
Upsizing Wizard Report

Database

Microsoft Access Database: C:\SEUA12\Code\Chaptr22\Upsize22A.accdb

SQL Server Database: UpsizeSQL

Upsizing Parameters

Table Attributes to Export

☒ **Indexes**

☒ **Validation rules**

☒ **Defaults**

☐ **Structure only, no data**

Table relationships:

Upsized using DRI

Timestamp fields added:

Some tables

Modifications to Existing Database

☐ **Attach newly created
SQL Server tables**

☐ **Save password and
user ID with attached
tables**

Client/Server Modifications

☒ **Create a new Access
client/server
application.**

☐ **Save password and
user ID with
application**

Tables

Table: Categories

Microsoft Access

SQL Server

Table Name: Categories

Categories

Attached Table Name:

Aliasing Query:

Validation Rule:

Timestamp field added to SQL Server table.

Fields

Microsoft Access	SQL Server
Field Name: CategoryID	CategoryID
Data Type: Number (Long)	int
Field Name: CategoryName	CategoryName
Data Type: Text(15)	nvarchar(15)
Field Name: Description	Description
Data Type: Memo	text
Field Name: Picture	Picture
Data Type: OLE Object	image

Indexes

Microsoft Access	SQL Server
Name: CategoryName	CategoryName
Fields: CategoryName	CategoryName
Type: Unique	Unique

Table: CustContacts

Microsoft Access

Table Name: CustContacts

Attached Table Name:

Aliasing Query:

Validation Rule:

SQL Server

CustContacts

Timestamp field added to SQL Server table.

Fields

Microsoft Access

SQL Server

Field Name: ContactID

ContactID

Data Type: Number (Long)

int

Field Name: CustomerID

CustomerID

Data Type: Text(5)

nvarchar(5)

Field Name: FirstName

FirstName

Data Type: Text(50)

nvarchar(50)

Field Name: LastName

LastName

Data Type: Text(50)

nvarchar(50)

Field Name: Dear

Dear

Data Type: Text(50)

nvarchar(50)

Field Name: Title

Title

Data Type: Text(50)

nvarchar(50)

Field Name: WorkPhone

WorkPhone

Data Type: Text(30)

nvarchar(30)

Field Name: WorkExtension

WorkExtension

Data Type: Text(20)

nvarchar(20)

Field Name: HomePhone

HomePhone

Data Type: Text(30)

nvarchar(30)

Field Name: MobilePhone

MobilePhone

Data Type: Text(30)

nvarchar(30)

Field Name: FaxNumber

FaxNumber

Data Type: Text(30)

nvarchar(30)

Field Name: EmailName

EmailName

Data Type: Text(50)

nvarchar(50)

Field Name: LastMeetingDate	LastMeetingDate
Data Type: Date/Time	datetime
Field Name: ContactTypeID	ContactTypeID
Data Type: Number (Long)	int
Field Name: ReferredBy	ReferredBy
Data Type: Text(50)	nvarchar(50)
Field Name: Notes	Notes
Data Type: Memo	text

Indexes	Microsoft Access	SQL Server
	Name: ContactTypeID	ContactTypeID
	Fields: ContactTypeID	ContactTypeID
	Type: DuplicatesOK	DuplicatesOK
	Name: CustomerID	CustomerID
	Fields: CustomerID	CustomerID
	Type: DuplicatesOK	DuplicatesOK
	Name: CustomersCustContacts	CustomersCustContacts
	Fields: CustomerID	CustomerID
	Type: DuplicatesOK	DuplicatesOK
	Name: EmailName	EmailName
	Fields: EmailName	EmailName
	Type: DuplicatesOK	DuplicatesOK
	Name: LastName	LastName
	Fields: LastName	LastName
	Type: DuplicatesOK	DuplicatesOK

Table: Customers

	Microsoft Access	SQL Server
Table Name:	Customers	Customers
Attached Table Name:		
Aliasing Query:		
Validation Rule:		

Fields	Microsoft Access	SQL Server
Field Name:	CustomerID	CustomerID
Data Type:	Text(5)	nvarchar(5)
Field Name:	CompanyName	CompanyName
Data Type:	Text(40)	nvarchar(40)
Field Name:	ContactName	ContactName
Data Type:	Text(30)	nvarchar(30)
Field Name:	ContactTitle	ContactTitle
Data Type:	Text(30)	nvarchar(30)
Field Name:	Address	Address
Data Type:	Text(60)	nvarchar(60)
Field Name:	City	City
Data Type:	Text(15)	nvarchar(15)
Field Name:	Region	Region
Data Type:	Text(15)	nvarchar(15)
Field Name:	PostalCode	PostalCode
Data Type:	Text(10)	nvarchar(10)
Field Name:	Country	Country
Data Type:	Text(15)	nvarchar(15)
Field Name:	Phone	Phone
Data Type:	Text(24)	nvarchar(24)
Field Name:	Fax	Fax
Data Type:	Text(24)	nvarchar(24)

Indexes	Microsoft Access	SQL Server
Name:	City	City
Fields:	City	City

Type:	DuplicatesOK	DuplicatesOK
Name:	CompanyName	CompanyName
Fields:	CompanyName	CompanyName
Type:	DuplicatesOK	DuplicatesOK
Name:	PostalCode	PostalCode
Fields:	PostalCode	PostalCode
Type:	DuplicatesOK	DuplicatesOK
Name:	Region	Region
Fields:	Region	Region
Type:	DuplicatesOK	DuplicatesOK

Table: Employees

	Microsoft Access	SQL Server
Table Name:	Employees	Employees
Attached Table Name:		
Aliasing Query:		
Validation Rule:		

Timestamp field added to SQL Server table.

Fields	Microsoft Access	SQL Server
Field Name:	EmployeeID	EmployeeID
Data Type:	Number (Long)	int
Field Name:	LastName	LastName
Data Type:	Text(20)	nvarchar(20)
Field Name:	FirstName	FirstName
Data Type:	Text(10)	nvarchar(10)
Field Name:	Title	Title
Data Type:	Text(30)	nvarchar(30)
Field Name:	TitleOfCourtesy	TitleOfCourtesy
Data Type:	Text(25)	nvarchar(25)
Field Name:	BirthDate	BirthDate
Data Type:	Date/Time	datetime
Validation Rule:	<Date()	ALTER TABLE [Employees] ADD CONSTRAINT [CK Employees BirthDate] CHECK (BirthDate<convert(datetime,convert(varchar ,getdate()),1),1)) EXEC sp_addextendedproperty N'MS_ConstraintText', N'Birth date can't be in the future.', N'user', N'dbo', N'table', N'Employees', N'constraint', N'CK Employees BirthDate'
Field Name:	HireDate	HireDate
Data Type:	Date/Time	datetime
Field Name:	Address	Address
Data Type:	Text(60)	nvarchar(60)
Field Name:	City	City
Data Type:	Text(15)	nvarchar(15)

Field Name: Region	Region
Data Type: Text(15)	nvarchar(15)
Field Name: PostalCode	PostalCode
Data Type: Text(10)	nvarchar(10)
Field Name: Country	Country
Data Type: Text(15)	nvarchar(15)
Field Name: HomePhone	HomePhone
Data Type: Text(24)	nvarchar(24)
Field Name: Extension	Extension
Data Type: Text(4)	nvarchar(4)
Field Name: Photo	Photo
Data Type: Text(255)	nvarchar(255)
Field Name: PhotoOLE	PhotoOLE
Data Type: OLE Object	image
Field Name: Notes	Notes
Data Type: Memo	text
Field Name: ReportsTo	ReportsTo
Data Type: Number (Long)	int

Indexes

Microsoft Access

Name: LastName
Fields: LastName
Type: DuplicatesOK

Name: PostalCode
Fields: PostalCode
Type: DuplicatesOK

SQL Server

LastName
 LastName
 DuplicatesOK

PostalCode
 PostalCode
 DuplicatesOK

Table: HRActions

	Microsoft Access	SQL Server
Table Name:	HRActions	HRActions
Attached Table Name:		
Aliasing Query:		
Validation Rule:	[EffectiveDate]>=[ScheduledDate] Or [EffectiveDate] Is Null	Not Exported
	<i>Timestamp field added to SQL Server table.</i>	

Fields	Microsoft Access	SQL Server
Field Name:	EmployeeID	EmployeeID
Data Type:	Number (Long)	int
Validation Rule:	>0	ALTER TABLE [HRActions] ADD CONSTRAINT [CK HRActions EmployeeID] CHECK (EmployeeID>0) EXEC sp_addextendedproperty N'MS_ConstraintText', N'Please enter a valid employee ID number.', N'user', N'dbo', N'table', N'HRActions', N'constraint', N'CK HRActions EmployeeID'
Field Name:	ActionType	ActionType
Data Type:	Text(50)	nvarchar(50)
Validation Rule:	In ("H","Q","Y","S","R","B","C","T")	ALTER TABLE [HRActions] ADD CONSTRAINT [CK HRActions ActionType] CHECK (ActionType In ('H','Q','Y','S','R','B','C','T')) EXEC sp_addextendedproperty N'MS_ConstraintText', N'Only H, Q, Y, S, R, B, C and T codes are valid.', N'user', N'dbo', N'table', N'HRActions', N'constraint', N'CK HRActions ActionType'
Field Name:	InitiatedBy	InitiatedBy
Data Type:	Number (Long)	int
Validation Rule:	>0	ALTER TABLE [HRActions] ADD CONSTRAINT [CK HRActions InitiatedBy] CHECK (InitiatedBy>0) EXEC sp_addextendedproperty N'MS_ConstraintText', N'Please enter a valid supervisor ID number.', N'user', N'dbo', N'table', N'HRActions', N'constraint', N'CK HRActions InitiatedBy'

Field Name: ScheduledDate	ScheduledDate
Data Type: Date/Time	datetime
Validation Rule: Between Date()-3650 And Date()+365	ALTER TABLE [HRActions] ADD CONSTRAINT [CK HRActions ScheduledDate] CHECK (ScheduledDate Between convert(datetime,convert(varchar,getdate()),1),1)-3650 And convert(datetime,convert(varchar,getdate()),1),1)+365) EXEC sp_addextendedproperty N'MS_ConstraintText', N'Scheduled dates can't be more than 10 years ago or more than one year from now.', N'user', N'dbo', N'table', N'HRActions', N'constraint', N'CK HRActions ScheduledDate'
Field Name: ApprovedBy	ApprovedBy
Data Type: Number (Long)	int
Validation Rule: >0 Or Is Null	ALTER TABLE [HRActions] ADD CONSTRAINT [CK HRActions ApprovedBy] CHECK (ApprovedBy>0 Or ApprovedBy Is Null) EXEC sp_addextendedproperty N'MS_ConstraintText', N'Enter a valid manager ID number or leave blank if not approved.', N'user', N'dbo', N'table', N'HRActions', N'constraint', N'CK HRActions ApprovedBy'
Field Name: EffectiveDate	EffectiveDate
Data Type: Date/Time	datetime
Field Name: HRRating	HRRating
Data Type: Byte	smallint
Validation Rule: Between 0 And 9 Or Is Null	ALTER TABLE [HRActions] ADD CONSTRAINT [CK HRActions HRRating] CHECK (HRRating Between 0 And 9 Or HRRating Is Null) EXEC sp_addextendedproperty N'MS_ConstraintText', N'Rating range is 0 for terminated employees, 1 to 9, or blank.', N'user', N'dbo', N'table', N'HRActions', N'constraint', N'CK HRActions HRRating'
Field Name: NewSalary	NewSalary
Data Type: Currency	money
Field Name: NewRate	NewRate
Data Type: Currency	money
Validation Rule: >5.5 Or Is Null	ALTER TABLE [HRActions] ADD CONSTRAINT [CK HRActions NewRate] CHECK (NewRate>5.5 Or NewRate Is Null) EXEC sp_addextendedproperty N'MS_ConstraintText', N'Hourly rate must be more than the minimum wage.', N'user', N'dbo', N'table', N'HRActions', N'constraint', N'CK HRActions NewRate'

Field Name: NewBonus	NewBonus
Data Type: Currency	money
Field Name: NewCommission	NewCommission
Data Type: Number (Single)	real
Validation Rule: <=0.1 Or Is Null	ALTER TABLE [HRActions] ADD CONSTRAINT [CK HRActions NewCommission] CHECK (NewCommission<=0.1 Or NewCommission Is Null) EXEC sp_addextendedproperty N'MS_ConstraintText', N'Commission rate can't exceed 10%.', N'user', N'dbo', N'table', N'HRActions', N'constraint', N'CK HRActions NewCommission'
Field Name: HRComments	HRComments
Data Type: Memo	text

Indexes	Microsoft Access	SQL Server
	Name: ActionTypeEffDate	ActionTypeEffDate
	Fields: ActionType, EffectiveDate	ActionType, EffectiveDate
	Type: DuplicatesOK	DuplicatesOK
	Name: EffectiveDate	EffectiveDate
	Fields: EffectiveDate	EffectiveDate
	Type: DuplicatesOK	DuplicatesOK
	Name: EmployeesHRActions	EmployeesHRActions
	Fields: EmployeeID	EmployeeID
	Type: DuplicatesOK	DuplicatesOK

Table: Order Details

	Microsoft Access	SQL Server
Table Name:	Order Details	Order Details
Attached Table Name:		
Aliasing Query:		
Validation Rule:		

Timestamp field added to SQL Server table.

Fields	Microsoft Access	SQL Server
Field Name:	OrderID	OrderID
Data Type:	Number (Long)	int
Field Name:	ProductID	ProductID
Data Type:	Number (Long)	int
Field Name:	UnitPrice	UnitPrice
Data Type:	Currency	money
Validation Rule:	>=0	ALTER TABLE [Order Details] ADD CONSTRAINT [CK Order Details UnitPrice] CHECK (UnitPrice>=0) EXEC sp_addextendedproperty N'MS_ConstraintText', N'You must enter a positive number.', N'user', N'dbo', N'table', N'Order Details', N'constraint', N'CK Order Details UnitPrice'
Field Name:	Quantity	Quantity
Data Type:	Number (Integer)	smallint
Validation Rule:	>0	ALTER TABLE [Order Details] ADD CONSTRAINT [CK Order Details Quantity] CHECK (Quantity>0) EXEC sp_addextendedproperty N'MS_ConstraintText', N'Quantity must be greater than 0', N'user', N'dbo', N'table', N'Order Details', N'constraint', N'CK Order Details Quantity'
Field Name:	Discount	Discount
Data Type:	Number (Single)	real
Validation Rule:	Between 0 And 1	ALTER TABLE [Order Details] ADD CONSTRAINT [CK Order Details Discount] CHECK (Discount Between 0 And 1) EXEC sp_addextendedproperty N'MS_ConstraintText', N'You must enter a value with a percent sign.', N'user', N'dbo', N'table', N'Order Details', N'constraint', N'CK Order Details Discount'

Indexes	Microsoft Access	SQL Server
Name:	OrderID	OrderID

Fields: OrderID	OrderID
Type: DuplicatesOK	DuplicatesOK
Name: OrdersOrder Details	"OrdersOrder Details"
Fields: OrderID	OrderID
Type: DuplicatesOK	DuplicatesOK
Name: ProductID	ProductID
Fields: ProductID	ProductID
Type: DuplicatesOK	DuplicatesOK
Name: ProductsOrder Details	"ProductsOrder Details"
Fields: ProductID	ProductID
Type: DuplicatesOK	DuplicatesOK

Table: Orders

	Microsoft Access	SQL Server
Table Name:	Orders	Orders
Attached Table Name:		
Aliasing Query:		
Validation Rule:		

Fields	Microsoft Access	SQL Server
Field Name:	OrderID	OrderID
Data Type:	Number (Long)	int
Field Name:	CustomerID	CustomerID
Data Type:	Text(5)	nvarchar(5)
Field Name:	EmployeeID	EmployeeID
Data Type:	Number (Long)	int
Field Name:	OrderDate	OrderDate
Data Type:	Date/Time	datetime
Field Name:	RequiredDate	RequiredDate
Data Type:	Date/Time	datetime
Field Name:	ShippedDate	ShippedDate
Data Type:	Date/Time	datetime
Field Name:	ShipVia	ShipVia
Data Type:	Number (Long)	int
Field Name:	Freight	Freight
Data Type:	Currency	money
Field Name:	ShipName	ShipName
Data Type:	Text(40)	nvarchar(40)
Field Name:	ShipAddress	ShipAddress
Data Type:	Text(60)	nvarchar(60)
Field Name:	ShipCity	ShipCity
Data Type:	Text(15)	nvarchar(15)
Field Name:	ShipRegion	ShipRegion
Data Type:	Text(15)	nvarchar(15)

Field Name: ShipPostalCode	ShipPostalCode
Data Type: Text(10)	nvarchar(10)
Field Name: ShipCountry	ShipCountry
Data Type: Text(15)	nvarchar(15)

Indexes	Microsoft Access	SQL Server
	Name: CustomerID	CustomerID
	Fields: CustomerID	CustomerID
	Type: DuplicatesOK	DuplicatesOK
	Name: CustomersOrders	CustomersOrders
	Fields: CustomerID	CustomerID
	Type: DuplicatesOK	DuplicatesOK
	Name: EmployeeID	EmployeeID
	Fields: EmployeeID	EmployeeID
	Type: DuplicatesOK	DuplicatesOK
	Name: EmployeesOrders	EmployeesOrders
	Fields: EmployeeID	EmployeeID
	Type: DuplicatesOK	DuplicatesOK
	Name: OrderDate	OrderDate
	Fields: OrderDate	OrderDate
	Type: DuplicatesOK	DuplicatesOK
	Name: ShippedDate	ShippedDate
	Fields: ShippedDate	ShippedDate
	Type: DuplicatesOK	DuplicatesOK
	Name: ShippersOrders	ShippersOrders
	Fields: ShipVia	ShipVia
	Type: DuplicatesOK	DuplicatesOK
	Name: ShipPostalCode	ShipPostalCode
	Fields: ShipPostalCode	ShipPostalCode
	Type: DuplicatesOK	DuplicatesOK

Table: Products

	Microsoft Access	SQL Server
Table Name:	Products	Products
Attached Table Name:		
Aliasing Query:		
Validation Rule:		

Fields	Microsoft Access	SQL Server
Field Name:	ProductID	ProductID
Data Type:	Number (Long)	int
Field Name:	ProductName	ProductName
Data Type:	Text(40)	nvarchar(40)
Field Name:	SupplierID	SupplierID
Data Type:	Number (Long)	int
Field Name:	CategoryID	CategoryID
Data Type:	Number (Long)	int
Field Name:	QuantityPerUnit	QuantityPerUnit
Data Type:	Text(20)	nvarchar(20)
Field Name:	UnitPrice	UnitPrice
Data Type:	Currency	money
Validation Rule:	>=0	ALTER TABLE [Products] ADD CONSTRAINT [CK Products UnitPrice] CHECK (UnitPrice>=0) EXEC sp_addextendedproperty N'MS_ConstraintText', N'You must enter a positive number.', N'user', N'dbo', N'table', N'Products', N'constraint', N'CK Products UnitPrice'
Field Name:	UnitsInStock	UnitsInStock
Data Type:	Number (Integer)	smallint
Validation Rule:	>=0	ALTER TABLE [Products] ADD CONSTRAINT [CK Products UnitsInStock] CHECK (UnitsInStock>=0) EXEC sp_addextendedproperty N'MS_ConstraintText', N'You must enter a positive number.', N'user', N'dbo', N'table', N'Products', N'constraint', N'CK Products UnitsInStock'

Field Name: UnitsOnOrder
Data Type: Number (Integer)
Validation Rule: >=0

UnitsOnOrder
smallint
ALTER TABLE [Products] ADD CONSTRAINT
[CK Products UnitsOnOrder] CHECK
(UnitsOnOrder>=0)
EXEC sp_addextendedproperty
N'MS_ConstraintText', N'You must enter a
positive number.', N'user', N'dbo', N'table',
N'Products', N'constraint', N'CK Products
UnitsOnOrder'

Field Name: ReorderLevel
Data Type: Number (Integer)
Validation Rule: >=0

ReorderLevel
smallint
ALTER TABLE [Products] ADD CONSTRAINT
[CK Products ReorderLevel] CHECK
(ReorderLevel>=0)
EXEC sp_addextendedproperty
N'MS_ConstraintText', N'You must enter a
positive number.', N'user', N'dbo', N'table',
N'Products', N'constraint', N'CK Products
ReorderLevel'

Field Name: Discontinued
Data Type: Yes/No

Discontinued
bit

Indexes

Microsoft Access

SQL Server

Name: CategoriesProducts

CategoriesProducts

Fields: CategoryID

CategoryID

Type: DuplicatesOK

DuplicatesOK

Name: CategoryID

CategoryID

Fields: CategoryID

CategoryID

Type: DuplicatesOK

DuplicatesOK

Name: ProductName

ProductName

Fields: ProductName

ProductName

Type: DuplicatesOK

DuplicatesOK

Name: SupplierID

SupplierID

Fields: SupplierID

SupplierID

Type: DuplicatesOK

DuplicatesOK

Name: SuppliersProducts

SuppliersProducts

Fields: SupplierID

SupplierID

Type: DuplicatesOK

DuplicatesOK

Table: ShipAddresses

	Microsoft Access	SQL Server
Table Name:	ShipAddresses	ShipAddresses
Attached Table Name:		
Aliasing Query:		
Validation Rule:		

Fields	Microsoft Access	SQL Server
Field Name:	CustomerID	CustomerID
Data Type:	Text(5)	nvarchar(5)
Field Name:	ShipName	ShipName
Data Type:	Text(255)	nvarchar(255)
Field Name:	ShipAddress	ShipAddress
Data Type:	Text(255)	nvarchar(255)
Field Name:	ShipCity	ShipCity
Data Type:	Text(255)	nvarchar(255)
Field Name:	ShipRegion	ShipRegion
Data Type:	Text(255)	nvarchar(255)
Field Name:	ShipPostalCode	ShipPostalCode
Data Type:	Text(255)	nvarchar(255)
Field Name:	ShipCountry	ShipCountry
Data Type:	Text(255)	nvarchar(255)

Table: Shippers

	Microsoft Access	SQL Server
Table Name:	Shippers	Shippers
Attached Table Name:		
Aliasing Query:		
Validation Rule:		

Fields	Microsoft Access	SQL Server
Field Name:	ShipperID	ShipperID
Data Type:	Number (Long)	int
Field Name:	CompanyName	CompanyName
Data Type:	Text(40)	nvarchar(40)
Field Name:	Phone	Phone
Data Type:	Text(24)	nvarchar(24)

Table: SuppContacts

Microsoft Access

Table Name: SuppContacts

Attached Table Name:

Aliasing Query:

Validation Rule:

SQL Server

SuppContacts

Timestamp field added to SQL Server table.

Fields

Microsoft Access

SQL Server

Field Name: ContactID

ContactID

Data Type: Number (Long)

int

Field Name: SupplierID

SupplierID

Data Type: Number (Long)

int

Field Name: FirstName

FirstName

Data Type: Text(50)

nvarchar(50)

Field Name: LastName

LastName

Data Type: Text(50)

nvarchar(50)

Field Name: Dear

Dear

Data Type: Text(50)

nvarchar(50)

Field Name: Title

Title

Data Type: Text(50)

nvarchar(50)

Field Name: WorkPhone

WorkPhone

Data Type: Text(30)

nvarchar(30)

Field Name: WorkExtension

WorkExtension

Data Type: Text(20)

nvarchar(20)

Field Name: HomePhone

HomePhone

Data Type: Text(30)

nvarchar(30)

Field Name: MobilePhone

MobilePhone

Data Type: Text(30)

nvarchar(30)

Field Name: FaxNumber

FaxNumber

Data Type: Text(30)

nvarchar(30)

Field Name: EmailName

EmailName

Data Type: Text(50)

nvarchar(50)

Field Name: LastMeetingDate	LastMeetingDate
Data Type: Date/Time	datetime
Field Name: ContactTypeID	ContactTypeID
Data Type: Number (Long)	int
Field Name: ReferredBy	ReferredBy
Data Type: Text(50)	nvarchar(50)
Field Name: Notes	Notes
Data Type: Memo	text

Indexes	Microsoft Access	SQL Server
	Name: ContactTypeID	ContactTypeID
	Fields: ContactTypeID	ContactTypeID
	Type: DuplicatesOK	DuplicatesOK
	Name: EmailName	EmailName
	Fields: EmailName	EmailName
	Type: DuplicatesOK	DuplicatesOK
	Name: LastName	LastName
	Fields: LastName	LastName
	Type: DuplicatesOK	DuplicatesOK
	Name: SupplierID	SupplierID
	Fields: SupplierID	SupplierID
	Type: DuplicatesOK	DuplicatesOK
	Name: SuppliersSuppContacts	SuppliersSuppContacts
	Fields: SupplierID	SupplierID
	Type: DuplicatesOK	DuplicatesOK

Table: Suppliers

Microsoft Access

Table Name: Suppliers

Attached Table Name:

Aliasing Query:

Validation Rule:

SQL Server

Suppliers

Timestamp field added to SQL Server table.

Fields

Microsoft Access

SQL Server

Field Name: SupplierID

SupplierID

Data Type: Number (Long)

int

Field Name: CompanyName

CompanyName

Data Type: Text(40)

nvarchar(40)

Field Name: ContactName

ContactName

Data Type: Text(30)

nvarchar(30)

Field Name: ContactTitle

ContactTitle

Data Type: Text(30)

nvarchar(30)

Field Name: Address

Address

Data Type: Text(60)

nvarchar(60)

Field Name: City

City

Data Type: Text(15)

nvarchar(15)

Field Name: Region

Region

Data Type: Text(15)

nvarchar(15)

Field Name: PostalCode

PostalCode

Data Type: Text(10)

nvarchar(10)

Field Name: Country

Country

Data Type: Text(15)

nvarchar(15)

Field Name: Phone

Phone

Data Type: Text(24)

nvarchar(24)

Field Name: Fax

Fax

Data Type: Text(24)

nvarchar(24)

Field Name: HomePage

HomePage

Data Type: Memo

text

Indexes

Microsoft Access

Name: CompanyName

Fields: CompanyName

Type: DuplicatesOK

Name: PostalCode

Fields: PostalCode

Type: DuplicatesOK

SQL Server

CompanyName

CompanyName

DuplicatesOK

PostalCode

PostalCode

DuplicatesOK

Queries

Query Nam

Alphabetical List of Products

Upsized using SQL:

```
CREATE VIEW "Alphabetical List of Products"
AS
SELECT Products.*, Categories.CategoryName AS Expr1001
FROM Categories INNER JOIN Products ON (Categories.CategoryID=Products.CategoryID)
WHERE (((Products.Discontinued)=0))
```

Query Nam

Product Sales for 2006

Upsized using SQL:

```
CREATE VIEW "Product Sales for 2006"
AS
SELECT Categories.CategoryName, Products.ProductName, sum(convert(money,"Order
Details".UnitPrice*Quantity*(1-Discount)/100)*100) AS ProductSales, 'Qtr ' +
datepart(q,ShippedDate) AS ShippedQuarter
FROM (Categories INNER JOIN Products ON (Categories.CategoryID=Products.CategoryID)) INNER
JOIN (Orders INNER JOIN "Order Details" ON (Orders.OrderID="Order Details".OrderID)) ON
(Products.ProductID="Order Details".ProductID)
WHERE (((Orders.ShippedDate) Between '1/1/2006' And '12/31/2006'))
GROUP BY Categories.CategoryName, Products.ProductName, 'Qtr ' + datepart(q,ShippedDate)
```

Query Nam

Current Product List

Upsized using SQL:

```
CREATE FUNCTION "Current Product List" ()
RETURNS TABLE
AS RETURN (SELECT TOP 100 PERCENT "Product List".ProductID, "Product List".ProductName
FROM Products AS "Product List"
WHERE (((("Product List".Discontinued)=0))
ORDER BY "Product List".ProductName)
```

Query Nam

Customers and Suppliers by City

Upsized using SQL:

```
CREATE PROCEDURE "Customers and Suppliers by City"
AS
SELECT City, CompanyName, ContactName, 'Customers' AS Relationship
FROM Customers UNION SELECT City, CompanyName, ContactName, 'Suppliers' AS _Suppliers_
FROM Suppliers
ORDER BY City, CompanyName
```

Query Nam Order Subtotals

Upsized using SQL:

```
CREATE VIEW "Order Subtotals"
AS
SELECT "Order Details".OrderID, sum(convert(money,UnitPrice*Quantity*(1-Discount)/100)*100)
AS Subtotal
FROM "Order Details"
GROUP BY "Order Details".OrderID
```

Query Nam fltOrdersNorthAmerica

Upsized using SQL:

```
CREATE FUNCTION fltOrdersNorthAmerica ()
RETURNS TABLE
AS RETURN (SELECT TOP 100 PERCENT Orders.*
FROM Orders
WHERE (((Orders.ShipCountry)='USA') AND ((Orders.OrderDate)>='1/1/1997')) OR
(((Orders.ShipCountry)='Canada') AND ((Orders.OrderDate)>='1/1/1997')) OR
(((Orders.ShipCountry)='Mexico') AND ((Orders.OrderDate)>='1/1/1997'))
ORDER BY Orders.ShipCountry, Orders.OrderDate)
```

Query Nam Invoices

Upsized using SQL:

```
CREATE VIEW Invoices
AS
SELECT Orders.ShipName, Orders.ShipAddress, Orders.ShipCity, Orders.ShipRegion,
Orders.ShipPostalCode, Orders.ShipCountry, Orders.CustomerID, Customers.CompanyName,
Customers.Address, Customers.City, Customers.Region, Customers.PostalCode,
Customers.Country, FirstName + ' ' + LastName AS Salesperson, Orders.OrderID,
Orders.OrderDate, Orders.RequiredDate, Orders.ShippedDate, Shippers.CompanyName AS
CompanyName1, "Order Details".ProductID AS ProductID, Products.ProductName, "Order
Details".UnitPrice AS UnitPrice, "Order Details".Quantity, "Order Details".Discount,
convert(money,"Order Details".UnitPrice*Quantity*(1-Discount)/100)*100 AS ExtendedPrice,
Orders.Freight
FROM Shippers INNER JOIN (Products INNER JOIN ((Employees INNER JOIN (Customers INNER
JOIN Orders ON (Customers.CustomerID=Orders.CustomerID)) ON
(Employees.EmployeeID=Orders.EmployeeID)) INNER JOIN "Order Details" ON
(Orders.OrderID="Order Details".OrderID)) ON (Products.ProductID="Order Details".ProductID))
ON (Shippers.ShipperID=Orders.ShipVia)
```

Query Nam Invoices Filter

Upsized using SQL:

```
CREATE FUNCTION "Invoices Filter" (@Forms___Orders___OrderID1 varchar (255)
)
RETURNS TABLE
AS RETURN (SELECT Invoices.*
FROM Invoices
WHERE (((Invoices.OrderID)=@Forms___Orders___OrderID1)))
```


Query Nam	<p>Order Details Extended</p> <p>Upsized using SQL:</p> <pre> CREATE FUNCTION "Order Details Extended" () RETURNS TABLE AS RETURN (SELECT TOP 100 PERCENT "Order Details".OrderID, "Order Details".ProductID AS ProductID, Products.ProductName, "Order Details".UnitPrice AS UnitPrice, "Order Details".Quantity, "Order Details".Discount, convert(money,"Order Details".UnitPrice*Quantity*(1-Discount)/100)*100 AS ExtendedPrice FROM Products INNER JOIN "Order Details" ON (Products.ProductID="Order Details".ProductID) ORDER BY "Order Details".OrderID) </pre>
Query Nam	<p>Employee Sales by Country</p> <p>Upsized using SQL:</p> <pre> CREATE FUNCTION "Employee Sales by Country" (@Beginning_Date1 datetime , @Ending_Date2 datetime) RETURNS TABLE AS RETURN (SELECT Employees.Country, Employees.LastName, Employees.FirstName, Orders.ShippedDate, Orders.OrderID, "Order Subtotals".Subtotal AS SaleAmount FROM Employees INNER JOIN (Orders INNER JOIN "Order Subtotals" ON (Orders.OrderID="Order Subtotals".OrderID)) ON (Employees.EmployeeID=Orders.EmployeeID) WHERE (((Orders.ShippedDate) Between @Beginning_Date1 And @Ending_Date2))) </pre>
Query Nam	<p>Orders Qry</p> <p>Upsized using SQL:</p> <pre> CREATE VIEW "Orders Qry" AS SELECT Orders.OrderID, Orders.CustomerID, Orders.EmployeeID, Orders.OrderDate, Orders.RequiredDate, Orders.ShippedDate, Orders.ShipVia, Orders.Freight, Orders.ShipName, Orders.ShipAddress, Orders.ShipCity, Orders.ShipRegion, Orders.ShipPostalCode, Orders.ShipCountry, Customers.CompanyName, Customers.Address, Customers.City, Customers.Region, Customers.PostalCode, Customers.Country FROM Customers INNER JOIN Orders ON (Customers.CustomerID=Orders.CustomerID) </pre>
Query Nam	<p>Category Sales for 2006</p> <p>Upsized using SQL:</p> <pre> CREATE VIEW "Category Sales for 2006" AS SELECT "Product Sales for 2006".CategoryName, sum("Product Sales for 2006".ProductSales) AS CategorySales FROM "Product Sales for 2006" GROUP BY "Product Sales for 2006".CategoryName </pre>
Query Nam	<p>Products Above Average Price</p> <p>Upsized using SQL:</p> <pre> CREATE FUNCTION "Products Above Average Price" () RETURNS TABLE AS RETURN (SELECT TOP 100 PERCENT Products.ProductName, Products.UnitPrice FROM Products WHERE (((Products.UnitPrice)>(SELECT avg(UnitPrice) From Products))) ORDER BY Products.UnitPrice DESC) </pre>

Query Nam Products by Category

Upsized using SQL:

```
CREATE FUNCTION "Products by Category" ()
RETURNS TABLE
AS RETURN (SELECT TOP 100 PERCENT Categories.CategoryName, Products.ProductName,
Products.QuantityPerUnit, Products.UnitsInStock, Products.Discontinued
FROM Categories INNER JOIN Products ON (Categories.CategoryID=Products.CategoryID)
WHERE (((Products.Discontinued)<>1))
ORDER BY Categories.CategoryName, Products.ProductName)
```

Query Nam qry2006OrderDataPT

Failed to upsize. Attempted to use SQL:

```
CREATE VIEW qry2006OrderDataPT
AS
SELECT Employees.LastName AS Name, Orders.OrderDate AS Date, Orders.ShipCountry AS
Country, "Order Details Extended".ExtendedPrice AS Orders, Products.ProductName AS Product,
Categories.CategoryName AS Category
FROM Employees INNER JOIN (Categories INNER JOIN ((Orders INNER JOIN [Order Details
Extended]() "Order Details Extended" ON (Orders.OrderID = "Order Details Extended".OrderID))
INNER JOIN Products ON ([Order Details Extended]() "Order Details Extended".ProductID =
Products.ProductID)) ON (Categories.CategoryID = Products.CategoryID)) ON
(Employees.EmployeeID = Orders.EmployeeID)
```

Query Nam qry2006SalesChart

Upsized using SQL:

```
CREATE VIEW qry2006SalesChart
AS
SELECT Categories.CategoryName, convert(money,"Order Details".UnitPrice*"Order
Details".Quantity*(1-"Order Details".Discount)) AS Amount, Orders.ShippedDate
FROM (Categories INNER JOIN Products ON (Categories.CategoryID = Products.CategoryID))
INNER JOIN (Orders INNER JOIN "Order Details" ON (Orders.OrderID = "Order Details".OrderID))
ON (Products.ProductID = "Order Details".ProductID)
WHERE (((Orders.ShippedDate) Between '1/1/2006' And '12/31/2006'))
```

Query Nam qry2006SalesByCategoryChart

Crosstab queries cannot be upsized.

Query Nam qryAQtblShipAddresses

Upsized using SQL:

```
CREATE PROCEDURE qryAQtblShipAddresses
AS
INSERT INTO tblShipAddresses (CustomerID, ShipName, ShipAddress, ShipCity, ShipRegion,
ShipPostalCode, ShipCountry)
SELECT DISTINCT Customers.CustomerID, Orders.ShipName, Orders.ShipAddress, Orders.ShipCity,
Orders.ShipRegion, Orders.ShipPostalCode, Orders.ShipCountry
FROM Customers INNER JOIN Orders ON (Customers.CustomerID=Orders.CustomerID)
WHERE (((Orders.ShipName)<>Customers.CompanyName)) Or
(((Orders.ShipAddress)<>Customers.Address)) Or (((Orders.ShipCity)<>Customers.City))
```

Query Nam qryCTWizSource

Upsized using SQL:

```
CREATE VIEW qryCTWizSource
AS
SELECT Orders.OrderDate, "Order Details Extended".ProductID AS "_ Order Details
Extended.ProductID_", "Order Details Extended".ProductName, "Order Details
Extended".ExtendedPrice
FROM Orders INNER JOIN [Order Details Extended]() "Order Details Extended" ON (Orders.OrderID
= "Order Details Extended".OrderID)
WHERE (((Orders.OrderDate) Between '1/1/2006' And '12/31/2006'))
```

Query Nam qryCustomersUSA

Upsized using SQL:

```
CREATE FUNCTION qryCustomersUSA ()
RETURNS TABLE
AS RETURN (SELECT TOP 100 PERCENT Customers.ContactName, Customers.CompanyName,
Customers.Address, Customers.City, Customers.Region, Customers.PostalCode
FROM Customers
WHERE (((Customers.Country)='USA'))
ORDER BY Customers.PostalCode)
```

Query Nam qryDQOrders

Upsized using SQL:

```
CREATE PROCEDURE qryDQOrders
AS
DELETE
FROM Orders
WHERE (((Orders.CustomerID)='ERNSH') AND ((Orders.ShippedDate) Is Null))
```

Query Nam qryHRActions

Upsized using SQL:

```
CREATE FUNCTION qryHRActions ()
RETURNS TABLE
AS RETURN (SELECT TOP 100 PERCENT HRActions.*, Employees.LastName AS Expr1001,
Employees.FirstName AS Expr1002, Employees.Title AS Expr1003, Employees.HireDate AS
Expr1004, Employees.Extension AS Expr1005, Employees.ReportsTo AS Expr1006,
Employees.Notes AS Expr1007, Employees.PhotoOLE AS Expr1008
FROM Employees INNER JOIN HRActions ON (Employees.EmployeeID=HRActions.EmployeeID)
ORDER BY Employees.LastName)
```

Query Nam qryInventory

Upsized using SQL:

```
CREATE VIEW qryInventory
AS
SELECT Products.*
FROM Products
WHERE (((Products.Discontinued)=0))
```

Query Nam qryInventoryChart

Upsized using SQL:

```
CREATE VIEW qryInventoryChart
AS
SELECT Products.*, Categories.CategoryName AS Expr1001
FROM Categories INNER JOIN Products ON (Categories.CategoryID=Products.CategoryID)
WHERE (((Products.Discontinued)=0))
```

Query Nam qryMonthlyOrders2006

Failed to upsize. Attempted to use SQL:

```
CREATE VIEW qryMonthlyOrders2006
AS
SELECT format(OrderDate,'yyyy-mm') AS Month, count("Order Subtotals".OrderID) AS
CountOfOrderID, sum("Order Subtotals".Subtotal) AS SumOfSubtotal, avg("Order
Subtotals".Subtotal) AS AvgOfSubtotal, min("Order Subtotals".Subtotal) AS MinOfSubtotal,
max("Order Subtotals".Subtotal) AS MaxOfSubtotal
FROM Orders INNER JOIN "Order Subtotals" ON (Orders.OrderID="Order Subtotals".OrderID)
WHERE (((Orders.OrderDate) Between '1/1/2006' And '12/31/2006'))
GROUP BY format(OrderDate,'yyyy-mm')
```

Query Nam qryMonthlyOrdersParam

Failed to upsize. Attempted to use SQL:

```
CREATE FUNCTION qryMonthlyOrdersParam (@Enter_the_year_and_month_in_YY1 varchar (255)
)
RETURNS TABLE
AS RETURN (SELECT format(OrderDate,'yyyy-mm') AS Month, count("Order Subtotals".OrderID) AS
CountOfOrderID, sum("Order Subtotals".Subtotal) AS SumOfSubtotal, avg("Order
Subtotals".Subtotal) AS AvgOfSubtotal, min("Order Subtotals".Subtotal) AS MinOfSubtotal,
max("Order Subtotals".Subtotal) AS MaxOfSubtotal
FROM Orders INNER JOIN "Order Subtotals" ON (Orders.OrderID="Order Subtotals".OrderID)
WHERE (((format(OrderDate,'yyyy-mm'))=@Enter_the_year_and_month_in_YY1))
GROUP BY format(OrderDate,'yyyy-mm'))
```

Query Nam qryMonthlySales2006

Failed to upsize. Attempted to use SQL:

```
CREATE VIEW qryMonthlySales2006
AS
SELECT format(OrderDate,'yyyy-mm') AS Month, count("Order Subtotals".OrderID) AS
CountOfOrderID, sum("Order Subtotals".Subtotal) AS SumOfSubtotal, avg("Order
Subtotals".Subtotal) AS AvgOfSubtotal, min("Order Subtotals".Subtotal) AS MinOfSubtotal,
max("Order Subtotals".Subtotal) AS MaxOfSubtotal
FROM Orders INNER JOIN "Order Subtotals" ON (Orders.OrderID="Order Subtotals".OrderID)
WHERE (((Orders.OrderDate) Between '1/1/2006' And '12/31/2006'))
GROUP BY format(OrderDate,'yyyy-mm')
```

Query Nam qryMTtblShipAddresses

Upsized using SQL:

```
CREATE PROCEDURE qryMTtblShipAddresses
AS
SELECT DISTINCT Customers.CustomerID, Orders.ShipName, Orders.ShipAddress, Orders.ShipCity,
Orders.ShipRegion, Orders.ShipPostalCode, Orders.ShipCountry INTO tblShipAddresses
FROM Customers INNER JOIN Orders ON (Customers.CustomerID=Orders.CustomerID)
WHERE (((Orders.ShipName)<>Customers.CompanyName)) Or
(((Orders.ShipAddress)<>Customers.Address)) Or (((Orders.ShipCity)<>Customers.City))
```

Query Nam qryOrderAmount

Failed to upsize. Attempted to use SQL:

```
CREATE FUNCTION qryOrderAmount ()
RETURNS TABLE
AS RETURN (SELECT TOP 100 PERCENT Customers.CompanyName, Customers.PostalCode,
Customers.Country, Orders.OrderID, Orders.OrderDate, sum(convert(money,UnitPrice*Quantity*(1-
Discount))) AS Amount, Orders.Freight, Freight/Amount AS FreightPct
FROM (Customers INNER JOIN Orders ON (Customers.CustomerID=Orders.CustomerID)) INNER
JOIN "Order Details" ON (Orders.OrderID="Order Details".OrderID)
GROUP BY Customers.CompanyName, Customers.PostalCode, Customers.Country, Orders.OrderID,
Orders.OrderDate, Orders.Freight
ORDER BY Orders.OrderID)
```

Query Nam qryOrderAmountsRecentTop5%

Upsized using SQL:

```
CREATE VIEW "qryOrderAmountsRecentTop5%"
AS
SELECT TOP 5 PERCENT Customers.CompanyName, Orders.OrderID, Orders.OrderDate, "Order
Subtotals".Subtotal, Orders.Freight
FROM Customers INNER JOIN (Orders INNER JOIN "Order Subtotals" ON (Orders.OrderID =
"Order Subtotals".OrderID)) ON (Customers.CustomerID = Orders.CustomerID)
ORDER BY Orders.OrderDate DESC
```

Query Nam qryOrderDetails

Upsized using SQL:

```
CREATE VIEW qryOrderDetails
AS
SELECT "Order Details".*
FROM "Order Details"
```

Query Nam qryOrdersAggregates

Upsized using SQL:

```
CREATE VIEW qryOrdersAggregates
AS
SELECT count("Order Subtotals".OrderID) AS CountOfOrderID, sum("Order Subtotals".Subtotal) AS
SumOfSubtotal, avg("Order Subtotals".Subtotal) AS AvgOfSubtotal, min("Order Subtotals".Subtotal)
AS MinOfSubtotal, max("Order Subtotals".Subtotal) AS MaxOfSubtotal
FROM "Order Subtotals"
```

Query Nam qryOrdersByCountry&Category

Upsized using SQL:

```
CREATE FUNCTION "qryOrdersByCountry&Category" ()
RETURNS TABLE
AS RETURN (SELECT TOP 100 PERCENT Customers.Country, Categories.CategoryName,
sum(convert(money,"Order Details".UnitPrice*Quantity*(1-Discount))) AS Amount
FROM (Categories INNER JOIN Products ON (Categories.CategoryID=Products.CategoryID)) INNER
JOIN ((Customers INNER JOIN Orders ON (Customers.CustomerID=Orders.CustomerID)) INNER
JOIN "Order Details" ON (Orders.OrderID="Order Details".OrderID)) ON
(Products.ProductID="Order Details".ProductID)
GROUP BY Customers.Country, Categories.CategoryName
ORDER BY Categories.CategoryName)
```

Query Nam	<p>qryOrdersWHEREClause</p> <p>Upsized using SQL:</p> <pre>CREATE FUNCTION qryOrdersWHEREClause () RETURNS TABLE AS RETURN (SELECT TOP 100 PERCENT Customers.Country, Categories.CategoryName, sum(convert(money,"Order Details".UnitPrice*Quantity*(1-Discout))) AS Amount FROM Categories, Products, "Order Details", Orders, Customers WHERE Categories.CategoryID=Products.CategoryID And Customers.CustomerID=Orders.CustomerID And Orders.OrderID="Order Details".OrderID And Products.ProductID="Order Details".ProductID GROUP BY Customers.Country, Categories.CategoryName ORDER BY Categories.CategoryName)</pre>
Query Nam	<p>qryShipAddressesUnion</p> <p>This query is not upsizeable</p>
Query Nam	<p>qryShipBillAddresses</p> <p>Upsized using SQL:</p> <pre>CREATE VIEW qryShipBillAddresses AS SELECT DISTINCT Customers.CompanyName, Customers.Address, Orders.ShipAddress FROM Customers INNER JOIN Orders ON (Customers.Address = Orders.ShipAddress)</pre>
Query Nam	<p>qryShortOrders</p> <p>Upsized using SQL:</p> <pre>CREATE VIEW qryShortOrders AS SELECT Orders.OrderID, Orders.CustomerID, Orders.OrderDate, Orders.ShippedDate, Orders.ShipVia FROM Orders</pre>
Query Nam	<p>qrySubQuery</p> <p>Upsized using SQL:</p> <pre>CREATE VIEW qrySubQuery AS SELECT Customers.CompanyName, Customers.ContactName, Customers.ContactTitle, Customers.Phone FROM Customers WHERE (((Customers.CustomerID) In (SELECT CustomerID FROM Orders WHERE OrderDate BETWEEN '1/1/1997' AND '6/30/1997')))</pre>
Query Nam	<p>qryUnionQuery</p> <p>Upsized using SQL:</p> <pre>CREATE PROCEDURE qryUnionQuery AS SELECT City, CompanyName, ContactName, CustomerID AS Code, 'Customer' AS Relationship FROM Customers UNION SELECT City, CompanyName, ContactName, SupplierID, 'Supplier' AS _Supplier_ FROM Suppliers ORDER BY City, CompanyName</pre>

Query Nam qryUSMailList

Upsized using SQL:

```
CREATE PROCEDURE qryUSMailList
AS
SELECT Customers.ContactName AS Contact, Customers.CompanyName AS Company,
Customers.Address, Customers.City, Customers.Region AS State, Customers.PostalCode AS ZIP
INTO tblUSMailList
FROM Customers
WHERE (((Customers.Country)='USA'))
```

Query Nam Quarterly Orders

Upsized using SQL:

```
CREATE VIEW "Quarterly Orders"
AS
SELECT Customers.CustomerID, Customers.CompanyName, Customers.City, Customers.Country
FROM Customers RIGHT JOIN Orders ON (Customers.CustomerID=Orders.CustomerID)
WHERE (((Orders.OrderDate) Between '1/1/1997' And '12/31/1997'))
```

Query Nam Sales by Category

Upsized using SQL:

```
CREATE FUNCTION "Sales by Category" ()
RETURNS TABLE
AS RETURN (SELECT TOP 100 PERCENT Categories.CategoryID, Categories.CategoryName,
Products.ProductName, sum("Order Details Extended".ExtendedPrice) AS ProductSales
FROM Categories INNER JOIN (Products INNER JOIN (Orders INNER JOIN [Order Details
Extended]() "Order Details Extended" ON (Orders.OrderID="Order Details Extended".OrderID)) ON
(Products.ProductID="Order Details Extended".ProductID)) ON
(Categories.CategoryID=Products.CategoryID)
WHERE (((Orders.OrderDate) Between '1/1/1997' And '12/31/1997'))
GROUP BY Categories.CategoryID, Categories.CategoryName, Products.ProductName
ORDER BY Categories.CategoryName)
```

Query Nam Sales by Year

Failed to upsize. Attempted to use SQL:

```
CREATE FUNCTION "Sales by Year" (@Forms___Sales_by_Year_Dialog__1 datetime
, @Forms___Sales_by_Year_Dialog__2 datetime
)
RETURNS TABLE
AS RETURN (SELECT Orders.ShippedDate, Orders.OrderID, "Order Subtotals".Subtotal,
format(ShippedDate,'yyyy') AS Year
FROM Orders INNER JOIN "Order Subtotals" ON (Orders.OrderID = "Order Subtotals".OrderID)
WHERE (((Orders.ShippedDate) Is Not Null And (Orders.ShippedDate) Between
@Forms___Sales_by_Year_Dialog__1 And @Forms___Sales_by_Year_Dialog__2)))
```

Query Nam Sales Totals by Amount

Upsized using SQL:

```
CREATE VIEW "Sales Totals by Amount"
AS
SELECT "Order Subtotals".Subtotal AS SaleAmount, Orders.OrderID, Customers.CompanyName,
Orders.ShippedDate
FROM Customers INNER JOIN (Orders INNER JOIN "Order Subtotals" ON (Orders.OrderID="Order
Subtotals".OrderID)) ON (Customers.CustomerID=Orders.CustomerID)
WHERE (((("Order Subtotals".Subtotal)>2500) AND ((Orders.ShippedDate) Between '1/1/1997' And
'12/31/1997'))
```

Query Nam	<p>Summary of Sales by Quarter</p> <p>Upsized using SQL:</p> <pre> CREATE FUNCTION "Summary of Sales by Quarter" () RETURNS TABLE AS RETURN (SELECT TOP 100 PERCENT Orders.ShippedDate, Orders.OrderID, "Order Subtotals".Subtotal FROM Orders INNER JOIN "Order Subtotals" ON (Orders.OrderID="Order Subtotals".OrderID) WHERE (((Orders.ShippedDate) Is Not Null)) ORDER BY Orders.ShippedDate) </pre>
Query Nam	<p>Summary of Sales by Year</p> <p>Upsized using SQL:</p> <pre> CREATE FUNCTION "Summary of Sales by Year" () RETURNS TABLE AS RETURN (SELECT TOP 100 PERCENT Orders.ShippedDate, Orders.OrderID, "Order Subtotals".Subtotal FROM Orders INNER JOIN "Order Subtotals" ON (Orders.OrderID="Order Subtotals".OrderID) WHERE (((Orders.ShippedDate) Is Not Null)) ORDER BY Orders.ShippedDate) </pre>
Query Nam	<p>Ten Most Expensive Products</p> <p>Upsized using SQL:</p> <pre> CREATE VIEW "Ten Most Expensive Products" AS SELECT TOP 10 Products.ProductName AS TenMostExpensiveProducts, Products.UnitPrice FROM Products ORDER BY Products.UnitPrice DESC </pre>