

٢. بعض الأنواع الأخرى:

- **Binary**: لتخزين البيانات الثنائية Binary Data .
- **Image**: ويستعمل لتخزين البيانات من نوع بايت، وأيضا لتخزين الصور.
- **Bit**: يسمح فقط بتخزين القيمتين 0 و 1 ، وهو يستعمل غالبا حينما نتعامل مع بيانات منطقية .

خاصيات الإدخال: وهي مجموعة من الأوامر التي نطبقها على الحقول، من أجل التحقق من القيمة المراد حفظها، وأول هذه الكلمات هي:

- **NOT NULL**: وتستعمل هذه الخاصية لمنع ترك قيمة حقل معين فارغة، وصيغتها هكذا:

```
CREATE TABLE MyTable (
```

```
ID INT NOT NULL,
```

```
FullName VARCHAR(60))
```

- **IDENTITY**: هذا النوع من الكلمات يطبق فقط على الحقول التي يكون نوعها رقميا من أجل جعل قيمها تزداد تلقائيا عند إضافة أي سطر جديد، مثلا لو

أنشأت الجدول التالي وطبقت هذه الكلمة على الحقل ID فسوف تزداد قيمته

تصاعديا:

```
CREATE TABLE MyTable (  
    ID INT IDENTITY(1,1),  
    FullName VARCHAR(60))
```

عند إضافة أي سطر جديد، ستلاحظ بأن خانة الحقل ID تمنعك من الكتابة فيها، ولكن بمجرد ما تتجاوزها تقوم برفع قيمة الحقل بواحد تلقائيا، كما تبين الصورة التالية:

	ID	FullName
	1	Khalid
	2	Ahmed
	3	Youssef
▶*	NULL	NULL

تستطيع تغيير القيمة التي يبدأ منها الحقل، وأيضا معامل زيادة القيمة هكذا:

```
CREATE TABLE MyTable (  
    ID INT IDENTITY(2,3),  
    FullName VARCHAR(60))
```

في هذا المثال، ستبدأ قيمة الحقل من الرقم 2 ، ويكون معامل الزيادة هو 3 ، أي أن القيمة الأولى ستكون 2 ، والقيمة الثانية ستكون 5 ، والثالثة ستكون ... 8 إلخ.

- **PRIMARY KEY**: تطبق هذه الخاصية على الحقل لجعل قيمته متفردة، غير قابلة للتكرار ولا ينبغي أن تترك قيمة الحقل الذي تطبق عليه هذه الخاصية فارغة أي ينبغي أن تكون Not Null ، فمثلا لو عندنا جدول يضم بيانات المواطنين، فمن غير شك هنالك حقل يميز كل مواطن عن غيره، في هذه الحالة الحقل الذي سيكون مفتاحا أساسيا PRIMARY KEY هو الحقل الذي يضم أرقام بطاقات التعريف الوطنية، لأنه من الممكن أن تجد أكثر من مواطن يحملون نفس الاسم، ولكن يستحيل أن يحملوا نفس رقم بطاقة التعريف، عند التعامل مع جداول متصلة فيما بينها عن طريق العلاقات يصبح إلزاميا استخدام المفاتيح الأساسية.

لجعل أحد الحقول مفتاحا أساسيا، نكتب الصيغة التالية:

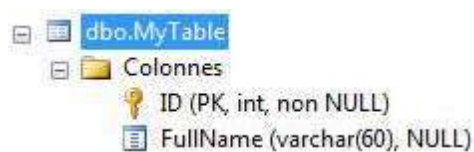
```
CREATE TABLE MyTable (
```

```
    ID INT PRIMARY KEY,
```

```
    FullName VARCHAR(60))
```

لو ذهبت إلى الجدول في القائمة، ودخلت إلى الأعمدة Columns ستجده بالشكل

التالي:



رمز المفتاح للدلالة على أن الحقل ID حقل أساسي PRIMARY KEY.

- **UNIQUE**: تستعمل هذه الخاصية لمنع قيمة حقل معين من التكرار، وبالتالي

لا يمكن أن يضم نفس الحقل قيمة أكثر من مرة، الفرق بين الخاصية UNIQUE

وبين الخاصية PRIMARY KEY أن هذه الأخيرة لا يمكن تطبيقها على

الحقول التي تسمح بالقيم الفارغة NULL بينما تسمح الخاصية UNIQUE

بالقيمة NULL، ويمكننا تطبيق هذه الخاصية على الحقل هكذا:

```
CREATE TABLE MyTable (
```

```
    ID INT UNIQUE,
```

```
    FullName VARCHAR(60))
```

- **REFERENCE**: هذه الخاصية تطبق على الحقول الأجنبية القادمة من

جدول آخر، وتستعمل لتعريف الحقل الأجنبي وتحديد الجدول القادم منه. في

هذا المثال سننشئ جدولين، ونحاول تطبيق الخاصية REFERENCE على

الجدول الثاني:

```
CREATE TABLE MyTable1 (
```

```
    ID INT NOT NULL PRIMARY KEY ,
```

```
    FullName VARCHAR(60))
```

```

CREATE TABLE MyTable2 (
    Code INT NOT NULL PRIMARY KEY ,
    ID INT CONSTRAINT PK_MyTable2 REFERENCES
    MyTable1(ID)
)

```

الجدول الثاني يتكون من حقلين، الأول اسمه Code وهو الحقل الأساسي، والثاني اسمه وهو أجنبي قادم من الجدول الأول. لو ذهبنا إلى الجدول في القائمة، ودخلنا إلى الأعمدة Columns ستجده بالشكل التالي:



المفتاح الذهبي للدلالة على أن الحقل أساسي، PRIMARY KEY والمفتاح الرمادي للدلالة على أن هذا الحقل أجنبي FOREIGN KEY.

• الجداول Tables :

الجدول عبارة عن وحدات لتخزين البيانات على شكل مصفوفة ثنائية الأبعاد تتكون من صفوف وأعمدة .

انشاء جدول :