- Get
the initial list of components for the bike assembly
        FROM [Production].[BillOfMaterials] b
        INNER JOIN [Production].[Product] p
        ON b.[ComponentID] = p.[ProductID]
        WHERE b.[ProductAssemblyID] = @StartProductID
        AND @CheckDate >= b.[StartDate]
        AND @CheckDate <= ISNULL(b.[EndDate], @CheckDate)
        UNION ALL
        SELECT b.[ProductAssemblyID], b.[ComponentID], p.[Name], b.[PerAssemblyQty], p.[StandardCost], p.[ListPrice], b.[BOMLevel],
[RecursionLevel] + 1 -- Join recursive member to anchor
        FROM [BOM_cte] cte
        INNER JOIN [Production].[BillOfMaterials] b
        ON b.[ProductAssemblyID] = cte.[ComponentID]
        INNER JOIN [Production].[Product] p
        ON b.[ComponentID] = p.[ProductID]
        WHERE @CheckDate >= b.[StartDate]
        AND @CheckDate <= ISNULL(b.[EndDate], @CheckDate)
    )
    -- Outer select from the CTE
    SELECT b.[ProductAssemblyID], b.[ComponentID], b.[ComponentDesc], SUM(b.[PerAssemblyQty]) AS [TotalQuantity] , b.[StandardCost],
b.[ListPrice], b.[BOMLevel], b.[RecursionLevel]
    FROM [BOM_cte] b
    GROUP BY b.[ComponentID], b.[ComponentDesc], b.[ProductAssemblyID], b.[BOMLevel], b.[RecursionLevel], b.[StandardCost], b.[ListPrice]
    ORDER BY b.[BOMLevel], b.[ProductAssemblyID], b.[ComponentID]
    OPTION (MAXRECURSION 25)
END;
```

## Dependent Objects

Dependent Obj	Dependent Obj Type
BillOfMaterials	USER_TABLE
Product	USER_TABLE

## uspGetEmployeeManagers

Definition

## Routine Definition

```
CREATE PROCEDURE [dbo].[uspGetEmployeeManagers]
    @EmployeeID [int]
AS
BEGIN
    SET NOCOUNT ON;

    -- Use recursive query to list out all Employees required for a particular Manager
    WITH [EMP_cte]([EmployeeID], [ManagerID], [FirstName], [LastName], [Title], [RecursionLevel]) -- CTE name and columns
    AS (
        SELECT e.[EmployeeID], e.[ManagerID], c.[FirstName], c.[LastName], e.[Title], 0 -- Get the initial Employee
        FROM [HumanResources].[Employee] e
            INNER JOIN [Person].[Contact] c
            ON e.[ContactID] = c.[ContactID]
        WHERE e.[EmployeeID] = @EmployeeID
        UNION ALL
        SELECT e.[EmployeeID], e.[ManagerID], c.[FirstName], c.[LastName], e.[Title], [RecursionLevel] + 1 -- Join recursive member to anchor
        FROM [HumanResources].[Employee] e
            INNER JOIN [EMP_cte]
            ON e.[EmployeeID] = [EMP_cte].[ManagerID]
            INNER JOIN [Person].[Contact] c
            ON e.[ContactID] = c.[ContactID]
    )
    -- Join back to Employee to return the manager name
    SELECT [EMP_cte].[RecursionLevel], [EMP_cte].[EmployeeID], [EMP_cte].[FirstName], [EMP_cte].[LastName],
        [EMP_cte].[ManagerID], c.[FirstName] AS 'ManagerFirstName', c.[LastName] AS 'ManagerLastName' -- Outer select from the CTE
    FROM [EMP_cte]
        INNER JOIN [HumanResources].[Employee] e
        ON [EMP_cte].[ManagerID] = e.[EmployeeID]
        INNER JOIN [Person].[Contact] c
        ON e.[ContactID] = c.[ContactID]
    ORDER BY [RecursionLevel], [ManagerID], [EmployeeID]
    OPTION (MAXRECURSION 25)
END;
```

Dependent Objects

Dependent Obj	Dependent Obj Type
Contact	USER_TABLE
Employee	USER_TABLE

## uspGetManagerEmployees

Definition



## Routine Definition

```
CREATE PROCEDURE [dbo].[uspGetManagerEmployees]
    @ManagerID [int]
AS
BEGIN
    SET NOCOUNT ON;

    -- Use recursive query to list out all Employees required for a particular Manager
    WITH [EMP_cte]([EmployeeID], [ManagerID], [FirstName], [LastName], [RecursionLevel]) -- CTE name and columns
    AS (
        SELECT e.[EmployeeID], e.[ManagerID], c.[FirstName], c.[LastName], 0 -- Get the initial list of Employees for Manager n
        FROM [HumanResources].[Employee] e
            INNER JOIN [Person].[Contact] c
            ON e.[ContactID] = c.[ContactID]
        WHERE [ManagerID] = @ManagerID
        UNION ALL
        SELECT e.[EmployeeID], e.[ManagerID], c.[FirstName], c.[LastName], [RecursionLevel] + 1 -- Join recursive member to anchor
        FROM [HumanResources].[Employee] e
            INNER JOIN [EMP_cte]
            ON e.[ManagerID] = [EMP_cte].[EmployeeID]
            INNER JOIN [Person].[Contact] c
            ON e.[ContactID] = c.[ContactID]
    )
    -- Join back to Employee to return the manager name
    SELECT [EMP_cte].[RecursionLevel], [EMP_cte].[ManagerID], c.[FirstName] AS 'ManagerFirstName', c.[LastName] AS 'ManagerLastName',
        [EMP_cte].[EmployeeID], [EMP_cte].[FirstName], [EMP_cte].[LastName] -- Outer select from the CTE
    FROM [EMP_cte]
        INNER JOIN [HumanResources].[Employee] e
        ON [EMP_cte].[ManagerID] = e.[EmployeeID]
        INNER JOIN [Person].[Contact] c
        ON e.[ContactID] = c.[ContactID]
    ORDER BY [RecursionLevel], [ManagerID], [EmployeeID]
    OPTION (MAXRECURSION 25)
END;
```

Dependent Objects

Dependent Obj	Dependent Obj Type
Contact	USER_TABLE
Employee	USER_TABLE

## uspGetWhereUsedProductID

Definition

## Routine Definition

```
CREATE PROCEDURE [dbo].[uspGetWhereUsedProductID]
    @StartProductID [int],
    @CheckDate [datetime]
AS
BEGIN
    SET NOCOUNT ON;

    --Use recursive query to generate a multi-level Bill of Material (i.e. all level 1 components of a level 0 assembly, all level 2 components of a
    level 1 assembly)
    WITH [BOM_cte]([ProductAssemblyID], [ComponentID], [ComponentDesc], [PerAssemblyQty], [StandardCost], [ListPrice], [BOMLevel],
    [RecursionLevel]) -- CTE name and columns
    AS (
        SELECT b.[ProductAssemblyID], b.[ComponentID], p.[Name], b.[PerAssemblyQty], p.[StandardCost], p.[ListPrice], b.[BOMLevel], 0 -- Get
        the initial list of components for the bike assembly
        FROM [Production].[BillOfMaterials] b
        INNER JOIN [Production].[Product] p
        ON b.[ProductAssemblyID] = p.[ProductID]
        WHERE b.[ComponentID] = @StartProductID
        AND @CheckDate >= b.[StartDate]
        AND @CheckDate <= ISNULL(b.[EndDate], @CheckDate)
        UNION ALL
        SELECT b.[ProductAssemblyID], b.[ComponentID], p.[Name], b.[PerAssemblyQty], p.[StandardCost], p.[ListPrice], b.[BOMLevel],
    [RecursionLevel] + 1 -- Join recursive member to anchor
        FROM [BOM_cte] cte
        INNER JOIN [Production].[BillOfMaterials] b
        ON cte.[ProductAssemblyID] = b.[ComponentID]
        INNER JOIN [Production].[Product] p
        ON b.[ProductAssemblyID] = p.[ProductID]
        WHERE @CheckDate >= b.[StartDate]
        AND @CheckDate <= ISNULL(b.[EndDate], @CheckDate)
    )
    -- Outer select from the CTE
    SELECT b.[ProductAssemblyID], b.[ComponentID], b.[ComponentDesc], SUM(b.[PerAssemblyQty]) AS [TotalQuantity] , b.[StandardCost],
    b.[ListPrice], b.[BOMLevel], b.[RecursionLevel]
    FROM [BOM_cte] b
    GROUP BY b.[ComponentID], b.[ComponentDesc], b.[ProductAssemblyID], b.[BOMLevel], b.[RecursionLevel], b.[StandardCost], b.[ListPrice]
    ORDER BY b.[BOMLevel], b.[ProductAssemblyID], b.[ComponentID]
    OPTION (MAXRECURSION 25)
END;
```

## Dependent Objects

Dependent Obj	Dependent Obj Type
BillOfMaterials	USER_TABLE
Product	USER_TABLE

## **uspUpdateEmployeeHireInfo**

Definition

## Routine Definition

```
CREATE PROCEDURE [HumanResources].[uspUpdateEmployeeHireInfo]
    @EmployeeID [int],
    @Title [nvarchar](50),
    @HireDate [datetime],
    @RateChangeDate [datetime],
    @Rate [money],
    @PayFrequency [tinyint],
    @CurrentFlag [dbo].[Flag]
WITH EXECUTE AS CALLER
AS
BEGIN
    SET NOCOUNT ON;

    BEGIN TRY
        BEGIN TRANSACTION;

        UPDATE [HumanResources].[Employee]
        SET [Title] = @Title
            ,[HireDate] = @HireDate
            ,[CurrentFlag] = @CurrentFlag
        WHERE [EmployeeID] = @EmployeeID;

        INSERT INTO [HumanResources].[EmployeePayHistory]
            ([EmployeeID]
            ,[RateChangeDate]
            ,[Rate]
            ,[PayFrequency])
        VALUES (@EmployeeID, @RateChangeDate, @Rate, @PayFrequency);

        COMMIT TRANSACTION;
    END TRY
    BEGIN CATCH
        -- Rollback any active or uncommittable transactions before
        -- inserting information in the ErrorLog
        IF @@TRANCOUNT > 0
            BEGIN
                ROLLBACK TRANSACTION;
            END

        EXECUTE [dbo].[uspLogError];
    END
END
```

## Routine Definition

```
END CATCH;  
END;
```

## Dependent Objects

Dependent Obj	Dependent Obj Type
uspLogError	SQL_STORED_PROCEDURE
Employee	USER_TABLE
EmployeePayHistory	USER_TABLE

## uspUpdateEmployeeLogin

Definition

## Routine Definition

```
CREATE PROCEDURE [HumanResources].[uspUpdateEmployeeLogin]
    @EmployeeID [int],
    @ManagerID [int],
    @LoginID [nvarchar](256),
    @Title [nvarchar](50),
    @HireDate [datetime],
    @CurrentFlag [dbo].[Flag]
WITH EXECUTE AS CALLER
AS
BEGIN
    SET NOCOUNT ON;

    BEGIN TRY
        UPDATE [HumanResources].[Employee]
        SET [ManagerID] = @ManagerID
            ,[LoginID] = @LoginID
            ,[Title] = @Title
            ,[HireDate] = @HireDate
            ,[CurrentFlag] = @CurrentFlag
        WHERE [EmployeeID] = @EmployeeID;
    END TRY
    BEGIN CATCH
        EXECUTE [dbo].[uspLogError];
    END CATCH;
END;
```

### Dependent Objects

Dependent Obj	Dependent Obj Type
uspLogError	SQL_STORED_PROCEDURE
Employee	USER_TABLE

## uspUpdateEmployeePersonallInfo

### Definition

#### Routine Definition

```
CREATE PROCEDURE [HumanResources].[uspUpdateEmployeePersonallInfo]
    @EmployeeID [int],
    @NationalIDNumber [nvarchar](15),
    @BirthDate [datetime],
    @MaritalStatus [nchar](1),
    @Gender [nchar](1)
WITH EXECUTE AS CALLER
AS
BEGIN
    SET NOCOUNT ON;

    BEGIN TRY
        UPDATE [HumanResources].[Employee]
        SET [NationalIDNumber] = @NationalIDNumber
            ,[BirthDate] = @BirthDate
            ,[MaritalStatus] = @MaritalStatus
            ,[Gender] = @Gender
        WHERE [EmployeeID] = @EmployeeID;
    END TRY
    BEGIN CATCH
        EXECUTE [dbo].[uspLogError];
    END CATCH;
END;
```

### Dependent Objects

Dependent Obj	Dependent Obj Type
uspLogError	SQL_STORED_PROCEDURE
Employee	USER_TABLE



## Functions

Function Name	Function Schema
ufnLeadingZeros	dbo
ufnGetAccountingStartDate	dbo
ufnGetAccountingEndDate	dbo
ufnGetContactInformation	dbo
ufnGetProductDealerPrice	dbo
ufnGetProductListPrice	dbo
ufnGetProductStandardCost	dbo
ufnGetStock	dbo
ufnGetDocumentStatusText	dbo
ufnGetPurchaseOrderStatusText	dbo
ufnGetSalesOrderStatusText	dbo

### ufnLeadingZeros

Definition

### Routine Definition

```
CREATE FUNCTION [dbo].[ufnLeadingZeros](
    @Value int
)
RETURNS varchar(8)
WITH SCHEMABINDING
AS
BEGIN
    DECLARE @ReturnValue varchar(8);

    SET @ReturnValue = CONVERT(varchar(8), @Value);
    SET @ReturnValue = REPLICATE('0', 8 - DATALENGTH(@ReturnValue)) + @ReturnValue;

    RETURN (@ReturnValue);
END;
```

### Dependent Objects

Dependent Obj	Dependent Obj Type
Customer	USER_TABLE

### ufnGetAccountingStartDate

#### Definition

### Routine Definition

```
CREATE FUNCTION [dbo].[ufnGetAccountingStartDate]()
RETURNS [datetime]
AS
BEGIN
    RETURN CONVERT(datetime, '20030701', 112);
END;
```

Dependent Objects

---None---

## **ufnGetAccountingEndDate**

Definition

### Routine Definition

```
CREATE FUNCTION [dbo].[ufnGetAccountingEndDate]()  
RETURNS [datetime]  
AS  
BEGIN  
    RETURN DATEADD(millisecond, -2, CONVERT(datetime, '20040701', 112));  
END;
```

Dependent Objects

---None---

## **ufnGetContactInformation**

Definition

## Routine Definition

```
CREATE FUNCTION [dbo].[ufnGetContactInformation](@ContactID int)
RETURNS @retContactInformation TABLE
(
  -- Columns returned by the function
  [ContactID] int PRIMARY KEY NOT NULL,
  [FirstName] [nvarchar](50) NULL,
  [LastName] [nvarchar](50) NULL,
  [JobTitle] [nvarchar](50) NULL,
  [ContactType] [nvarchar](50) NULL
)
AS
-- Returns the first name, last name, job title and contact type for the specified contact.
BEGIN
  DECLARE
    @FirstName [nvarchar](50),
    @LastName [nvarchar](50),
    @JobTitle [nvarchar](50),
    @ContactType [nvarchar](50);

  -- Get common contact information
  SELECT
    @ContactID = ContactID,
    @FirstName = FirstName,
    @LastName = LastName
  FROM [Person].[Contact]
  WHERE [ContactID] = @ContactID;

  SET @JobTitle =
    CASE
      -- Check for employee
      WHEN EXISTS(SELECT * FROM [HumanResources].[Employee] e
        WHERE e.[ContactID] = @ContactID)
      THEN (SELECT [Title]
        FROM [HumanResources].[Employee]
        WHERE [ContactID] = @ContactID)

      -- Check for vendor
      WHEN EXISTS(SELECT * FROM [Purchasing].[VendorContact] vc
        INNER JOIN [Person].[ContactType] ct
        ON vc.[ContactTypeID] = ct.[ContactTypeID])
```

## Routine Definition

```
WHERE vc.[ContactID] = @ContactID)
THEN (SELECT ct.[Name]
      FROM [Purchasing].[VendorContact] vc
      INNER JOIN [Person].[ContactType] ct
      ON vc.[ContactTypeID] = ct.[ContactTypeID]
      WHERE vc.[ContactID] = @ContactID)

-- Check for store
WHEN EXISTS(SELECT * FROM [Sales].[StoreContact] sc
           INNER JOIN [Person].[ContactType] ct
           ON sc.[ContactTypeID] = ct.[ContactTypeID]
           WHERE sc.[ContactID] = @ContactID)
THEN (SELECT ct.[Name]
      FROM [Sales].[StoreContact] sc
      INNER JOIN [Person].[ContactType] ct
      ON sc.[ContactTypeID] = ct.[ContactTypeID]
      WHERE [ContactID] = @ContactID)

ELSE NULL
END;

SET @ContactType =
CASE
-- Check for employee
WHEN EXISTS(SELECT * FROM [HumanResources].[Employee] e
           WHERE e.[ContactID] = @ContactID)
THEN 'Employee'

-- Check for vendor
WHEN EXISTS(SELECT * FROM [Purchasing].[VendorContact] vc
           INNER JOIN [Person].[ContactType] ct
           ON vc.[ContactTypeID] = ct.[ContactTypeID]
           WHERE vc.[ContactID] = @ContactID)
THEN 'Vendor Contact'

-- Check for store
WHEN EXISTS(SELECT * FROM [Sales].[StoreContact] sc
           INNER JOIN [Person].[ContactType] ct
           ON sc.[ContactTypeID] = ct.[ContactTypeID]
           WHERE sc.[ContactID] = @ContactID)
THEN 'Store Contact'
```

## Routine Definition

```
-- Check for individual consumer
WHEN EXISTS(SELECT * FROM [Sales].[Individual] i
  WHERE i.[ContactID] = @ContactID)
  THEN 'Consumer'
END;

-- Return the information to the caller
IF @ContactID IS NOT NULL
BEGIN
  INSERT @retContactInformation
  SELECT @ContactID, @FirstName, @LastName, @JobTitle, @ContactType;
END;

RETURN;
END;
```

### Dependent Objects

Dependent Obj	Dependent Obj Type
Contact	USER_TABLE
ContactType	USER_TABLE
Employee	USER_TABLE
Individual	USER_TABLE
StoreContact	USER_TABLE
VendorContact	USER_TABLE

## ufnGetProductDealerPrice

Definition

## Routine Definition

```
CREATE FUNCTION [dbo].[ufnGetProductDealerPrice](@ProductID [int], @OrderDate [datetime])
RETURNS [money]
AS
-- Returns the dealer price for the product on a specific date.
BEGIN
    DECLARE @DealerPrice money;
    DECLARE @DealerDiscount money;

    SET @DealerDiscount = 0.60 -- 60% of list price

    SELECT @DealerPrice = plph.[ListPrice] * @DealerDiscount
    FROM [Production].[Product] p
        INNER JOIN [Production].[ProductListPriceHistory] plph
            ON p.[ProductID] = plph.[ProductID]
            AND p.[ProductID] = @ProductID
            AND @OrderDate BETWEEN plph.[StartDate] AND COALESCE(plph.[EndDate], CONVERT(datetime, '99991231', 112)); -- Make sure
we get all the prices!

    RETURN @DealerPrice;
END;
```

### Dependent Objects

Dependent Obj	Dependent Obj Type
Product	USER_TABLE
ProductListPriceHistory	USER_TABLE

## ufnGetProductListPrice

### Definition

## Routine Definition

```
CREATE FUNCTION [dbo].[ufnGetProductListPrice](@ProductID [int], @OrderDate [datetime])
RETURNS [money]
AS
BEGIN
    DECLARE @ListPrice money;

    SELECT @ListPrice = plph.[ListPrice]
    FROM [Production].[Product] p
    INNER JOIN [Production].[ProductListPriceHistory] plph
    ON p.[ProductID] = plph.[ProductID]
    AND p.[ProductID] = @ProductID
    AND @OrderDate BETWEEN plph.[StartDate] AND COALESCE(plph.[EndDate], CONVERT(datetime, '99991231', 112)); -- Make sure
we get all the prices!

    RETURN @ListPrice;
END;
```

### Dependent Objects

Dependent Obj	Dependent Obj Type
Product	USER_TABLE
ProductListPriceHistory	USER_TABLE

### **ufnGetProductStandardCost**

#### Definition



## Routine Definition

```
CREATE FUNCTION [dbo].[ufnGetProductStandardCost](@ProductID [int], @OrderDate [datetime])
RETURNS [money]
AS
-- Returns the standard cost for the product on a specific date.
BEGIN
    DECLARE @StandardCost money;

    SELECT @StandardCost = pch.[StandardCost]
    FROM [Production].[Product] p
        INNER JOIN [Production].[ProductCostHistory] pch
            ON p.[ProductID] = pch.[ProductID]
            AND p.[ProductID] = @ProductID
            AND @OrderDate BETWEEN pch.[StartDate] AND COALESCE(pch.[EndDate], CONVERT(datetime, '99991231', 112)); -- Make sure
we get all the prices!

    RETURN @StandardCost;
END;
```

### Dependent Objects

Dependent Obj	Dependent Obj Type
Product	USER_TABLE
ProductCostHistory	USER_TABLE

### ufnGetStock

#### Definition

## Routine Definition

```
CREATE FUNCTION [dbo].[ufnGetStock](@ProductID [int])
RETURNS [int]
AS
-- Returns the stock level for the product. This function is used internally only
BEGIN
    DECLARE @ret int;

    SELECT @ret = SUM(p.[Quantity])
    FROM [Production].[ProductInventory] p
    WHERE p.[ProductID] = @ProductID
        AND p.[LocationID] = '6'; -- Only look at inventory in the misc storage

    IF (@ret IS NULL)
        SET @ret = 0

    RETURN @ret
END;
```

### Dependent Objects

Dependent Obj	Dependent Obj Type
ProductInventory	USER_TABLE

## **ufnGetDocumentStatusText**

Definition

## Routine Definition

```
CREATE FUNCTION [dbo].[ufnGetDocumentStatusText](@Status [tinyint])
RETURNS [nvarchar](16)
AS
-- Returns the sales order status text representation for the status value.
BEGIN
    DECLARE @ret [nvarchar](16);

    SET @ret =
        CASE @Status
            WHEN 1 THEN N'Pending approval'
            WHEN 2 THEN N'Approved'
            WHEN 3 THEN N'Obsolete'
            ELSE N'** Invalid **'
        END;

    RETURN @ret
END;
```

Dependent Objects

---None---

## **ufnGetPurchaseOrderStatusText**

Definition

## Routine Definition

```
CREATE FUNCTION [dbo].[ufnGetPurchaseOrderStatusText](@Status [tinyint])
RETURNS [nvarchar](15)
AS
-- Returns the sales order status text representation for the status value.
BEGIN
    DECLARE @ret [nvarchar](15);

    SET @ret =
        CASE @Status
            WHEN 1 THEN 'Pending'
            WHEN 2 THEN 'Approved'
            WHEN 3 THEN 'Rejected'
            WHEN 4 THEN 'Complete'
            ELSE '** Invalid **'
        END;

    RETURN @ret
END;
```

Dependent Objects

---None---

**ufnGetSalesOrderStatusText**

Definition

## Routine Definition

```
CREATE FUNCTION [dbo].[ufnGetSalesOrderStatusText](@Status [tinyint])
RETURNS [nvarchar](15)
AS
-- Returns the sales order status text representation for the status value.
BEGIN
    DECLARE @ret [nvarchar](15);

    SET @ret =
        CASE @Status
            WHEN 1 THEN 'In process'
            WHEN 2 THEN 'Approved'
            WHEN 3 THEN 'Backordered'
            WHEN 4 THEN 'Rejected'
            WHEN 5 THEN 'Shipped'
            WHEN 6 THEN 'Cancelled'
            ELSE '** Invalid **'
        END;

    RETURN @ret
END;
```

## Dependent Objects

---None---

## Indexes

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
Address	AK_Address_rowguid	rowguid	NONCLUSTERED	true	false	false	false	false	false

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
Address	IX_Address_AddressLine1_AddressLine2_City_StateProvinceID_PostalCode	AddressLine1	NONCLUSTERED	true	false	false	false	false	false
Address	IX_Address_AddressLine1_AddressLine2_City_StateProvinceID_PostalCode	AddressLine2	NONCLUSTERED	true	false	false	false	false	false
Address	IX_Address_AddressLine1_AddressLine2_City_StateProvinceID_PostalCode	City	NONCLUSTERED	true	false	false	false	false	false
Address	IX_Address_AddressLine1_AddressLine2_City_StateProvinceID_PostalCode	StateProvinceID	NONCLUSTERED	true	false	false	false	false	false
Address	IX_Address_AddressLine1_AddressLine2_City_StateProvinceID_PostalCode	PostalCode	NONCLUSTERED	true	false	false	false	false	false
Address	IX_Address_StateProvinceID	StateProvinceID	NONCLUSTERED	false	false	false	false	false	false
Address	PK_Address_AddressID	AddressID	CLUSTERED	true	true	false	false	false	false
AddressType	AK_AddressType_Name	Name	NONCLUSTERED	true	false	false	false	false	false

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
AddressType	AK_AddressType_rowguid	rowguid	NONCLUSTERED	true	false	false	false	false	false
AddressType	PK_AddressType_AddressTypeID	AddressTypeID	CLUSTERED	true	true	false	false	false	false
AWBuildVersion	PK_AWBuildVersion_SystemInformationID	SystemInformationID	CLUSTERED	true	true	false	false	false	false
BillOfMaterials	AK_BillOfMaterials_ProductAssemblyID_ComponentID_StartDate	ProductAssemblyID	CLUSTERED	true	false	false	false	false	false
BillOfMaterials	AK_BillOfMaterials_ProductAssemblyID_ComponentID_StartDate	ComponentID	CLUSTERED	true	false	false	false	false	false
BillOfMaterials	AK_BillOfMaterials_ProductAssemblyID_ComponentID_StartDate	StartDate	CLUSTERED	true	false	false	false	false	false
BillOfMaterials	IX_BillOfMaterials_UnitMeasureCode	UnitMeasureCode	NONCLUSTERED	false	false	false	false	false	false
BillOfMaterials	PK_BillOfMaterials_BillOfMaterialsID	BillOfMaterialsID	NONCLUSTERED	true	true	false	false	false	false
Contact	AK_Contact_rowguid	rowguid	NONCLUSTERED	true	false	false	false	false	false
Contact	IX_Contact_EmailAddress	EmailAddress	NONCLUSTERED	false	false	false	false	false	false
Contact	PK_Contact_ContactID	ContactID	CLUSTERED	true	true	false	false	false	false

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
Contact	PXML_Contact_AddContact	AdditionalContactInfo	XML	false	false	false	false	false	false
ContactCreditCard	PK_ContactCreditCard_ContactID_CreditCardID	ContactID	CLUSTERED	true	true	false	false	false	false
ContactCreditCard	PK_ContactCreditCard_ContactID_CreditCardID	CreditCardID	CLUSTERED	true	true	false	false	false	false
ContactType	AK_ContactType_Name	Name	NONCLUSTERED	true	false	false	false	false	false
ContactType	PK_ContactType_ContactTypeID	ContactTypeID	CLUSTERED	true	true	false	false	false	false
CountryRegion	AK_CountryRegion_Name	Name	NONCLUSTERED	true	false	false	false	false	false
CountryRegion	PK_CountryRegion_CountryRegionCode	CountryRegionCode	CLUSTERED	true	true	false	false	false	false
CountryRegionCurrency	IX_CountryRegionCurrency_CurrencyCode	CurrencyCode	NONCLUSTERED	false	false	false	false	false	false
CountryRegionCurrency	PK_CountryRegionCurrency_CountryRegionCode_CurrencyCode	CountryRegionCode	CLUSTERED	true	true	false	false	false	false
CountryRegionCurrency	PK_CountryRegionCurrency_CountryRegionCode_CurrencyCode	CurrencyCode	CLUSTERED	true	true	false	false	false	false
CreditCard	AK_CreditCard_CardNumber	CardNumber	NONCLUSTERED	true	false	false	false	false	false



Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
CreditCard	PK_CreditCard_CreditCardID	CreditCardID	CLUSTERED	true	true	false	false	false	false
Culture	AK_Culture_Name	Name	NONCLUSTERED	true	false	false	false	false	false
Culture	PK_Culture_CultureID	CultureID	CLUSTERED	true	true	false	false	false	false
Currency	AK_Currency_Name	Name	NONCLUSTERED	true	false	false	false	false	false
Currency	PK_Currency_CurrencyCode	CurrencyCode	CLUSTERED	true	true	false	false	false	false
CurrencyRate	AK_CurrencyRate_CurrencyRateDate_FromCurrencyCode_ToCurrencyCode	CurrencyRateDate	NONCLUSTERED	true	false	false	false	false	false
CurrencyRate	AK_CurrencyRate_CurrencyRateDate_FromCurrencyCode_ToCurrencyCode	FromCurrencyCode	NONCLUSTERED	true	false	false	false	false	false
CurrencyRate	AK_CurrencyRate_CurrencyRateDate_FromCurrencyCode_ToCurrencyCode	ToCurrencyCode	NONCLUSTERED	true	false	false	false	false	false
CurrencyRate	PK_CurrencyRate_CurrencyRateID	CurrencyRateID	CLUSTERED	true	true	false	false	false	false
Customer	AK_Customer_AccountNumber	AccountNumber	NONCLUSTERED	true	false	false	false	false	false
Customer	AK_Customer_rowguid	rowguid	NONCLUSTERED	true	false	false	false	false	false

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
Customer	IX_Customer_TerritoryID	TerritoryID	NONCLUSTERED	false	false	false	false	false	false
Customer	PK_Customer_CustomerID	CustomerID	CLUSTERED	true	true	false	false	false	false
CustomerAddress	AK_CustomerAddress_rowguid	rowguid	NONCLUSTERED	true	false	false	false	false	false
CustomerAddress	PK_CustomerAddress_CustomerID_AddressID	CustomerID	CLUSTERED	true	true	false	false	false	false
CustomerAddress	PK_CustomerAddress_CustomerID_AddressID	AddressID	CLUSTERED	true	true	false	false	false	false
DatabaseLog	PK_DatabaseLog_DatabaseLogID	DatabaseLogID	NONCLUSTERED	true	true	false	false	false	false
Department	AK_Department_Name	Name	NONCLUSTERED	true	false	false	false	false	false
Department	PK_Department_DepartmentID	DepartmentID	CLUSTERED	true	true	false	false	false	false
Document	AK_Document_FileName_Revision	FileName	NONCLUSTERED	true	false	false	false	false	false
Document	AK_Document_FileName_Revision	Revision	NONCLUSTERED	true	false	false	false	false	false
Document	PK_Document_DocumentID	DocumentID	CLUSTERED	true	true	false	false	false	false
Employee	AK_Employee_LoginID	LoginID	NONCLUSTERED	true	false	false	false	false	false

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
Employee	AK_Employee_NationalIDNumber	NationalIDNumber	NONCLUSTERED	true	false	false	false	false	false
Employee	AK_Employee_rowguid	rowguid	NONCLUSTERED	true	false	false	false	false	false
Employee	IX_Employee_ManagerID	ManagerID	NONCLUSTERED	false	false	false	false	false	false
Employee	PK_Employee_EmployeeID	EmployeeID	CLUSTERED	true	true	false	false	false	false
EmployeeAddress	AK_EmployeeAddress_rowguid	rowguid	NONCLUSTERED	true	false	false	false	false	false
EmployeeAddress	PK_EmployeeAddress_EmployeeID_AddressID	EmployeeID	CLUSTERED	true	true	false	false	false	false
EmployeeAddress	PK_EmployeeAddress_EmployeeID_AddressID	AddressID	CLUSTERED	true	true	false	false	false	false
EmployeeDepartmentHistory	IX_EmployeeDepartmentHistory_DepartmentID	DepartmentID	NONCLUSTERED	false	false	false	false	false	false
EmployeeDepartmentHistory	IX_EmployeeDepartmentHistory_ShiftID	ShiftID	NONCLUSTERED	false	false	false	false	false	false
EmployeeDepartmentHistory	PK_EmployeeDepartmentHistory_EmployeeID_StartDate_DepartmentID	EmployeeID	CLUSTERED	true	true	false	false	false	false



Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
Individual	XMLPROPERTY_Individual_Demographics	Demographics	XML	false	false	false	false	false	false
Individual	XMLVALUE_Individual_Demographics	Demographics	XML	false	false	false	false	false	false
JobCandidate	IX_JobCandidate_EmployeeID	EmployeeID	NONCLUSTERED	false	false	false	false	false	false
JobCandidate	PK_JobCandidate_JobCandidateID	JobCandidateID	CLUSTERED	true	true	false	false	false	false
Location	AK_Location_Name	Name	NONCLUSTERED	true	false	false	false	false	false
Location	PK_Location_LocationID	LocationID	CLUSTERED	true	true	false	false	false	false
Product	AK_Product_Name	Name	NONCLUSTERED	true	false	false	false	false	false
Product	AK_Product_ProductNumber	ProductNumber	NONCLUSTERED	true	false	false	false	false	false
Product	AK_Product_rowguid	rowguid	NONCLUSTERED	true	false	false	false	false	false
Product	PK_Product_ProductID	ProductID	CLUSTERED	true	true	false	false	false	false
ProductCategory	AK_ProductCategory_Name	Name	NONCLUSTERED	true	false	false	false	false	false
ProductCategory	AK_ProductCategory_rowguid	rowguid	NONCLUSTERED	true	false	false	false	false	false
ProductCategory	PK_ProductCategory_ProductCategoryID	ProductCategoryID	CLUSTERED	true	true	false	false	false	false
ProductCostHistory	PK_ProductCostHistory_ProductID_StartDate	ProductID	CLUSTERED	true	true	false	false	false	false

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
ProductCostHistory	PK_ProductCostHistory_ProductID_StartDate	StartDate	CLUSTERED	true	true	false	false	false	false
ProductDescription	AK_ProductDescription_rowguid	rowguid	NONCLUSTERED	true	false	false	false	false	false
ProductDescription	PK_ProductDescription_ProductDescriptionID	ProductDescriptionID	CLUSTERED	true	true	false	false	false	false
ProductDocument	PK_ProductDocument_ProductID_DocumentID	ProductID	CLUSTERED	true	true	false	false	false	false
ProductDocument	PK_ProductDocument_ProductID_DocumentID	DocumentID	CLUSTERED	true	true	false	false	false	false
ProductInventory	PK_ProductInventory_ProductID_LocationID	ProductID	CLUSTERED	true	true	false	false	false	false
ProductInventory	PK_ProductInventory_ProductID_LocationID	LocationID	CLUSTERED	true	true	false	false	false	false
ProductListPriceHistory	PK_ProductListPriceHistory_ProductID_StartDate	ProductID	CLUSTERED	true	true	false	false	false	false
ProductListPriceHistory	PK_ProductListPriceHistory_ProductID_StartDate	StartDate	CLUSTERED	true	true	false	false	false	false
ProductModel	AK_ProductModel_Name	Name	NONCLUSTERED	true	false	false	false	false	false

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
ProductModel	AK_ProductModel_rowguid	rowguid	NONCLUSTERED	true	false	false	false	false	false
ProductModel	PK_ProductModel_ProductModelID	ProductModelID	CLUSTERED	true	true	false	false	false	false
ProductModel	PXML_ProductModel_CatalogDescription	CatalogDescription	XML	false	false	false	false	false	false
ProductModel	PXML_ProductModel_Instructions	Instructions	XML	false	false	false	false	false	false
ProductModelIllustration	PK_ProductModelIllustration_ProductModelID_IllustrationID	ProductModelID	CLUSTERED	true	true	false	false	false	false
ProductModelIllustration	PK_ProductModelIllustration_ProductModelID_IllustrationID	IllustrationID	CLUSTERED	true	true	false	false	false	false
ProductModelProductDescriptionCulture	PK_ProductModelProductDescriptionCulture_ProductModelID_ProductDescriptionID_CultureID	ProductModelID	CLUSTERED	true	true	false	false	false	false
ProductModelProductDescriptionCulture	PK_ProductModelProductDescriptionCulture_ProductModelID_ProductDescriptionID_CultureID	ProductDescriptionID	CLUSTERED	true	true	false	false	false	false

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
ProductModelProductDescriptionCulture	PK_ProductModelProductDescriptionCulture_ProductModelID_ProductDescriptionID_CultureID	CultureID	CLUSTERED	true	true	false	false	false	false
ProductPhoto	PK_ProductPhoto_ProductPhotoID	ProductPhotoID	CLUSTERED	true	true	false	false	false	false
ProductProductPhoto	PK_ProductProductPhoto_ProductID_ProductPhotoID	ProductID	NONCLUSTERED	true	true	false	false	false	false
ProductProductPhoto	PK_ProductProductPhoto_ProductID_ProductPhotoID	ProductPhotoID	NONCLUSTERED	true	true	false	false	false	false
ProductReview	IX_ProductReview_ProductID_Name	ProductID	NONCLUSTERED	false	false	false	false	false	false
ProductReview	IX_ProductReview_ProductID_Name	ReviewerName	NONCLUSTERED	false	false	false	false	false	false
ProductReview	IX_ProductReview_ProductID_Name	Comments	NONCLUSTERED	false	false	false	false	false	false
ProductReview	PK_ProductReview_ProductReviewID	ProductReviewID	CLUSTERED	true	true	false	false	false	false
ProductSubcategory	AK_ProductSubcategory_Name	Name	NONCLUSTERED	true	false	false	false	false	false
ProductSubcategory	AK_ProductSubcategory_rowguid	rowguid	NONCLUSTERED	true	false	false	false	false	false





Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
PurchaseOrderHeader	PK_PurchaseOrderHeader_PurchaseOrderID	PurchaseOrderID	CLUSTERED	true	true	false	false	false	false
queue_messages_1977058079	queue_clustered_index	status	CLUSTERED	true	false	false	false	false	false
queue_messages_1977058079	queue_clustered_index	queuing_order	CLUSTERED	true	false	false	false	false	false
queue_messages_1977058079	queue_clustered_index	conversation_group_id	CLUSTERED	true	false	false	false	false	false
queue_messages_1977058079	queue_clustered_index	conversation_handle	CLUSTERED	true	false	false	false	false	false
queue_messages_1977058079	queue_secondary_index	status	NONCLUSTERED	true	false	false	false	false	false
queue_messages_1977058079	queue_secondary_index	priority	NONCLUSTERED	true	false	false	false	false	false
queue_messages_1977058079	queue_secondary_index	queuing_order	NONCLUSTERED	true	false	false	false	false	false
queue_messages_1977058079	queue_secondary_index	conversation_group_id	NONCLUSTERED	true	false	false	false	false	false
queue_messages_1977058079	queue_secondary_index	conversation_handle	NONCLUSTERED	true	false	false	false	false	false
queue_messages_1977058079	queue_secondary_index	service_id	NONCLUSTERED	true	false	false	false	false	false
queue_messages_2009058193	queue_clustered_index	status	CLUSTERED	true	false	false	false	false	false
queue_messages_2009058193	queue_clustered_index	queuing_order	CLUSTERED	true	false	false	false	false	false
queue_messages_2009058193	queue_clustered_index	conversation_group_id	CLUSTERED	true	false	false	false	false	false
queue_messages_2009058193	queue_clustered_index	conversation_handle	CLUSTERED	true	false	false	false	false	false

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
queue_messages_2009058193	queue_secondary_index	status	NONCLUSTERED	true	false	false	false	false	false
queue_messages_2009058193	queue_secondary_index	priority	NONCLUSTERED	true	false	false	false	false	false
queue_messages_2009058193	queue_secondary_index	queuing_order	NONCLUSTERED	true	false	false	false	false	false
queue_messages_2009058193	queue_secondary_index	conversation_group_id	NONCLUSTERED	true	false	false	false	false	false
queue_messages_2009058193	queue_secondary_index	conversation_handle	NONCLUSTERED	true	false	false	false	false	false
queue_messages_2009058193	queue_secondary_index	service_id	NONCLUSTERED	true	false	false	false	false	false
queue_messages_2041058307	queue_clustered_index	status	CLUSTERED	true	false	false	false	false	false
queue_messages_2041058307	queue_clustered_index	queuing_order	CLUSTERED	true	false	false	false	false	false
queue_messages_2041058307	queue_clustered_index	conversation_group_id	CLUSTERED	true	false	false	false	false	false
queue_messages_2041058307	queue_clustered_index	conversation_handle	CLUSTERED	true	false	false	false	false	false
queue_messages_2041058307	queue_secondary_index	status	NONCLUSTERED	true	false	false	false	false	false
queue_messages_2041058307	queue_secondary_index	priority	NONCLUSTERED	true	false	false	false	false	false
queue_messages_2041058307	queue_secondary_index	queuing_order	NONCLUSTERED	true	false	false	false	false	false
queue_messages_2041058307	queue_secondary_index	conversation_group_id	NONCLUSTERED	true	false	false	false	false	false
queue_messages_2041058307	queue_secondary_index	conversation_handle	NONCLUSTERED	true	false	false	false	false	false

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
queue_messages_2041058307	queue_secondary_index	service_id	NONCLUSTERED	true	false	false	false	false	false
SalesOrderDetail	AK_SalesOrderDetail_rowguid	rowguid	NONCLUSTERED	true	false	false	false	false	false
SalesOrderDetail	IX_SalesOrderDetail_ProductID	ProductID	NONCLUSTERED	false	false	false	false	false	false
SalesOrderDetail	PK_SalesOrderDetail_SalesOrderID_SalesOrderDetailID	SalesOrderID	CLUSTERED	true	true	false	false	false	false
SalesOrderDetail	PK_SalesOrderDetail_SalesOrderID_SalesOrderDetailID	SalesOrderDetailID	CLUSTERED	true	true	false	false	false	false
SalesOrderHeader	AK_SalesOrderHeader_rowguid	rowguid	NONCLUSTERED	true	false	false	false	false	false
SalesOrderHeader	AK_SalesOrderHeader_SalesOrderNumber	SalesOrderNumber	NONCLUSTERED	true	false	false	false	false	false
SalesOrderHeader	IX_SalesOrderHeader_CustomerID	CustomerID	NONCLUSTERED	false	false	false	false	false	false
SalesOrderHeader	IX_SalesOrderHeader_SalesPersonID	SalesPersonID	NONCLUSTERED	false	false	false	false	false	false
SalesOrderHeader	PK_SalesOrderHeader_SalesOrderID	SalesOrderID	CLUSTERED	true	true	false	false	false	false
SalesOrderHeaderSalesReason	PK_SalesOrderHeaderSalesReason_SalesOrderID_SalesReasonID	SalesOrderID	CLUSTERED	true	true	false	false	false	false

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
SalesOrderHeaderSalesReason	PK_SalesOrderHeaderSalesReason_SalesOrderID_SalesReasonID	SalesReasonID	CLUSTERED	true	true	false	false	false	false
SalesPerson	AK_SalesPerson_rowguid	rowguid	NONCLUSTERED	true	false	false	false	false	false
SalesPerson	PK_SalesPerson_SalesPersonID	SalesPersonID	CLUSTERED	true	true	false	false	false	false
SalesPersonQuotaHistory	AK_SalesPersonQuotaHistory_rowguid	rowguid	NONCLUSTERED	true	false	false	false	false	false
SalesPersonQuotaHistory	PK_SalesPersonQuotaHistory_SalesPersonID_QuotaDate	SalesPersonID	CLUSTERED	true	true	false	false	false	false
SalesPersonQuotaHistory	PK_SalesPersonQuotaHistory_SalesPersonID_QuotaDate	QuotaDate	CLUSTERED	true	true	false	false	false	false
SalesReason	PK_SalesReason_SalesReasonID	SalesReasonID	CLUSTERED	true	true	false	false	false	false
SalesTaxRate	AK_SalesTaxRate_rowguid	rowguid	NONCLUSTERED	true	false	false	false	false	false
SalesTaxRate	AK_SalesTaxRate_StateProvinceID_TaxType	StateProvinceID	NONCLUSTERED	true	false	false	false	false	false
SalesTaxRate	AK_SalesTaxRate_StateProvinceID_TaxType	TaxType	NONCLUSTERED	true	false	false	false	false	false

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
SalesTaxRate	PK_SalesTaxRate_SalesTaxRateID	SalesTaxRateID	CLUSTERED	true	true	false	false	false	false
SalesTerritory	AK_SalesTerritory_Name	Name	NONCLUSTERED	true	false	false	false	false	false
SalesTerritory	AK_SalesTerritory_rowguid	rowguid	NONCLUSTERED	true	false	false	false	false	false
SalesTerritory	PK_SalesTerritory_TerritoryID	TerritoryID	CLUSTERED	true	true	false	false	false	false
SalesTerritoryHistory	AK_SalesTerritoryHistory_rowguid	rowguid	NONCLUSTERED	true	false	false	false	false	false
SalesTerritoryHistory	PK_SalesTerritoryHistory_SalesPersonID_StartDate_TerritoryID	SalesPersonID	CLUSTERED	true	true	false	false	false	false
SalesTerritoryHistory	PK_SalesTerritoryHistory_SalesPersonID_StartDate_TerritoryID	TerritoryID	CLUSTERED	true	true	false	false	false	false
SalesTerritoryHistory	PK_SalesTerritoryHistory_SalesPersonID_StartDate_TerritoryID	StartDate	CLUSTERED	true	true	false	false	false	false
ScrapReason	AK_ScrapReason_Name	Name	NONCLUSTERED	true	false	false	false	false	false
ScrapReason	PK_ScrapReason_ScrapReasonID	ScrapReasonID	CLUSTERED	true	true	false	false	false	false
Shift	AK_Shift_Name	Name	NONCLUSTERED	true	false	false	false	false	false
Shift	AK_Shift_StartTime_EndTime	StartTime	NONCLUSTERED	true	false	false	false	false	false



Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
SpecialOfferProduct	PK_SpecialOfferProduct_SpecialOfferID_ProductID	SpecialOfferID	CLUSTERED	true	true	false	false	false	false
SpecialOfferProduct	PK_SpecialOfferProduct_SpecialOfferID_ProductID	ProductID	CLUSTERED	true	true	false	false	false	false
StateProvince	AK_StateProvince_Name	Name	NONCLUSTERED	true	false	false	false	false	false
StateProvince	AK_StateProvince_rowguid	rowguid	NONCLUSTERED	true	false	false	false	false	false
StateProvince	AK_StateProvince_StateProvinceCode_CountryRegionCode	StateProvinceCode	NONCLUSTERED	true	false	false	false	false	false
StateProvince	AK_StateProvince_StateProvinceCode_CountryRegionCode	CountryRegionCode	NONCLUSTERED	true	false	false	false	false	false
StateProvince	PK_StateProvince_StateProvinceID	StateProvinceID	CLUSTERED	true	true	false	false	false	false
Store	AK_Store_rowguid	rowguid	NONCLUSTERED	true	false	false	false	false	false
Store	IX_Store_SalesPersonID	SalesPersonID	NONCLUSTERED	false	false	false	false	false	false
Store	PK_Store_CustomerID	CustomerID	CLUSTERED	true	true	false	false	false	false
Store	PXML_Store_Demographics	Demographics	XML	false	false	false	false	false	false
StoreContact	AK_StoreContact_rowguid	rowguid	NONCLUSTERED	true	false	false	false	false	false



Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
StoreContact	IX_StoreContact_ContactID	ContactID	NONCLUSTERED	false	false	false	false	false	false
StoreContact	IX_StoreContact_ContactTypeID	ContactTypeID	NONCLUSTERED	false	false	false	false	false	false
StoreContact	PK_StoreContact_CustomerID_ContactID	CustomerID	CLUSTERED	true	true	false	false	false	false
StoreContact	PK_StoreContact_CustomerID_ContactID	ContactID	CLUSTERED	true	true	false	false	false	false
sysallocunits	clust	auid	CLUSTERED	true	false	false	false	false	false
sysasymkeys	cl	id	CLUSTERED	true	false	false	false	false	false
sysasymkeys	nc1	name	NONCLUSTERED	true	false	false	false	false	false
sysasymkeys	nc3	thumbprint	NONCLUSTERED	true	false	false	false	false	false
sysbinobjs	clst	class	CLUSTERED	true	false	false	false	false	false
sysbinobjs	clst	id	CLUSTERED	true	false	false	false	false	false
sysbinobjs	nc1	class	NONCLUSTERED	true	false	false	false	false	false
sysbinobjs	nc1	nsid	NONCLUSTERED	true	false	false	false	false	false
sysbinobjs	nc1	name	NONCLUSTERED	true	false	false	false	false	false
sysbinsubobjs	clst	class	CLUSTERED	true	false	false	false	false	false
sysbinsubobjs	clst	idmajor	CLUSTERED	true	false	false	false	false	false

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
sysbinsubobjs	clst	subid	CLUSTERED	true	false	false	false	false	false
sysbinsubobjs	nc1	name	NONCLUSTERED	true	false	false	false	false	false
sysbinsubobjs	nc1	idmajor	NONCLUSTERED	true	false	false	false	false	false
sysbinsubobjs	nc1	class	NONCLUSTERED	true	false	false	false	false	false
syscerts	cl	id	CLUSTERED	true	false	false	false	false	false
syscerts	nc1	name	NONCLUSTERED	true	false	false	false	false	false
syscerts	nc2	issuer	NONCLUSTERED	true	false	false	false	false	false
syscerts	nc2	snum	NONCLUSTERED	true	false	false	false	false	false
syscerts	nc3	thumbprint	NONCLUSTERED	true	false	false	false	false	false
sysclsobjs	clst	class	CLUSTERED	true	false	false	false	false	false
sysclsobjs	clst	id	CLUSTERED	true	false	false	false	false	false
sysclsobjs	nc	name	NONCLUSTERED	true	false	false	false	false	false
sysclsobjs	nc	class	NONCLUSTERED	true	false	false	false	false	false
syscolpars	clst	id	CLUSTERED	true	false	false	false	false	false
syscolpars	clst	number	CLUSTERED	true	false	false	false	false	false
syscolpars	clst	colid	CLUSTERED	true	false	false	false	false	false
syscolpars	nc	name	NONCLUSTERED	true	false	false	false	false	false

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
syscolpars	nc	id	NONCLUSTERED	true	false	false	false	false	false
syscolpars	nc	number	NONCLUSTERED	true	false	false	false	false	false
sysconvgroup	clst	id	CLUSTERED	true	false	false	false	false	false
sysdbfiles	clst	dbid	CLUSTERED	true	false	false	false	false	false
sysdbfiles	clst	fileid	CLUSTERED	true	false	false	false	false	false
sysdercv	cl	diagid	CLUSTERED	true	false	false	false	false	false
sysdercv	cl	initiator	CLUSTERED	true	false	false	false	false	false
sysdesend	cl	handle	CLUSTERED	true	false	false	false	false	false
sysftinds	clst	id	CLUSTERED	true	false	false	false	false	false
sysguidrefs	cl	class	CLUSTERED	true	false	false	false	false	false
sysguidrefs	cl	id	CLUSTERED	true	false	false	false	false	false
sysguidrefs	cl	subid	CLUSTERED	true	false	false	false	false	false
sysguidrefs	nc	guid	NONCLUSTERED	true	false	false	false	false	false
sysguidrefs	nc	class	NONCLUSTERED	true	false	false	false	false	false
syslobtcolumns	clust	lobtid	CLUSTERED	true	false	false	false	false	false
syslobtcolumns	clust	lobtcolumnid	CLUSTERED	true	false	false	false	false	false
syslobts	clust	lobtid	CLUSTERED	true	false	false	false	false	false

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
sysidxstats	clst	id	CLUSTERED	true	false	false	false	false	false
sysidxstats	clst	indid	CLUSTERED	true	false	false	false	false	false
sysidxstats	nc	name	NONCLUSTERED	true	false	false	false	false	false
sysidxstats	nc	id	NONCLUSTERED	true	false	false	false	false	false
sysiscols	clst	idmajor	CLUSTERED	true	false	false	false	false	false
sysiscols	clst	idminor	CLUSTERED	true	false	false	false	false	false
sysiscols	clst	subid	CLUSTERED	true	false	false	false	false	false
sysmultiobjrefs	clst	class	CLUSTERED	true	false	false	false	false	false
sysmultiobjrefs	clst	depid	CLUSTERED	true	false	false	false	false	false
sysmultiobjrefs	clst	depsubid	CLUSTERED	true	false	false	false	false	false
sysmultiobjrefs	clst	indepid	CLUSTERED	true	false	false	false	false	false
sysmultiobjrefs	clst	indepsubid	CLUSTERED	true	false	false	false	false	false
sysmultiobjrefs	nc1	indepid	NONCLUSTERED	true	false	false	false	false	false
sysmultiobjrefs	nc1	class	NONCLUSTERED	true	false	false	false	false	false
sysmultiobjrefs	nc1	indepsubid	NONCLUSTERED	true	false	false	false	false	false
sysmultiobjrefs	nc1	depid	NONCLUSTERED	true	false	false	false	false	false
sysmultiobjrefs	nc1	depsubid	NONCLUSTERED	true	false	false	false	false	false

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
sysnsobjjs	clst	class	CLUSTERED	true	false	false	false	false	false
sysnsobjjs	clst	id	CLUSTERED	true	false	false	false	false	false
sysnsobjjs	nc	name	NONCLUSTERED	true	false	false	false	false	false
sysnsobjjs	nc	nsid	NONCLUSTERED	true	false	false	false	false	false
sysnsobjjs	nc	class	NONCLUSTERED	true	false	false	false	false	false
sysobjkeycrypts	cl	class	CLUSTERED	true	false	false	false	false	false
sysobjkeycrypts	cl	id	CLUSTERED	true	false	false	false	false	false
sysobjkeycrypts	cl	thumbprint	CLUSTERED	true	false	false	false	false	false
sysobjvalues	clst	valclass	CLUSTERED	true	false	false	false	false	false
sysobjvalues	clst	objid	CLUSTERED	true	false	false	false	false	false
sysobjvalues	clst	subobjid	CLUSTERED	true	false	false	false	false	false
sysobjvalues	clst	valnum	CLUSTERED	true	false	false	false	false	false
sysowners	clst	id	CLUSTERED	true	false	false	false	false	false
sysowners	nc1	name	NONCLUSTERED	true	false	false	false	false	false
sysowners	nc2	sid	NONCLUSTERED	true	false	false	false	false	false
sysowners	nc2	id	NONCLUSTERED	true	false	false	false	false	false
sysprivs	clust	class	CLUSTERED	true	false	false	false	false	false

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
sysprivs	clust	id	CLUSTERED	true	false	false	false	false	false
sysprivs	clust	subid	CLUSTERED	true	false	false	false	false	false
sysprivs	clust	grantee	CLUSTERED	true	false	false	false	false	false
sysprivs	clust	grantor	CLUSTERED	true	false	false	false	false	false
sysprivs	clust	type	CLUSTERED	true	false	false	false	false	false
sysqnames	clst	qid	CLUSTERED	true	false	false	false	false	false
sysqnames	clst	hash	CLUSTERED	true	false	false	false	false	false
sysqnames	clst	nid	CLUSTERED	true	false	false	false	false	false
sysqnames	nc1	nid	NONCLUSTERED	true	false	false	false	false	false
sysremsvcbinds	clst	id	CLUSTERED	true	false	false	false	false	false
sysremsvcbinds	nc1	name	NONCLUSTERED	true	false	false	false	false	false
sysremsvcbinds	nc2	scid	NONCLUSTERED	true	false	false	false	false	false
sysremsvcbinds	nc2	remsvc	NONCLUSTERED	true	false	false	false	false	false
sysrowsetcolumns	clust	rowsetid	CLUSTERED	true	false	false	false	false	false
sysrowsetcolumns	clust	rowsetcolid	CLUSTERED	true	false	false	false	false	false
sysrowsetrefs	clust	class	CLUSTERED	true	false	false	false	false	false
sysrowsetrefs	clust	objid	CLUSTERED	true	false	false	false	false	false

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
sysrowsetrefs	clust	indexid	CLUSTERED	true	false	false	false	false	false
sysrowsetrefs	clust	rowsetnum	CLUSTERED	true	false	false	false	false	false
sysrowsets	clust	rowsetid	CLUSTERED	true	false	false	false	false	false
sysrts	clst	id	CLUSTERED	true	false	false	false	false	false
sysrts	nc1	remsvc	NONCLUSTERED	true	false	false	false	false	false
sysrts	nc1	brkrinst	NONCLUSTERED	true	false	false	false	false	false
sysrts	nc1	id	NONCLUSTERED	true	false	false	false	false	false
sysrts	nc2	name	NONCLUSTERED	true	false	false	false	false	false
sysscalartypes	clst	id	CLUSTERED	true	false	false	false	false	false
sysscalartypes	nc1	schid	NONCLUSTERED	true	false	false	false	false	false
sysscalartypes	nc1	name	NONCLUSTERED	true	false	false	false	false	false
sysscalartypes	nc2	name	NONCLUSTERED	true	false	false	false	false	false
sysscalartypes	nc2	schid	NONCLUSTERED	true	false	false	false	false	false
syssschobjs	clst	id	CLUSTERED	true	false	false	false	false	false
syssschobjs	nc1	nsclass	NONCLUSTERED	true	false	false	false	false	false
syssschobjs	nc1	nsid	NONCLUSTERED	true	false	false	false	false	false
syssschobjs	nc1	name	NONCLUSTERED	true	false	false	false	false	false

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
sysschobjs	nc2	name	NONCLUSTERED	true	false	false	false	false	false
sysschobjs	nc2	nsid	NONCLUSTERED	true	false	false	false	false	false
sysschobjs	nc2	nsclass	NONCLUSTERED	true	false	false	false	false	false
sysschobjs	nc3	pid	NONCLUSTERED	false	false	false	false	false	false
sysschobjs	nc3	pclass	NONCLUSTERED	false	false	false	false	false	false
sysserefs	clust	class	CLUSTERED	true	false	false	false	false	false
sysserefs	clust	depid	CLUSTERED	true	false	false	false	false	false
sysserefs	clust	indepid	CLUSTERED	true	false	false	false	false	false
syssingleobjrefs	clst	class	CLUSTERED	true	false	false	false	false	false
syssingleobjrefs	clst	depid	CLUSTERED	true	false	false	false	false	false
syssingleobjrefs	clst	depsubid	CLUSTERED	true	false	false	false	false	false
syssingleobjrefs	nc1	indepid	NONCLUSTERED	true	false	false	false	false	false
syssingleobjrefs	nc1	class	NONCLUSTERED	true	false	false	false	false	false
syssingleobjrefs	nc1	indepsubid	NONCLUSTERED	true	false	false	false	false	false
syssingleobjrefs	nc1	depid	NONCLUSTERED	true	false	false	false	false	false
syssingleobjrefs	nc1	depsubid	NONCLUSTERED	true	false	false	false	false	false
syssqlguides	clst	id	CLUSTERED	true	false	false	false	false	false



Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
syssqlguides	nc1	name	NONCLUSTERED	true	false	false	false	false	false
syssqlguides	nc2	scopetype	NONCLUSTERED	true	false	false	false	false	false
syssqlguides	nc2	scopeid	NONCLUSTERED	true	false	false	false	false	false
syssqlguides	nc2	hash	NONCLUSTERED	true	false	false	false	false	false
syssqlguides	nc2	id	NONCLUSTERED	true	false	false	false	false	false
systypedsubobjs	clst	class	CLUSTERED	true	false	false	false	false	false
systypedsubobjs	clst	idmajor	CLUSTERED	true	false	false	false	false	false
systypedsubobjs	clst	subid	CLUSTERED	true	false	false	false	false	false
systypedsubobjs	nc	name	NONCLUSTERED	true	false	false	false	false	false
systypedsubobjs	nc	idmajor	NONCLUSTERED	true	false	false	false	false	false
systypedsubobjs	nc	class	NONCLUSTERED	true	false	false	false	false	false
sysxmitqueue	clst	dlgid	CLUSTERED	true	false	false	false	false	false
sysxmitqueue	clst	finitiator	CLUSTERED	true	false	false	false	false	false
sysxmitqueue	clst	msgsequ m	CLUSTERED	true	false	false	false	false	false
sysxmlcomponent	cl	id	CLUSTERED	true	false	false	false	false	false
sysxmlcomponent	nc1	xsdid	NONCLUSTERED	true	false	false	false	false	false
sysxmlcomponent	nc1	uriord	NONCLUSTERED	true	false	false	false	false	false



Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
TransactionHistory	IX_TransactionHistory_ReferenceOrderID_ReferenceOrderLineID	ReferenceOrderID	NONCLUSTERED	false	false	false	false	false	false
TransactionHistory	IX_TransactionHistory_ReferenceOrderID_ReferenceOrderLineID	ReferenceOrderLineID	NONCLUSTERED	false	false	false	false	false	false
TransactionHistory	PK_TransactionHistory_TransactionID	TransactionID	CLUSTERED	true	true	false	false	false	false
TransactionHistoryArchive	IX_TransactionHistoryArchive_ProductID	ProductID	NONCLUSTERED	false	false	false	false	false	false
TransactionHistoryArchive	IX_TransactionHistoryArchive_ReferenceOrderID_ReferenceOrderLineID	ReferenceOrderID	NONCLUSTERED	false	false	false	false	false	false
TransactionHistoryArchive	IX_TransactionHistoryArchive_ReferenceOrderID_ReferenceOrderLineID	ReferenceOrderLineID	NONCLUSTERED	false	false	false	false	false	false
TransactionHistoryArchive	PK_TransactionHistoryArchive_TransactionID	TransactionID	CLUSTERED	true	true	false	false	false	false
ufnGetContactInformation	PK_ufnGetContactInf__60083D91	ContactID	CLUSTERED	true	true	false	false	false	false
UnitMeasure	AK_UnitMeasure_Name	Name	NONCLUSTERED	true	false	false	false	false	false

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
UnitMeasure	PK_UnitMeasure_UnitMeasureCode	UnitMeasureCode	CLUSTERED	true	true	false	false	false	false
Vendor	AK_Vendor_AccountNumber	AccountNumber	NONCLUSTERED	true	false	false	false	false	false
Vendor	PK_Vendor_VendorID	VendorID	CLUSTERED	true	true	false	false	false	false
VendorAddress	IX_VendorAddress_AddressID	AddressID	NONCLUSTERED	false	false	false	false	false	false
VendorAddress	PK_VendorAddress_VendorID_AddressID	VendorID	CLUSTERED	true	true	false	false	false	false
VendorAddress	PK_VendorAddress_VendorID_AddressID	AddressID	CLUSTERED	true	true	false	false	false	false
VendorContact	IX_VendorContact_ContactID	ContactID	NONCLUSTERED	false	false	false	false	false	false
VendorContact	IX_VendorContact_ContactTypeID	ContactTypeID	NONCLUSTERED	false	false	false	false	false	false
VendorContact	PK_VendorContact_VendorID_ContactID	VendorID	CLUSTERED	true	true	false	false	false	false
VendorContact	PK_VendorContact_VendorID_ContactID	ContactID	CLUSTERED	true	true	false	false	false	false
vProductAndDescription	IX_vProductAndDescription	ProductID	CLUSTERED	true	false	false	false	false	false
vProductAndDescription	IX_vProductAndDescription	CultureID	CLUSTERED	true	false	false	false	false	false
vStateProvinceCountryRegion	IX_vStateProvinceCountryRegion	StateProvinceID	CLUSTERED	true	false	false	false	false	false

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
vStateProvinceCountryRegion	IX_vStateProvinceCountryRegion	CountryRegionCode	CLUSTERED	true	false	false	false	false	false
WorkOrder	IX_WorkOrder_ProductID	ProductID	NONCLUSTERED	false	false	false	false	false	false
WorkOrder	IX_WorkOrder_ScrapReasonID	ScrapReasonID	NONCLUSTERED	false	false	false	false	false	false
WorkOrder	PK_WorkOrder_WorkOrderID	WorkOrderID	CLUSTERED	true	true	false	false	false	false
WorkOrderRouting	IX_WorkOrderRouting_ProductID	ProductID	NONCLUSTERED	false	false	false	false	false	false
WorkOrderRouting	PK_WorkOrderRouting_WorkOrderID_ProductID_OperationSequence	WorkOrderID	CLUSTERED	true	true	false	false	false	false
WorkOrderRouting	PK_WorkOrderRouting_WorkOrderID_ProductID_OperationSequence	ProductID	CLUSTERED	true	true	false	false	false	false
WorkOrderRouting	PK_WorkOrderRouting_WorkOrderID_ProductID_OperationSequence	OperationSequence	CLUSTERED	true	true	false	false	false	false
xml_index_nodes_1269579561_32000	PXML_IndividualDemographics	id	CLUSTERED	true	false	false	false	false	false
xml_index_nodes_1269579561_32000	PXML_IndividualDemographics	pk1	CLUSTERED	true	false	false	false	false	false

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
xml_index_nodes_1269579561_32000	XMLPATH_Individual_Demographics	hid	NONCLUSTERED	false	false	false	false	false	false
xml_index_nodes_1269579561_32000	XMLPATH_Individual_Demographics	value	NONCLUSTERED	false	false	false	false	false	false
xml_index_nodes_1269579561_32000	XMLPROPERTY_Individual_Demographics	pk1	NONCLUSTERED	false	false	false	false	false	false
xml_index_nodes_1269579561_32000	XMLPROPERTY_Individual_Demographics	hid	NONCLUSTERED	false	false	false	false	false	false
xml_index_nodes_1269579561_32000	XMLPROPERTY_Individual_Demographics	value	NONCLUSTERED	false	false	false	false	false	false
xml_index_nodes_1269579561_32000	XMLVALUE_Individual_Demographics	value	NONCLUSTERED	false	false	false	false	false	false
xml_index_nodes_1269579561_32000	XMLVALUE_Individual_Demographics	hid	NONCLUSTERED	false	false	false	false	false	false
xml_index_nodes_2021582240_32000	PXML_ProductModel_CatalogDescription	id	CLUSTERED	true	false	false	false	false	false
xml_index_nodes_2021582240_32000	PXML_ProductModel_CatalogDescription	pk1	CLUSTERED	true	false	false	false	false	false
xml_index_nodes_2021582240_32001	PXML_ProductModel_Instructions	id	CLUSTERED	true	false	false	false	false	false
xml_index_nodes_2021582240_32001	PXML_ProductModel_Instructions	pk1	CLUSTERED	true	false	false	false	false	false
xml_index_nodes_2130106629_32000	PXML_Store_Demographics	id	CLUSTERED	true	false	false	false	false	false

Table Name	Index Name	Column Name	Index Type	Is Unique	Is Primary Key	Is Unique Constraint	Is Disabled	Is Padded	Is Hypothetical
xml_index_nodes_2130106629_32000	PXML_Store_Demographics	pk1	CLUSTERED	true	false	false	false	false	false
xml_index_nodes_309576141_32000	PXML_Contact_AddContact	id	CLUSTERED	true	false	false	false	false	false
xml_index_nodes_309576141_32000	PXML_Contact_AddContact	pk1	CLUSTERED	true	false	false	false	false	false

## Triggers

Name	Trigger Schema
dEmployee	HumanResources
iWorkOrder	Production
uWorkOrder	Production
iPurchaseOrderDetail	Purchasing
uPurchaseOrderDetail	Purchasing
uPurchaseOrderHeader	Purchasing
dVendor	Purchasing
iduSalesOrderDetail	Sales
uSalesOrderHeader	Sales
iStore	Sales
iuIndividual	Sales

## dEmployee

Trigger Body

## Text

```
CREATE TRIGGER [HumanResources].[dEmployee] ON [HumanResources].[Employee]
INSTEAD OF DELETE NOT FOR REPLICATION AS
BEGIN
    DECLARE @Count int;

    SET @Count = @@ROWCOUNT;
    IF @Count = 0
        RETURN;

    SET NOCOUNT ON;

    BEGIN
        RAISERROR
            (N'Employees cannot be deleted. They can only be marked as not current.', -- Message
            10, -- Severity.
            1); -- State.

        -- Rollback any active or uncommittable transactions
        IF @@TRANCOUNT > 0
            BEGIN
                ROLLBACK TRANSACTION;
            END
    END;
END;
```



Dependent Objects

---None---

## iWorkOrder

Trigger Body

Text

```
CREATE TRIGGER [Production].[iWorkOrder] ON [Production].[WorkOrder]
AFTER INSERT AS
BEGIN
    DECLARE @Count int;

    SET @Count = @@ROWCOUNT;
    IF @Count = 0
        RETURN;

    SET NOCOUNT ON;

    BEGIN TRY
        INSERT INTO [Production].[TransactionHistory](
            [ProductID]
            ,[ReferenceOrderID]
```

## Text

```
,[TransactionType]
,[TransactionDate]
,[Quantity]
,[ActualCost])
SELECT
inserted.[ProductID]
,inserted.[WorkOrderID]
,'W'
,GETDATE()
,inserted.[OrderQty]
,0
FROM inserted;
END TRY
BEGIN CATCH
EXECUTE [dbo].[uspPrintError];

-- Rollback any active or uncommittable transactions before
-- inserting information in the ErrorLog
IF @@TRANCOUNT > 0
BEGIN
ROLLBACK TRANSACTION;
END

EXECUTE [dbo].[uspLogError];
END CATCH;
```

Text

END;

Dependent Objects

Dependent Obj	Dependent Obj Type
uspLogError	SQL_STORED_PROCEDURE
uspPrintError	SQL_STORED_PROCEDURE
TransactionHistory	USER_TABLE

**uWorkOrder**

Trigger Body

Text

CREATE TRIGGER [Production].[uWorkOrder] ON [Production].[WorkOrder]

AFTER UPDATE AS

BEGIN

    DECLARE @Count int;

    SET @Count = @@ROWCOUNT;

    IF @Count = 0

        RETURN;

SET NOCOUNT ON;

## Text

```
BEGIN TRY
```

```
IF UPDATE([ProductID]) OR UPDATE([OrderQty])
```

```
BEGIN
```

```
INSERT INTO [Production].[TransactionHistory](
```

```
    [ProductID]
```

```
    ,[ReferenceOrderID]
```

```
    ,[TransactionType]
```

```
    ,[TransactionDate]
```

```
    ,[Quantity])
```

```
SELECT
```

```
    inserted.[ProductID]
```

```
    ,inserted.[WorkOrderID]
```

```
    ,'W'
```

```
    ,GETDATE()
```

```
    ,inserted.[OrderQty]
```

```
FROM inserted;
```

```
END;
```

```
END TRY
```

```
BEGIN CATCH
```

```
EXECUTE [dbo].[uspPrintError];
```

```
-- Rollback any active or uncommittable transactions before
```

```
-- inserting information in the ErrorLog
```

```
IF @@TRANCOUNT > 0
```

Text

```
BEGIN  
    ROLLBACK TRANSACTION;  
END  
  
EXECUTE [dbo].[uspLogError];  
END CATCH;  
END;
```

Dependent Objects

Dependent Obj	Dependent Obj Type
uspLogError	SQL_STORED_PROCEDURE
uspPrintError	SQL_STORED_PROCEDURE
TransactionHistory	USER_TABLE
WorkOrder	USER_TABLE

**iPurchaseOrderDetail**

Trigger Body

Text

```
CREATE TRIGGER [Purchasing].[iPurchaseOrderDetail] ON [Purchasing].[PurchaseOrderDetail]  
AFTER INSERT AS  
BEGIN
```

## Text

```
DECLARE @Count int;
```

```
SET @Count = @@ROWCOUNT;
```

```
IF @Count = 0
```

```
    RETURN;
```

```
SET NOCOUNT ON;
```

```
BEGIN TRY
```

```
    INSERT INTO [Production].[TransactionHistory]
```

```
        ([ProductID]
```

```
        ,[ReferenceOrderID]
```

```
        ,[ReferenceOrderLineID]
```

```
        ,[TransactionType]
```

```
        ,[TransactionDate]
```

```
        ,[Quantity]
```

```
        ,[ActualCost])
```

```
    SELECT
```

```
        inserted.[ProductID]
```

```
        ,inserted.[PurchaseOrderID]
```

```
        ,inserted.[PurchaseOrderDetailID]
```

```
        ,'P'
```

```
        ,GETDATE()
```

```
        ,inserted.[OrderQty]
```

```
        ,inserted.[UnitPrice]
```

Text

FROM inserted

INNER JOIN [Purchasing].[PurchaseOrderHeader]

ON inserted.[PurchaseOrderID] = [Purchasing].[PurchaseOrderHeader].[PurchaseOrderID];

-- Update SubTotal in PurchaseOrderHeader record. Note that this causes the

-- PurchaseOrderHeader trigger to fire which will update the RevisionNumber.

UPDATE [Purchasing].[PurchaseOrderHeader]

SET [Purchasing].[PurchaseOrderHeader].[SubTotal] =

(SELECT SUM([Purchasing].[PurchaseOrderDetail].[LineTotal])

FROM [Purchasing].[PurchaseOrderDetail]

WHERE [Purchasing].[PurchaseOrderHeader].[PurchaseOrderID] = [Purchasing].[PurchaseOrderDetail].[PurchaseOrderID])

WHERE [Purchasing].[PurchaseOrderHeader].[PurchaseOrderID] IN (SELECT inserted.[PurchaseOrderID] FROM inserted);

END TRY

BEGIN CATCH

EXECUTE [dbo].[uspPrintError];

-- Rollback any active or uncommittable transactions before

-- inserting information in the ErrorLog

IF @@TRANCOUNT > 0

BEGIN

ROLLBACK TRANSACTION;

END

EXECUTE [dbo].[uspLogError];

END CATCH;

Text

END;

Dependent Objects

Dependent Obj	Dependent Obj Type
uspLogError	SQL_STORED_PROCEDURE
uspPrintError	SQL_STORED_PROCEDURE
PurchaseOrderDetail	USER_TABLE
PurchaseOrderHeader	USER_TABLE
TransactionHistory	USER_TABLE

**uPurchaseOrderDetail**

Trigger Body

Text

```
CREATE TRIGGER [Purchasing].[uPurchaseOrderDetail] ON [Purchasing].[PurchaseOrderDetail]
AFTER UPDATE AS
BEGIN
    DECLARE @Count int;

    SET @Count = @@ROWCOUNT;
    IF @Count = 0
        RETURN;
```



## Text

```
SET NOCOUNT ON;
```

```
BEGIN TRY
```

```
IF UPDATE([ProductID]) OR UPDATE([OrderQty]) OR UPDATE([UnitPrice])
```

```
-- Insert record into TransactionHistory
```

```
BEGIN
```

```
INSERT INTO [Production].[TransactionHistory]
```

```
    ([ProductID]
```

```
    ,[ReferenceOrderID]
```

```
    ,[ReferenceOrderLineID]
```

```
    ,[TransactionType]
```

```
    ,[TransactionDate]
```

```
    ,[Quantity]
```

```
    ,[ActualCost])
```

```
SELECT
```

```
    inserted.[ProductID]
```

```
    ,inserted.[PurchaseOrderID]
```

```
    ,inserted.[PurchaseOrderDetailID]
```

```
    ,'P'
```

```
    ,GETDATE()
```

```
    ,inserted.[OrderQty]
```

```
    ,inserted.[UnitPrice]
```

```
FROM inserted
```

```
    INNER JOIN [Purchasing].[PurchaseOrderDetail]
```

## Text

```
ON inserted.[PurchaseOrderID] = [Purchasing].[PurchaseOrderDetail].[PurchaseOrderID];
```

```
-- Update SubTotal in PurchaseOrderHeader record. Note that this causes the  
-- PurchaseOrderHeader trigger to fire which will update the RevisionNumber.
```

```
UPDATE [Purchasing].[PurchaseOrderHeader]
```

```
SET [Purchasing].[PurchaseOrderHeader].[SubTotal] =
```

```
(SELECT SUM([Purchasing].[PurchaseOrderDetail].[LineTotal])
```

```
FROM [Purchasing].[PurchaseOrderDetail]
```

```
WHERE [Purchasing].[PurchaseOrderHeader].[PurchaseOrderID]
```

```
= [Purchasing].[PurchaseOrderDetail].[PurchaseOrderID])
```

```
WHERE [Purchasing].[PurchaseOrderHeader].[PurchaseOrderID]
```

```
IN (SELECT inserted.[PurchaseOrderID] FROM inserted);
```

```
UPDATE [Purchasing].[PurchaseOrderDetail]
```

```
SET [Purchasing].[PurchaseOrderDetail].[ModifiedDate] = GETDATE()
```

```
FROM inserted
```

```
WHERE inserted.[PurchaseOrderID] = [Purchasing].[PurchaseOrderDetail].[PurchaseOrderID]
```

```
AND inserted.[PurchaseOrderDetailID] = [Purchasing].[PurchaseOrderDetail].[PurchaseOrderDetailID];
```

```
END;
```

```
END TRY
```

```
BEGIN CATCH
```

```
EXECUTE [dbo].[uspPrintError];
```

```
-- Rollback any active or uncommittable transactions before
```

```
-- inserting information in the ErrorLog
```

Text

```
IF @@TRANCOUNT > 0
```

```
BEGIN
```

```
    ROLLBACK TRANSACTION;
```

```
END
```

```
EXECUTE [dbo].[uspLogError];
```

```
END CATCH;
```

```
END;
```

Dependent Objects

Dependent Obj	Dependent Obj Type
uspLogError	SQL_STORED_PROCEDURE
uspPrintError	SQL_STORED_PROCEDURE
PurchaseOrderDetail	USER_TABLE
PurchaseOrderHeader	USER_TABLE
TransactionHistory	USER_TABLE

**uPurchaseOrderHeader**

Trigger Body

Text

```
CREATE TRIGGER [Purchasing].[uPurchaseOrderHeader] ON [Purchasing].[PurchaseOrderHeader]
```

## Text

```
AFTER UPDATE AS
```

```
BEGIN
```

```
    DECLARE @Count int;
```

```
    SET @Count = @@ROWCOUNT;
```

```
    IF @Count = 0
```

```
        RETURN;
```

```
    SET NOCOUNT ON;
```

```
    BEGIN TRY
```

```
        -- Update RevisionNumber for modification of any field EXCEPT the Status.
```

```
        IF NOT UPDATE([Status])
```

```
            BEGIN
```

```
                UPDATE [Purchasing].[PurchaseOrderHeader]
```

```
                SET [Purchasing].[PurchaseOrderHeader].[RevisionNumber] =
```

```
                    [Purchasing].[PurchaseOrderHeader].[RevisionNumber] + 1
```

```
                WHERE [Purchasing].[PurchaseOrderHeader].[PurchaseOrderID] IN
```

```
                    (SELECT inserted.[PurchaseOrderID] FROM inserted);
```

```
            END;
```

```
    END TRY
```

```
    BEGIN CATCH
```

```
        EXECUTE [dbo].[uspPrintError];
```

```
        -- Rollback any active or uncommittable transactions before
```

**Text**

```
-- inserting information in the ErrorLog  
IF @@TRANCOUNT > 0  
BEGIN  
    ROLLBACK TRANSACTION;  
END  
  
EXECUTE [dbo].[uspLogError];  
END CATCH;  
END;
```

**Dependent Objects**

Dependent Obj	Dependent Obj Type
uspLogError	SQL_STORED_PROCEDURE
uspPrintError	SQL_STORED_PROCEDURE
PurchaseOrderHeader	USER_TABLE

**dVendor**

**Trigger Body**

**Text**

```
CREATE TRIGGER [Purchasing].[dVendor] ON [Purchasing].[Vendor]  
INSTEAD OF DELETE NOT FOR REPLICATION AS
```

## Text

```
BEGIN
```

```
    DECLARE @Count int;
```

```
    SET @Count = @@ROWCOUNT;
```

```
    IF @Count = 0
```

```
        RETURN;
```

```
    SET NOCOUNT ON;
```

```
BEGIN TRY
```

```
    DECLARE @DeleteCount int;
```

```
    SELECT @DeleteCount = COUNT(*) FROM deleted;
```

```
    IF @DeleteCount > 0
```

```
        BEGIN
```

```
            RAISERROR
```

```
                (N'Vendors cannot be deleted. They can only be marked as not active.', -- Message
```

```
                10, -- Severity.
```

```
                1); -- State.
```

```
        -- Rollback any active or uncommittable transactions
```

```
            IF @@TRANCOUNT > 0
```

```
                BEGIN
```

```
                    ROLLBACK TRANSACTION;
```

```
                END
```

**Text**

```
END;  
END TRY  
BEGIN CATCH  
EXECUTE [dbo].[uspPrintError];  
  
-- Rollback any active or uncommittable transactions before  
-- inserting information in the ErrorLog  
IF @@TRANCOUNT > 0  
BEGIN  
ROLLBACK TRANSACTION;  
END  
  
EXECUTE [dbo].[uspLogError];  
END CATCH;  
END;
```

**Dependent Objects**

Dependent Obj	Dependent Obj Type
uspLogError	SQL_STORED_PROCEDURE
uspPrintError	SQL_STORED_PROCEDURE

## iduSalesOrderDetail

Trigger Body

Text

```
CREATE TRIGGER [Sales].[iduSalesOrderDetail] ON [Sales].[SalesOrderDetail]
AFTER INSERT, DELETE, UPDATE AS
BEGIN
    DECLARE @Count int;

    SET @Count = @@ROWCOUNT;
    IF @Count = 0
        RETURN;

    SET NOCOUNT ON;

    BEGIN TRY
        -- If inserting or updating these columns
        IF UPDATE([ProductID]) OR UPDATE([OrderQty]) OR UPDATE([UnitPrice]) OR UPDATE([UnitPriceDiscount])
        -- Insert record into TransactionHistory
        BEGIN
            INSERT INTO [Production].[TransactionHistory]
                ([ProductID]
                ,[ReferenceOrderID]
                ,[ReferenceOrderLineID]
                ,[TransactionType]
                ,[TransactionDate]
```



Text

```
,[Quantity]
,[ActualCost])
SELECT
    inserted.[ProductID]
    ,inserted.[SalesOrderID]
    ,inserted.[SalesOrderDetailID]
    ,'S'
    ,GETDATE()
    ,inserted.[OrderQty]
    ,inserted.[UnitPrice]
FROM inserted
    INNER JOIN [Sales].[SalesOrderHeader]
    ON inserted.[SalesOrderID] = [Sales].[SalesOrderHeader].[SalesOrderID];

UPDATE [Sales].[Individual]
SET [Demographics].modify('declare default element namespace
"http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/IndividualSurvey";
replace value of (/IndividualSurvey/TotalPurchaseYTD)[1]
with data(/IndividualSurvey/TotalPurchaseYTD)[1] + sql:column ("inserted.LineTotal")')
FROM inserted
    INNER JOIN [Sales].[SalesOrderHeader]
    ON inserted.[SalesOrderID] = [Sales].[SalesOrderHeader].[SalesOrderID]
WHERE [Sales].[SalesOrderHeader].[CustomerID] = [Sales].[Individual].[CustomerID];
END;
```

## Text

```
-- Update SubTotal in SalesOrderHeader record. Note that this causes the
-- SalesOrderHeader trigger to fire which will update the RevisionNumber.
UPDATE [Sales].[SalesOrderHeader]
SET [Sales].[SalesOrderHeader].[SubTotal] =
    (SELECT SUM([Sales].[SalesOrderDetail].[LineTotal])
     FROM [Sales].[SalesOrderDetail]
     WHERE [Sales].[SalesOrderHeader].[SalesOrderID] = [Sales].[SalesOrderDetail].[SalesOrderID])
WHERE [Sales].[SalesOrderHeader].[SalesOrderID] IN (SELECT inserted.[SalesOrderID] FROM inserted);

UPDATE [Sales].[Individual]
SET [Demographics].modify('declare default element namespace
    "http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/IndividualSurvey";
    replace value of (/IndividualSurvey/TotalPurchaseYTD)[1]
    with data(/IndividualSurvey/TotalPurchaseYTD)[1] - sql:column("deleted.LineTotal")')
FROM deleted
    INNER JOIN [Sales].[SalesOrderHeader]
        ON deleted.[SalesOrderID] = [Sales].[SalesOrderHeader].[SalesOrderID]
    WHERE [Sales].[SalesOrderHeader].[CustomerID] = [Sales].[Individual].[CustomerID];
END TRY
BEGIN CATCH
    EXECUTE [dbo].[uspPrintError];

-- Rollback any active or uncommittable transactions before
-- inserting information in the ErrorLog
IF @@TRANCOUNT > 0
```

Text

```
BEGIN  
    ROLLBACK TRANSACTION;  
END  
  
EXECUTE [dbo].[uspLogError];  
END CATCH;  
END;
```

Dependent Objects

Dependent Obj	Dependent Obj Type
uspLogError	SQL_STORED_PROCEDURE
uspPrintError	SQL_STORED_PROCEDURE
Individual	USER_TABLE
SalesOrderDetail	USER_TABLE
SalesOrderHeader	USER_TABLE
TransactionHistory	USER_TABLE

**uSalesOrderHeader**

Trigger Body

Text

```
CREATE TRIGGER [Sales].[uSalesOrderHeader] ON [Sales].[SalesOrderHeader]
```

## Text

```
AFTER UPDATE NOT FOR REPLICATION AS
```

```
BEGIN
```

```
    DECLARE @Count int;
```

```
    SET @Count = @@ROWCOUNT;
```

```
    IF @Count = 0
```

```
        RETURN;
```

```
    SET NOCOUNT ON;
```

```
    BEGIN TRY
```

```
        -- Update RevisionNumber for modification of any field EXCEPT the Status.
```

```
        IF NOT UPDATE([Status])
```

```
            BEGIN
```

```
                UPDATE [Sales].[SalesOrderHeader]
```

```
                SET [Sales].[SalesOrderHeader].[RevisionNumber] =
```

```
                    [Sales].[SalesOrderHeader].[RevisionNumber] + 1
```

```
                WHERE [Sales].[SalesOrderHeader].[SalesOrderID] IN
```

```
                    (SELECT inserted.[SalesOrderID] FROM inserted);
```

```
            END;
```

```
        -- Update the SalesPerson SalesYTD when SubTotal is updated
```

```
        IF UPDATE([SubTotal])
```

```
            BEGIN
```

```
                DECLARE @StartDate datetime,
```

Text

@EndDate datetime

SET @StartDate = [dbo].[ufnGetAccountingStartDate]();

SET @EndDate = [dbo].[ufnGetAccountingEndDate]();

UPDATE [Sales].[SalesPerson]

SET [Sales].[SalesPerson].[SalesYTD] =

(SELECT SUM([Sales].[SalesOrderHeader].[SubTotal])

FROM [Sales].[SalesOrderHeader]

WHERE [Sales].[SalesPerson].[SalesPersonID] = [Sales].[SalesOrderHeader].[SalesPersonID]

AND ([Sales].[SalesOrderHeader].[Status] = 5) -- Shipped

AND [Sales].[SalesOrderHeader].[OrderDate] BETWEEN @StartDate AND @EndDate)

WHERE [Sales].[SalesPerson].[SalesPersonID]

IN (SELECT DISTINCT inserted.[SalesPersonID] FROM inserted

WHERE inserted.[OrderDate] BETWEEN @StartDate AND @EndDate);

-- Update the SalesTerritory SalesYTD when SubTotal is updated

UPDATE [Sales].[SalesTerritory]

SET [Sales].[SalesTerritory].[SalesYTD] =

(SELECT SUM([Sales].[SalesOrderHeader].[SubTotal])

FROM [Sales].[SalesOrderHeader]

WHERE [Sales].[SalesTerritory].[TerritoryID] = [Sales].[SalesOrderHeader].[TerritoryID]

AND ([Sales].[SalesOrderHeader].[Status] = 5) -- Shipped

AND [Sales].[SalesOrderHeader].[OrderDate] BETWEEN @StartDate AND @EndDate)

WHERE [Sales].[SalesTerritory].[TerritoryID]

**Text**

```
IN (SELECT DISTINCT inserted.[TerritoryID] FROM inserted
WHERE inserted.[OrderDate] BETWEEN @StartDate AND @EndDate);
END;
END TRY
BEGIN CATCH
EXECUTE [dbo].[uspPrintError];

-- Rollback any active or uncommittable transactions before
-- inserting information in the ErrorLog
IF @@TRANCOUNT > 0
BEGIN
ROLLBACK TRANSACTION;
END

EXECUTE [dbo].[uspLogError];
END CATCH;
END;
```

**Dependent Objects**

Dependent Obj	Dependent Obj Type
uspLogError	SQL_STORED_PROCEDURE
uspPrintError	SQL_STORED_PROCEDURE
SalesOrderHeader	USER_TABLE

Dependent Obj	Dependent Obj Type
SalesPerson	USER_TABLE
SalesTerritory	USER_TABLE

## iStore

Trigger Body

Text
CREATE TRIGGER [Sales].[iStore] ON [Sales].[Store]
AFTER INSERT AS
BEGIN
DECLARE @Count int;
SET @Count = @@ROWCOUNT;
IF @Count = 0
RETURN;
SET NOCOUNT ON;
BEGIN TRY
-- Only allow the Customer to be a Store OR Individual
IF EXISTS (SELECT * FROM inserted INNER JOIN [Sales].[Individual]
ON inserted.[CustomerID] = [Sales].[Individual].[CustomerID])
BEGIN

## Text

```
-- Rollback any active or uncommittable transactions
IF @@TRANCOUNT > 0
BEGIN
    ROLLBACK TRANSACTION;
END
END;
END TRY
BEGIN CATCH
    EXECUTE [dbo].[uspPrintError];

-- Rollback any active or uncommittable transactions before
-- inserting information in the ErrorLog
IF @@TRANCOUNT > 0
BEGIN
    ROLLBACK TRANSACTION;
END

EXECUTE [dbo].[uspLogError];
END CATCH;
END;
```

Dependent Objects



Dependent Obj	Dependent Obj Type
uspLogError	SQL_STORED_PROCEDURE
uspPrintError	SQL_STORED_PROCEDURE
Individual	USER_TABLE

## iuIndividual

Trigger Body

Text
CREATE TRIGGER [Sales].[iuIndividual] ON [Sales].[Individual]
AFTER INSERT, UPDATE NOT FOR REPLICATION AS
BEGIN
DECLARE @Count int;
SET @Count = @@ROWCOUNT;
IF @Count = 0
RETURN;
SET NOCOUNT ON;
-- Only allow the Customer to be a Store OR Individual
IF EXISTS (SELECT * FROM inserted INNER JOIN [Sales].[Store]
ON inserted.[CustomerID] = [Sales].[Store].[CustomerID])
BEGIN

## Text

```
-- Rollback any active or uncommittable transactions
IF @@TRANCOUNT > 0
BEGIN
    ROLLBACK TRANSACTION;
END
END;

IF UPDATE([CustomerID]) OR UPDATE([Demographics])
BEGIN
    UPDATE [Sales].[Individual]
    SET [Sales].[Individual].[Demographics] = N'<IndividualSurvey xmlns="http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/IndividualSurvey">
        <TotalPurchaseYTD>0.00</TotalPurchaseYTD>
    </IndividualSurvey>'
    FROM inserted
    WHERE [Sales].[Individual].[CustomerID] = inserted.[CustomerID]
        AND inserted.[Demographics] IS NULL;

    UPDATE [Sales].[Individual]
    SET [Demographics].modify(N'declare default element namespace "http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/IndividualSurvey";
    insert <TotalPurchaseYTD>0.00</TotalPurchaseYTD>
    as first
    into (/IndividualSurvey)[1]')
    FROM inserted
    WHERE [Sales].[Individual].[CustomerID] = inserted.[CustomerID]
```

**Text**

AND inserted.[Demographics] IS NOT NULL

AND inserted.[Demographics].exist(N'declare default element namespace

"http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/IndividualSurvey";

/IndividualSurvey/TotalPurchaseYTD') <> 1;

END;

END;

**Dependent Objects**

Dependent Obj	Dependent Obj Type
Individual	USER_TABLE
Store	USER_TABLE