

## Views

### نظرة عامة:

رأينا عند دراسة وتحليل البيانات، أن الاستعلام تقنية تسمح بعزل سلسلة من الحقول و/أو السجلات من الجداول، يتم ذلك عادة لغرض تحليل البيانات. ويمكن أيضا القيام بذلك لأجل إنشاء قائمة جديدة من البيانات لأغراض معينة.

يتم إنشاء الاستعلام غالبا بشكل مؤقت، مثلا خلال تحليل البيانات، باستخدام الجداول، النماذج، صفحات على شبكة الإنترنت. يتم التخلص من الاستعلام مباشرة بعد الحصول على القائمة المطلوبة.

تسمح العديد من تطبيقات قواعد البيانات، بما في ذلك SQL Server، بإنشاء استعلامات مع إمكانية تخزينها لاستخدامها لاحقا، أو حتى لاستخدامها كما لو كانت جداول من قاعدة البيانات. هذه هي الفكرة الأساسية حول طريقة عمل كائنات تدعى **Views**.

View هي لائحة من الحقول أو مجموعة من السجلات التي تم استردادها من جدول أو أكثر، أو مزيج من View واحدة أو أكثر، أو Views وجداول أخرى. بناء على هذا، قبل إنشاء View يجب أولا تحديد الحقول والسجلات التي سيتم التعامل معها.

### تطبيق عملي: التمهيد لإنشاء View

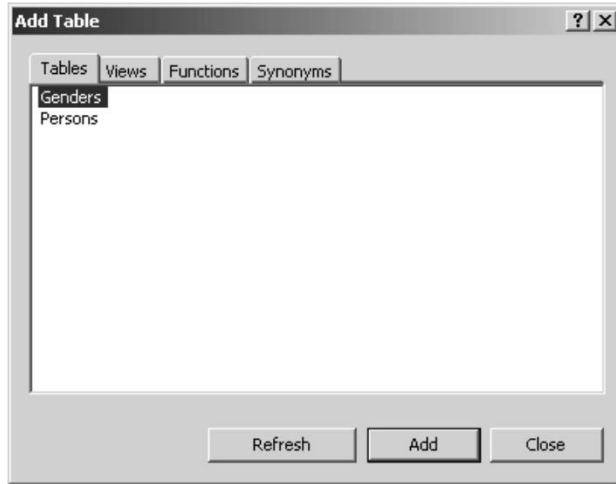
1- قم بتشغيل SQL Server Management Studio واتصل بالخادم

2- تأكد من وجود قاعدة بيانات RealEstate2 التي تم إنشاؤها في درس سابق.

### أساسيات قبل إنشاء View:

من أجل إنشاء View، يمكنك استخدام متصفح الكائنات أو إطار الاستعلام. وقبل بدء إنشاء View ينبغي تحديد الجدول (أو الجداول) التي سيتم إدراجها في View.

لإنشاء View من متصفح الكائنات، يمكنك توسيع البند Databases ثم انقر بزر الأيمن على Views ثم اختر New View. وسيفتح مربع الحوار لإضافة الجداول:



يتم العمل على عتبة الحوار هذه، بنفس الطريقة التي استعرضناها في درس سابق

- لتحديد الجدول الذي سيتم استخدامه كمصدر للبيانات، انقر على الصفحة Tables.

- إذا أردت إضافة View


آخر، حدد الصفحة Views من الأعلى، ثم اختر كائن View الذي تود إدراجه.

- إذا أردت استخدام إحدى الدوال لتوليد السجلات، يمكنك العثور عليها في الصفحة Functions.

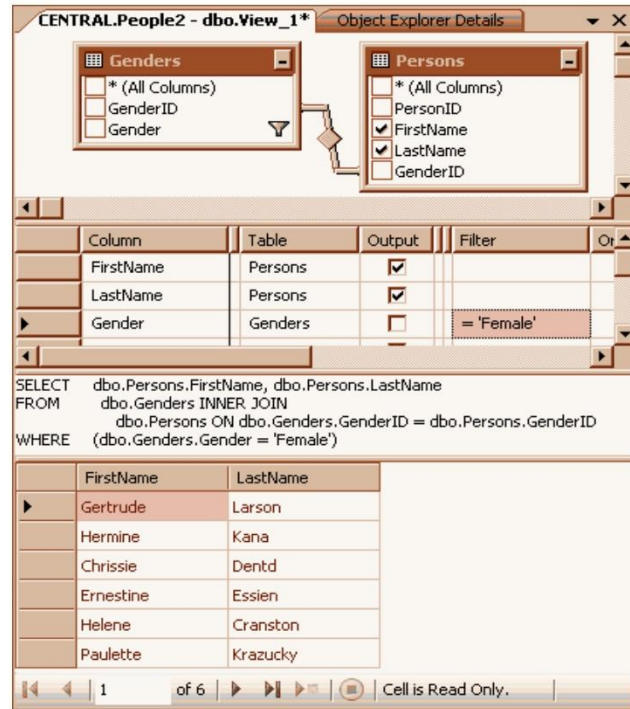
بعد تحديد كائنات المصدر التي تود استخدامها لإنشاء View، يمكنك النقر نقرا مزدوجا عليها أو قم بتحديدتها ثم انقر على Add.

انقر على Close من مربع الحوار بعد إضافة جميع الكائنات التي تحتاجها لبناء View.

كما رأينا في درس بناء الاستعلام، يمكنك إنشاء صلة بين الجداول بسحب أي حقل من جدول وإسقاطه على أي حقل من جدول آخر، كذلك كتابة الاستعلام مباشرة وغير ذلك....

يمكنك في أي وقت اختبار ناتج View بتنفيذ الاستعلام بالنقر على زر التنفيذ .

وبذلك تظهر في القسم السفلي من واجهة تصميم View نتائج الاستعلام. مثال:



من أهم مميزات كائنات Views هو إمكانية استخدامها مرارا وتكرارا. لتحقيق ذلك، يجب حفظ View في قاعدة البيانات. مثل باقي الكائنات في SQL Server يجب أن تمتلك View اسما خاصا بها، وأن يتم حفظها في قاعدة البيانات. لحفظ View في واجهة التصميم يمكن النقر على زر الحفظ Save من شريط الأدوات. يمكنك أيضا غلق واجهة التصميم ثم حفظ View بعد أن تتلقى علبة حوار تأكيد خيار الحفظ، يتبع اسم View نفس القواعد التي رأيناها لتسمية الكائنات في SQL Server.

#### تطبيق عملي: إنشاء View في واجهة التصميم

- 1- في متصفح الكائنات، قم بتوسيع البند Databases ثم RealEstate2
- 2- بالزر الأيمن انقر على Views ثم اختر New View
- 3- في مربع الحوار Add Table انقر نقرًا مزدوجًا على PropertyTypes، Properties و Conditions

4- انقر على Close

5- من الجدول PropertyTypes، اسحب الحقل PropertyTypeID وأسقطه على الحقل Properties من الجدول PropertyTypeID

6- من الجدول Conditions، اسحب الحقل ConditionID وأسقطه على الحقل Properties من الجدول ConditionID

7- قم بتحديد الحقول التالية:

PropertyNumber, PropertyType, Condition, City, ZIPCode, Bedrooms, Bathrooms, FinishedBasement, Stories, YearBuilt, MarketValue

8- في قسم المعايير (Criteria)، انقر على PropertyType واضغط Tab 6 مرات، لتصل إلى عمود التصفية (Filter) واكتب Single Family

9- اضغط Tab، واكتب Townhouse في العمود Or...

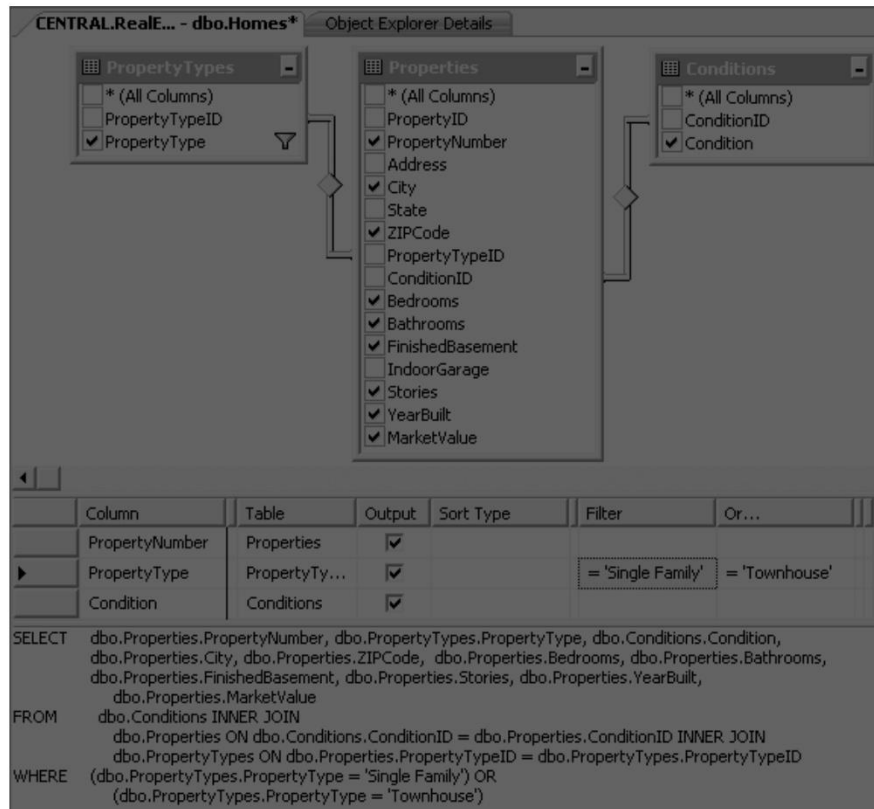
سننتج في تسمية كائنات View في هذه الدروس نفس طريقة تسمية الكائنات الأخرى.

تصبح View بعد حفظها في قاعدة البيانات جزءاً منها، وسيتم إضافتها تحت البند الرئيسي Views بالاسم الذي تم حفظها به.

### تطبيق عملي: حفظ كائن View

1- لحفظ View، انقر من شريط الأدوات Standard، على الزر Save

2- اكتب اسم View مثلاً: Homes، ثم انقر على OK



3- أغلق واجهة تصميم View

4- من متصفح الكائنات، تحت الـ RealEstate2، انقر بالزر الأيمن على Views ثم

اختر New View

5- من مربع الحوار Add Table، اختر الجداول PropertyTypes, Properties, Conditions

6- انقر على Close

7- من الجدول PropertyTypes، اسحب الحقل PropertyTypeID وأسقطه على الحقل

PropertyTypeID من الجدول Properties

8- من الجدول Conditions، اسحب ConditionID وأسقطه على الحقل ConditionID

من الجدول Properties

9- حدد من كل الجداول الحقول التالية:

PropertyNumber, PropertyType, Condition, City, ZIPCode, Bedrooms,  
Bathrooms, MarketValue

10- في قسم المعايير، انقر على PropertyType، اضغط على الزر Tab 3 مرات

واضغط على الزر Space لإزالة علامة الاختيار من العمود Output

11- اضغط على 3 مرات على Tab، لتصل إلى عمود التصفية، وثم اكتب

Enter Condominiums واضغط

12- من القائمة الرئيسية، انقر على Save -> File

13- اكتب اسم View مثلاً Condos وانقر على OK

14- أغلق نافذة التصميم.

### إنشاء View باستخدام SQL:

لإنشاء View برمجياً، يمكنك استخدام SQL وفقاً للجملة التالية:

```
CREATE VIEW ViewName
AS
SELECT Statement
```

يمكن أن يقوم SQL Server Management Studio بتوليد هيكل SQL لإنشاء View.

للقيام بذلك، أولاً أنشئ إطار استعلام جديد، ثم من Template Explorer اسحب وأفلت البند

Create View تحت البند الرئيسي View.

SQL إنشاء View يبدأ بعبارة CREATE VIEW متبوعاً باسم مناسب، استخدم الكلمة AS

بعد ذلك للدلالة على أن ما سيأتي من الاستعلام لتعريف محتوى View.

يتم تعريف View باستخدام عبارة SELECT، بإتباع نفس القواعد التي رأيناها خلال

تحليل البيانات. مثال:

```
CREATE VIEW dbo.ListOfMen
AS
SELECT dbo.Genders.Gender,
       dbo.Persons.FirstName, dbo.Persons.LastName
FROM   dbo.Genders INNER JOIN dbo.Persons
ON     dbo.Genders.GenderID = dbo.Persons.GenderID
WHERE  (dbo.Genders.Gender = 'Male');
GO
```

بعد كتابة استعلام إنشاء View ثم تنفيذه بضغط F5 أو الزر Execute، أي بمجرد تنفيذ عبارة SQL، سيضاف اسم الكائن View بنفس الاسم الذي ورد في الاستعلام، وستتم إضافة إلى قائمة Views من قاعدة البيانات الحالية حتى لو لم يتم بحفظ الاستعلام.

### تطبيق عملي: إنشاء View

1- تأكد من وجود قاعدة البيانات Yugo National Bank التي تم إنشاؤها في درس سابق.

2- من متصفح الكائنات، انقر بالزر الأيمن على YugoNationalBank وانقر على New Query

3- لإنشاء كائن View جديد، اكتب ما يلي:

```
Use YugoNationalBank;
GO
CREATE VIEW PayrollPreparation
AS
SELECT  dbo.Employees.EmployeeNumber,
        dbo.Employees.LastName + ', ' +
        dbo.Employees.FirstName AS [Full Name],
        dbo.Employees.HourlySalary,
        dbo.TimeSheets.TimeSheetCode, dbo.TimeSheets.Week1Monday,
        dbo.TimeSheets.Week1Tuesday, dbo.TimeSheets.Week1Wednesday,
        dbo.TimeSheets.Week1Thursday, dbo.TimeSheets.Week1Friday,
        dbo.TimeSheets.Week1Saturday, dbo.TimeSheets.Week1Sunday,
        dbo.TimeSheets.Week2Monday, dbo.TimeSheets.Week2Tuesday,
        dbo.TimeSheets.Week2Wednesday, dbo.TimeSheets.Week2Thursday,
        dbo.TimeSheets.Week2Friday, dbo.TimeSheets.Week2Saturday,
        dbo.TimeSheets.Week2Sunday
FROM    dbo.Employees INNER JOIN dbo.TimeSheets
ON      dbo.Employees.EmployeeNumber = dbo.TimeSheets.EmplNumber;
GO
```

4- قم بتنفيذ الاستعلام بالضغط F5

5- احذف محتوى نافذة الاستعلام.

فتح أو تنفيذ View

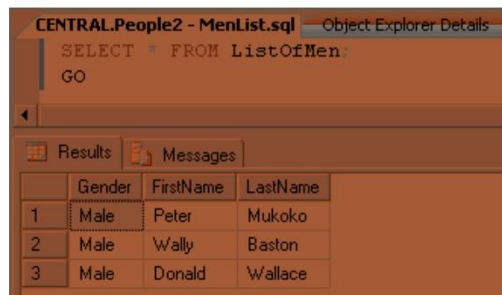


بعد إنشاء كائن View، يمكنك عرضه بطريقتين:

- لمشاهدة هيكل View (الذي يمثل الجداول والعلاقات...)، انقر بالزر الأيمن من متصفح الكائنات على الكائن View ثم اختر Design
- لمشاهدة استعلام SQL الذي يمثل View، انقر بالزر الأيمن من متصفح الكائنات على الكائن View ثم اختر Edit.

تنفيذ كائن View يعني مشاهدة نتائجه، للقيام بذلك يمكنك:

- فتح نافذة استعلام مرتبطة بقاعدة البيانات التي تحتوي على View. ثم كتابة استعلام SELECT وفق نفس الصيغ والقواعد التي رأينا مع الجداول. مثال:



- من متصفح الكائنات، بالزر الأيمن انقر على الكائن View ثم اختر Open View.

#### تطبيق عملي: تنفيذ View

- 1- من متصفح الكائنات، تحت البند RealEste2، قم بتوسيع Views، ثم انقر بالزر الأيمن على Condos ثم اختر Open View



CENTRAL.RealEs...2 - dbo.Condos							
Object Explorer Details							
	PropertyNumber	Condition	City	ZIPCode	Bedrooms	Bathrooms	MarketValue
▶	200417	Excellent	Germantown	NULL	2	1	215495.0000
	NULL	Good	Rockville	NULL	2	2	436885.0000
	288540	Good	Silver Spring	20906	1	1	242775.0000
	NULL	Excellent	Takoma Park	NULL	2	NULL	360885.0000
	NULL	Good	Rockville	NULL	1	1	418885.0000
	811155	Good	Silver Spring	20906	1	1	252775.0000
	NULL	Good	Rockville	NULL	1	1	420555.0000
	644114	Good	Silver Spring	20906	2	2	258445.0000
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL

2- أغلق واجهة ناتج View

3- من متصفح الكائنات، انقر بالزر الأيمن على YugoNationalBank ثم اختر New Query

4- لتنفيذ كائن View اكتب استعلام كما يلي:

```
USE YugoNationalBank;
GO
SELECT PayrollPreparation.* FROM PayrollPreparation;
GO
```

5- لتنفيذ الكائن View، اضغط على F5

6- احذف محتوى نافذة الاستعلام.

صيانة كائنات View:

خصائص الكائن View:

يعتبر كائن View في SQL Server غرض (Object) كبقية الأغراض، للأجل ذلك يمكن مشاهدة محتواه، تعديله أو حذفه، ويملك كائن View خصائصه الخاصة. لمشاهدتها يمكنك النقر بالزر الأيمن على كائن View ثم وانقر على Properties، لتشاهد نافذة الخصائص حاملة معلومات مختلفة، مثل اسم قاعدة البيانات وكائن View الذي ينتمي إليها، تاريخ إنشاء View، وغير ذلك...

تطبيق عملي: خصائص View

## 1- لإنشاء كائن View آخر، اكتب ما يلي:

```
-- =====
-- Database: YugoNationalBank
-- View:      TimeSheet
-- =====
USE YugoNationalBank
GO

IF object_id(N'TimeSheet', 'V') IS NOT NULL
    DROP VIEW dbo.TimeSheet
GO

CREATE VIEW dbo.TimeSheet
AS
SELECT EmplNumber, StartDate,
       Week1Monday, Week1Tuesday, Week1Wednesday,
       Week1Thursday, Week1Friday,
       Week1Saturday, Week1Sunday,
       Week2Monday, Week2Tuesday, Week2Wednesday,
       Week2Thursday, Week2Friday, Week2Saturday,
       Week2Sunday, Notes
FROM   TimeSheets;
```

## 2- اضغط F5 لتنفيذ الاستعلام

## 3- قم بحذف الاستعلام

## 4- من متصفح الكائنات، تحت البند YugoNationalBank، انقر بالزر الأيمن على

Views واختر Refresh

## 5- ثم قم بتوسيع Views. انقر بالزر الأيمن على TimeSheet واختر Properties

## 6- اضغط على Esc لغلق نافذة الخصائص.

## تعديل كائن View:

بعد إنشاء كائن View، يمكنك تغييره أو تعديله حسب الحاجة، لأجل ذلك هناك خياران:

- من متصفح الكائنات، انقر بالزر الأيمن على View، ثم اختر Design، يمكنك حذف أو إضافة الحقول من واجهة التصميم كما يمكنك تغيير أي من الخيارات في أي قسم من واجهة التصميم. بعد التعديلات على الكائن View قم بحفظها وأغلق واجهة التصميم.

- من متصفح الكائنات، انقر بالزر الأيمن على كائن View، اختر Edit، ثم قم بإجراء التعديلات اللازمة ولا تنسى حفظها.

- بطريقة أخرى من متصفح الكائنات، انقر بالزر الأيمن على الكائن View، ثم اختر:

Script View As -> ALTER To -> New Query Editor Window

الصيغة الأساسية لبرمجة لتعديل View هي:

```
ALTER VIEW ViewName
AS
SELECT Statement
```

تبدأ الاستعلام بعبارة ALTER VIEW متبوعة باسم كائن View الذي تود تعديله، بعد ذلك أضف الكلمة AS مشيرة إلى أن ما سيأتي من الاستعلام هو الذي سيغير محتوى View، ثم اكتب الاستعلام على النحو الذي تراه مناسباً. مثلاً، يمكنك إنشاء عبارة SELECT التي تتضمن تعديلات أو استعلام جديد تماماً.

يعرض كائن View الذي قمنا بإنشائه قائمة الذكور فقط من جدول الأشخاص، أضفنا أيضاً الحقل Gender، هذا الحقل زائد عن الحاجة، لأننا نعلم مسبقاً أن القائمة لا تضم سوى الذكور. مثال على تغيير View لإزالة (أو بالأحرى إقصاء) الحقل Gender من جدول الأشخاص:

```
ALTER VIEW dbo.ListOfMen
AS
SELECT dbo.Persons.FirstName, dbo.Persons.LastName
FROM   dbo.Genders INNER JOIN dbo.Persons
ON     dbo.Genders.GenderID = dbo.Persons.GenderID
WHERE  (dbo.Genders.Gender = 'Male');
```

**حذف كائن View:**

يمكن حذف كائن View من قاعدة البيانات إذا لم تكن بحاجة إليه، يمكن ذلك بطرق مختلفة:

- من متصفح الكائنات ومن قاعدة بيانات، انقر بالزر الأيمن على اسم الكائن View الذي تود حذفه وانقر على Delete. وحينها ستتاح لك فرصة لتأكيد خيارك أو لتغيير رأيك.
- من متصفح الكائنات، انقر بالزر الأيمن الكائن View، ثم اختر:

Script View As -> DROP To New Query Editor Window

- بطريقة أخرى، افتح نافذة استعلام مرتبطة بقاعدة البيانات التي تحتوي على كائن View الذي تريد حذفه، من Template Explorer، من البند View، اسحب البند Drop View ثم أسقطه في إطار الاستعلام.

الصيغة العامة لحذف View هي:

`DROP VIEW ViewName`

في الجانب الأيمن لعبارة `DROP VIEW` أضف اسم الكائن View الذي تود حذفه، ثم قم بتنفيذ العبارة لحذف View. ولن تتلقى في هذه الحال أي تحذير من قبل SQL Server.

**استخدام الكائن View:**

**إدخال البيانات باستخدام View:**

تخيل أن لديك جدول الموظفين وتريد إنشاء قائمة بأسمائهم فقط. لأجل ذلك، يمكنك إنشاء View على النحو التالي:

```
CREATE VIEW dbo.EmployeesNames
AS
SELECT FirstName,
       LastName,
       LastName + ', ' + FirstName AS FullName FROM Persons;
GO
```

يستند كائن View السابق إلى جدول، حيث يمكنك القيام بإدخال البيانات إلى هذا الجدول من خلال كائن View، عوضاً عن إدخال البيانات مباشرة إلى الجدول. للقيام بذلك، يمكنك إتباع نفس القواعد التي استعرضنا في درس سابق. مثال:

```
INSERT INTO dbo.EmployeesNames(FirstName, LastName)
VALUES('Peter', 'Justice');
```

إدخال البيانات باستخدام View يعني إدخال البيانات في الجدول الأصل، وهذا يعني أن الجدول سيتم تحديثه تلقائياً. استناداً إلى هذه الميزة، يمكنك إنشاء View خصيصاً لتحديث البيانات في الجدول، بحيث تقوم بعرض الحقول التي يسمح بتحديثها فقط.

**تطبيق عملي: إدخال البيانات باستخدام View**

1- لإنشاء دالة سنستخدمها لاحقاً، أدخل الاستعلام التالي:

```

=====
-- Database: YugoNationalBank
-- Function: CreateTimeSheetCode
-- Purpose: This function takes an employee number
--           and the start date of a time sheet.
--           Then it creates a unique number
--           in the format 00000000000000
--           The first 5 digits represent the
--           employee number,
--           the second 4 digits represent the year,
--           the 2 digits represent the month,
--           that last 2 digits represent the day
=====
USE YugoNationalBank
GO

IF OBJECT_ID (N'dbo.CreateTimeSheetCode') IS NOT NULL
    DROP FUNCTION dbo.CreateTimeSheetCode
GO

CREATE FUNCTION dbo.CreateTimeSheetCode(@EmplNbr varchar(6),
                                         @dteStart datetime)
RETURNS varchar(15)
AS
BEGIN
    DECLARE @strMonth AS varchar(20);
    DECLARE @strDay   AS varchar(20);
    DECLARE @iMonth AS int;
    DECLARE @iDay AS int;
    DECLARE @strTimeSheetCode varchar(20);

    SET @iMonth = CONVERT(varchar(20), MONTH(@dteStart));
    SET @iDay   = CONVERT(varchar(20), DAY(@dteStart));

    IF @iMonth < 10
        SET @strMonth = CONVERT(varchar(20), YEAR(@dteStart)) +
            '0' + CONVERT(varchar(20), @iMonth);
    ELSE
        SET @strMonth = CONVERT(varchar(20), YEAR(@dteStart)) +
            CONVERT(varchar(20), @iMonth);

    IF @iDay < 10
        SET @strDay = @strMonth + '0' +
            CONVERT(varchar(20), @iDay);
    ELSE
        SET @strDay = @strMonth + CONVERT(varchar(2), @iDay);

    SET @strTimeSheetCode = @EmplNbr + @strDay;
    RETURN @strTimeSheetCode;
END
GO

```

3- اضغط F5 لتنفيذ الاستعلام

4- لإدخال البيانات باستخدام View، أكتب الاستعلام التالي:

```

USE YugoNationalBank
GO
-- The following code performs data entry using a view
INSERT INTO dbo.TimeSheet

```



```
VALUES('46288', '1/1/2007',
      0.00, 8.50, 9.50, 8.50, 9.00, 0.00, 0.00,
      10.00, 9.50, 8.50, 10.50, 9.00, 0.00, 0.00, '');
GO
INSERT INTO dbo.TimeSheet
VALUES('66286', '1/1/2007',
      0.00, 8.50, 6.50, 5.50, 6.50, 0.00, 0.00,
      4.00, 6.00, 6.50, 6.00, 5.50, 0.00, 0.00, '');
GO
INSERT INTO dbo.TimeSheet
VALUES('92493', '1/1/2007',
      0.00, 8.00, 9.00, 8.50, 9.50, 0.00, 0.00,
      5.50, 6.50, 4.50, 6.00, 4.00, 0.00, 0.00, '');
GO
INSERT INTO dbo.TimeSheet
VALUES('27199', '1/15/2007',
      6.00, 8.50, 0.00, 4.00, 6.50, 0.00, 0.00,
      4.00, 0.00, 6.00, 4.00, 0.00, 0.00, 0.00, '');
GO
INSERT INTO dbo.TimeSheet
VALUES('39538', '1/15/2007',
      8.00, 8.00, 6.00, 8.50, 6.00, 0.00, 0.00,
      9.50, 10.50, 8.00, 8.00, 8.50, 0.00, 0.00,
      'There were a few missing times in the time sheet. ' +
      'They have been recorded.');
```

```
GO
INSERT INTO dbo.TimeSheet
VALUES('40550', '1/15/2007',
      8.50, 8.00, 0.00, 8.50, 0.00, 0.00, 0.00,
      6.00, 6.50, 6.50, 0.00, 4.00, 0.00, 0.00, '');
GO
INSERT INTO dbo.TimeSheet
VALUES('66286', '1/29/2007',
      8.00, 6.50, 9.50, 8.00, 7.50, 0.00, 0.00,
      10.50, 9.50, 8.50, 8.00, 10.00, 0.00, 0.00, '');
GO
INSERT INTO dbo.TimeSheet
VALUES('90026', '2/12/2007',
      8.50, 6.50, 8.00, 8.00, 9.50, 0.00, 0.00,
      9.50, 8.00, 8.50, 8.00, 8.00, 0.00, 0.00, '');
GO
INSERT INTO dbo.TimeSheet
VALUES('92493', '2/12/2007',
      4.00, 6.50, 5.50, 8.00, 6.50, 0.00, 0.00,
      8.00, 8.00, 8.00, 6.00, 8.00, 0.00, 0.00, '');
GO
-- The following code updates a table using a function
UPDATE dbo.TimeSheets
SET TimeSheetCode = dbo.CreateTimeSheetCode('46288', '1/1/2007')
WHERE (EmplNumber = '46288') AND (StartDate = '1/1/2007');
GO
UPDATE dbo.TimeSheets
SET TimeSheetCode = dbo.CreateTimeSheetCode('66286', '1/1/2007')
WHERE (EmplNumber = '66286') AND (StartDate = '1/1/2007');
GO
UPDATE dbo.TimeSheets
SET TimeSheetCode = dbo.CreateTimeSheetCode('92493', '1/1/2007')
WHERE (EmplNumber = '92493') AND (StartDate = '1/1/2007');
GO
UPDATE dbo.TimeSheets
SET TimeSheetCode = dbo.CreateTimeSheetCode('27199', '1/15/2007')
```

```
WHERE (EmplNumber = '27199') AND (StartDate = '1/15/2007');
GO
UPDATE dbo.TimeSheets
SET TimeSheetCode = dbo.CreateTimeSheetCode('39538', '1/15/2007')
WHERE (EmplNumber = '39538') AND (StartDate = '1/15/2007');
GO
UPDATE dbo.TimeSheets
SET TimeSheetCode = dbo.CreateTimeSheetCode('40550', '1/15/2007')
WHERE (EmplNumber = '40550') AND (StartDate = '1/15/2007');
GO
UPDATE dbo.TimeSheets
SET TimeSheetCode = dbo.CreateTimeSheetCode('66286', '1/29/2007')
WHERE (EmplNumber = '66286') AND (StartDate = '1/29/2007');
GO
UPDATE dbo.TimeSheets
SET TimeSheetCode = dbo.CreateTimeSheetCode('90026', '2/12/2007')
WHERE (EmplNumber = '90026') AND (StartDate = '2/12/2007');
GO
UPDATE dbo.TimeSheets
SET TimeSheetCode = dbo.CreateTimeSheetCode('92493', '2/12/2007')
WHERE (EmplNumber = '92493') AND (StartDate = '2/12/2007');
GO
```

5- اضغط F5 لتنفيذ الاستعلام

6- قم بحذف محتوى نافذة الاستعلام.

**استخدام View باسم مستعار:**

إن كائن View لا يمثل دالة SQL Server لكنه يمكن أن يستخدم كدالة. والمعلومات هنا لا تعني قيم يمكن تمريرها إلى View لكن هذه القيم ستكون محددة فقط عند إنشاء الكائن View، فهي ليست معلومات حقيقية.

عند إنشاء View، يمكنك إنشاء خانات لتمرير قيمة الحقول بين قوسين. وفق التالي:

```
CREATE VIEW CarIdentifier([Tag #], Manufacturer, [Type of Car],
Available)
...
```

هذه التقنية تسمح لك بتحديد أسماء الحقول وفق اختيارك. ويمكنك أيضا استخدام الاسم الفعلي للحقول، إذا كان اسم كلمة واحدة يمكنك كتابته مباشرة، وإذا كان يتضمن عدة كلمات يجب ضمها بين معقوفتين [].

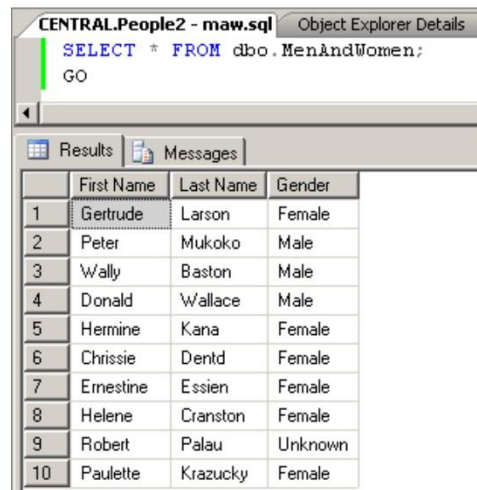
بعد كتابة أسماء الحقول، يجب استخدام عدد المعلومات بنفس عدد الحقول، حيث تمثل كل معلومة أحد الحقول على الترتيب، مثال:

```
CREATE VIEW dbo.MenAndWomen([First Name], [Last Name], Gender)
AS
```



```
SELECT dbo.Persons.FirstName,
       dbo.Persons.LastName,
       dbo.Genders.Gender
FROM   dbo.Genders INNER JOIN dbo.Persons
ON     dbo.Genders.GenderID = dbo.Persons.GenderID;
GO
```

ولأن (كما ذكرنا سابقاً) كائن View لا يمثل دالة SQL Server حيث لا يمكن تمرير قيم مباشرة إلى View، فإن تنفيذ View لا يتطلب تحديد قيم المعلمات، ويتم تنفيذها فقط بكتابة استعلام SELECT، مثال:



The screenshot shows a SQL query window with the following text:

```
SELECT * FROM dbo.MenAndWomen;
GO
```

The results are displayed in a table with the following data:

	First Name	Last Name	Gender
1	Gertrude	Larson	Female
2	Peter	Mukoko	Male
3	Wally	Baston	Male
4	Donald	Wallace	Male
5	Hermine	Kana	Female
6	Chrissie	Dentd	Female
7	Ernestine	Essien	Female
8	Helene	Cranston	Female
9	Robert	Palau	Unknown
10	Paulette	Krazucky	Female

#### كائن View والمعايير:

من أهم خصائص الاستعلام أنه يتميز بالمرونة، حيث يمكنك الوصول إلى أية نتيجة من خلال بيانات معقدة عن طريق معالجة البيانات باستخدام الشروط. يمكن استخدام الشروط لعرض محتوى View بدلاً من الجداول التي من شأنها تعقيد الاستعلام أكثر. يمكنك عند إنشاء View ضمن عبارة SELECT، تحديد الحقول وفق ترتيب معين، وإضافة معايير لاستبعاد بعض السجلات. مثال:

CENTRAL.Peopl...jects\maw.sql Object Explorer Details

```
SELECT * FROM dbo.MenAndWomen
WHERE (Gender = 'Male') OR (Gender = 'Female')
GO
```

Results Messages

	First Name	Last Name	Gender
1	Gertrude	Larson	Female
2	Peter	Mukoko	Male
3	Wally	Baston	Male
4	Donald	Wallace	Male
5	Hermine	Kana	Female
6	Chrissie	Dentd	Female
7	Ernestine	Essien	Female
8	Helene	Cranston	Female
9	Paulette	Krazucky	Female

### كائن View والتوابيع:

يمكن إنشاء View أكثر تعقيداً أو ذات وظائف متقدمة، ويمكن إشراك التوابيع المبرمجة كما يمكن استخدام التوابيع المضمنة مع SQL Server.

إذا لم تكن التوابيع المضمنة تفي بالغرض، يمكنك إنشاء تابع خاص بك. مثال:

```
USE People;
GO
CREATE FUNCTION dbo.GetFullName
(
    @FName varchar(20),
    @LName varchar(20)
)
RETURNS varchar(41)
AS
BEGIN
    RETURN @LName + ', ' + @FName;
END
GO
```

بعد إنشاء التابع الذي تريد، يمكنك إدراجه في جسم استعلام View إذا لزم الأمر. مثال:

```
CREATE VIEW dbo.MyPeople
AS
SELECT dbo.GetFullName(FirstName, LastName) AS [Full Name],
       dbo.Genders.Gender
FROM   Genders INNER JOIN dbo.Persons
ON     dbo.Genders.GenderID = dbo.Persons.GenderID;
```

CENTRAL.Peopl... dbo.MyPeople		Object Explorer Details
	Full Name	Gender
▶	Larson, Gertrude	Female
	Mukoko, Peter	Male
	Baston, Wally	Male
	Wallace, Donald	Male
	Kana, Hermine	Female
	Dentd, Chrissie	Female
	Essien, Ernestine	Female
	Cranston, Helene	Female
	Palau, Robert	Unknown
	Krazucky, Paulette	Female
*	NULL	NULL

تطبيق عملي: استخدام تابع ضمن View

1- من متصفح الكائنات، تحت YugoNationalBank قم بتوسيع Views إذا لزم الأمر، ثم انقر بالزر الأيمن PayrollPreparation واختر:

Script View As -> ALTER To -> New Query Editor Window

2- اكتب الاستعلام التالي:

```
USE [YugoNationalBank]
GO
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
ALTER VIEW [dbo].[PayrollPreparation]
AS
SELECT  dbo.Employees.EmployeeNumber,
        dbo.Employees.LastName + ', ' +
        dbo.Employees.FirstName AS [Full Name],
        dbo.Employees.HourlySalary,
        dbo.TimeSheets.TimeSheetCode, dbo.TimeSheets.Week1Monday,
        dbo.TimeSheets.Week1Tuesday, dbo.TimeSheets.Week1Wednesday,
        dbo.TimeSheets.Week1Thursday, dbo.TimeSheets.Week1Friday,
        dbo.TimeSheets.Week1Saturday, dbo.TimeSheets.Week1Sunday,
        dbo.TimeSheets.Week2Monday, dbo.TimeSheets.Week2Tuesday,
        dbo.TimeSheets.Week2Wednesday, dbo.TimeSheets.Week2Thursday,
        dbo.TimeSheets.Week2Friday, dbo.TimeSheets.Week2Saturday,
        dbo.TimeSheets.Week2Sunday,

        CAST(dbo.TimeSheets.Week1Monday AS SmallMoney) +
        CAST(dbo.TimeSheets.Week1Tuesday AS SmallMoney) +
        CAST(dbo.TimeSheets.Week1Wednesday AS SmallMoney) +
        CAST(dbo.TimeSheets.Week1Thursday AS SmallMoney) +
        CAST(dbo.TimeSheets.Week1Friday AS SmallMoney) +
        CAST(dbo.TimeSheets.Week1Saturday AS SmallMoney) +
```

```

CAST(dbo.TimeSheets.Week1Sunday AS SmallMoney)
AS [Total Week1],
CAST(dbo.TimeSheets.Week2Monday AS SmallMoney) +
CAST(dbo.TimeSheets.Week2Tuesday AS SmallMoney) +
CAST(dbo.TimeSheets.Week2Wednesday AS SmallMoney) +
CAST(dbo.TimeSheets.Week2Thursday AS SmallMoney) +
CAST(dbo.TimeSheets.Week2Friday AS SmallMoney) +
CAST(dbo.TimeSheets.Week2Saturday AS SmallMoney) +
CAST(dbo.TimeSheets.Week2Sunday AS SmallMoney)
AS [Total Week2],
CAST(dbo.TimeSheets.Week1Monday AS SmallMoney) +
CAST(dbo.TimeSheets.Week1Tuesday AS SmallMoney) +
CAST(dbo.TimeSheets.Week1Wednesday AS SmallMoney) +
CAST(dbo.TimeSheets.Week1Thursday AS SmallMoney) +
CAST(dbo.TimeSheets.Week1Friday AS SmallMoney) +
CAST(dbo.TimeSheets.Week1Saturday AS SmallMoney) +
CAST(dbo.TimeSheets.Week1Sunday AS SmallMoney) +
CAST(dbo.TimeSheets.Week2Monday AS SmallMoney) +
CAST(dbo.TimeSheets.Week2Tuesday AS SmallMoney) +
CAST(dbo.TimeSheets.Week2Wednesday AS SmallMoney) +
CAST(dbo.TimeSheets.Week2Thursday AS SmallMoney) +
CAST(dbo.TimeSheets.Week2Friday AS SmallMoney) +
CAST(dbo.TimeSheets.Week2Saturday AS SmallMoney) +
CAST(dbo.TimeSheets.Week2Sunday AS SmallMoney)
AS [Total Time Sheet]
FROM   dbo.Employees INNER JOIN dbo.TimeSheets
ON      dbo.Employees.EmployeeNumber = dbo.TimeSheets.EmplNumber;

```

3- لتحديث الاستعلام، من القائمة الرئيسية انقر على Query ثم Execute

4- احذف محتوى نافذة الاستعلام

5- لمشاهدة نتيجة الاستعلام، اكتب ما يلي:

```

Use YugoNationalBank;
GO
USE YugoNationalBank;
GO
SELECT PayrollPreparation.* FROM PayrollPreparation;
GO

```

6- احذف الاستعلام.

CENTRAL.YugoN...IQuery15.sql\* Object Explorer Details

```

Use YugoNationalBank;
GO
USE YugoNationalBank;
GO
SELECT PayrollPreparation.* FROM PayrollPreparation;
GO

```

Results Messages

	Empl...	Full Name	HourlySalary	TimeSheetCode	Week1Mond...	Week1Tuesd...
1	46288	Marconi, Catherine	22.55	4628820070101	0.00	8.50
2	27199	Ankoma, Leonie	14.88	2719920070115	6.00	8.50
3	66286	Zbrniz, Lienev	15.75	6628620070101	0.00	8.50
4	66286	Zbrniz, Lienev	15.75	6628620070129	8.00	6.50
5	39538	Waste, Plant	16.75	3953820070115	8.00	8.00
6	92493	Roberts, Kirsten	18.05	9249320070101	0.00	8.00
7	92493	Roberts, Kirsten	18.05	9249320070212	4.00	6.50
8	90026	Lamy, Roger	10.24	9002620070212	8.50	6.50
9	40550	Kirkland, Lorraine	12.86	4055020070115	8.50	8.00

```
USE [prev]

GO

/***** Object: Trigger [dbo].[ins_trig]  Script Date: 16/11/2022 10:46:14 AM *****/

SET ANSI_NULLS ON

GO

SET QUOTED_IDENTIFIER ON

GO

ALTER trigger [dbo].[ins_trig] on [dbo].[adminview] instead of insert

as

declare @previd int


if exists (select * from privilege where prevname in (select pname from
inserted))

select @previd=previd from privilege where prevname=(select pname from inserted)

else

begin

select @previd=isnull(max(previd)+1,1) from privilege

insert into privilege values(@previd,(select pname from inserted))

end


declare @roleid int


if exists (select * from roles where rolename in (select rname from
inserted))
```

```
select @roleid=roleid from roles where rolename=(select rname from inserted)
else
begin
select @roleid=isnull(max(roleid)+1,1) from roles
insert into roles values(@roleid,(select rname from inserted))
end
insert into prevrole values(@roleid,@previd)
```

```
USE [prev]
GO
/***** Object: Trigger [dbo].[deltrig]  Script Date: 16/11/2022 10:46:13 AM *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
ALTER trigger [dbo].[deltrig] on [dbo].[adminview] instead of delete
as
delete from prevrole where roleid in(select rid from deleted)
```



```
SELECT    dbo.privilage.prevname AS pname, dbo.roles.rolename AS rname, dbo.roles.roleid AS rid
FROM      dbo.privilage INNER JOIN
          dbo.prevrole ON dbo.privilage.previd = dbo.prevrole.previd INNER JOIN
          dbo.roles ON dbo.roles.roleid = dbo.prevrole.roleid
```

```
USE [prev]
```

```
GO
```

```
/***** Object: Table [dbo].[privilage]  Script Date: 16/11/2022 10:45:59 AM *****/
```

```
SET ANSI_NULLS ON
```

```
GO
```

```
SET QUOTED_IDENTIFIER ON
```

```
GO
```

```
CREATE TABLE [dbo].[privilage](
    [previd] [int] NOT NULL,
    [prevname] [nvarchar](50) NULL,
    CONSTRAINT [PK_privilage] PRIMARY KEY CLUSTERED
```

```
(
```

```
    [previd] ASC
```

```
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
```

```
) ON [PRIMARY]
```

```
GO
```

USE [prev]

GO

/\*\*\*\*\* Object: Table [dbo].[roles] Script Date: 16/11/2022 10:46:03 AM \*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[roles](

[roleid] [int] NOT NULL,

[rolename] [nvarchar](50) NULL,

CONSTRAINT [PK\_roles] PRIMARY KEY CLUSTERED

(

[roleid] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF,  
ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

USE [prev]

GO

/\*\*\*\*\* Object: Table [dbo].[prevrole] Script Date: 16/11/2022 10:45:56 AM \*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[prevrole](

[roleid] [int] NOT NULL,

[previd] [int] NOT NULL,

CONSTRAINT [PK\_prevrole] PRIMARY KEY CLUSTERED

(

[roleid] ASC,

[previd] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF,  
ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

ALTER TABLE [dbo].[prevrole] WITH CHECK ADD CONSTRAINT [FK\_prevrole\_privilage] FOREIGN  
KEY([previd])

REFERENCES [dbo].[privilage] ([previd])

GO

ALTER TABLE [dbo].[prevrole] CHECK CONSTRAINT [FK\_prevrole\_privilage]

GO

```
ALTER TABLE [dbo].[prevrole] WITH CHECK ADD CONSTRAINT [FK_prevrole_roles] FOREIGN  
KEY([roleid])
```

```
REFERENCES [dbo].[roles] ([roleid])
```

```
GO
```

```
ALTER TABLE [dbo].[prevrole] CHECK CONSTRAINT [FK_prevrole_roles]
```

```
GO
```

```
=====
```

```
USE [trigtrial]
```

```
GO
```

```
/***** Object: Table [dbo].[sec] Script Date: 16/11/2022 10:47:29 AM *****/
```

```
SET ANSI_NULLS ON
```

```
GO
```

```
SET QUOTED_IDENTIFIER ON
```

```
GO
```

```
CREATE TABLE [dbo].[sec](
```

```
    [id] [int] NOT NULL,
```

```
    [username] [nvarchar](50) NULL,
```

```
    [pw] [nvarchar](50) NULL,
```

```
    [role] [nvarchar](50) NULL,
```

```
    CONSTRAINT [PK_sec] PRIMARY KEY CLUSTERED
```

```
(
```

```
    [id] ASC
```

```
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,  
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
```

) ON [PRIMARY]

GO

USE [trigtrial]

GO

/\*\*\*\*\* Object: Trigger [dbo].[del\_tri] Script Date: 16/11/2022 10:47:38 AM \*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

ALTER trigger [dbo].[del\_tri] on [dbo].[sec] instead of delete as

declare @id int

declare @username nvarchar(50)

declare @c int

select @id =id,@username=username from deleted

select @c=isnull(count(\*),0) from sec where role ='admin'

if @c<=2

print ('error @c less than 2')

else if @username='admin'

print ('error')

else

delete from sec where id =@id

```
USE [trigtrial]
GO
/***** Object: Trigger [dbo].[new_ins_tri]  Script Date: 16/11/2022 10:47:39 AM *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
ALTER trigger [dbo].[new_ins_tri] on [dbo].[sec]
instead of insert
as
if (exists (select * from sec inner join inserted on
sec.username=inserted.username and sec.role=inserted.role))
print ('error in insert')
else
insert into sec select id,username,pw,role from inserted
```