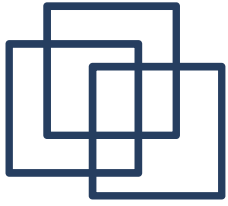


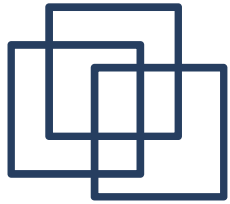
SQL como DDL

Definición de tablas
y restricciones básicas



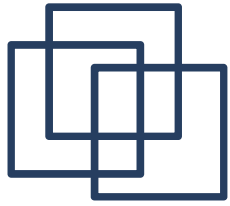
Tipos de datos

- Numéricos exactos
- Numéricos aproximados
- Fecha y hora
- Cadenas de caracteres
- Cadenas de caracteres Unicode
- Cadenas binarias
- Otros tipos de datos



Numéricos exactos (I)

Tipo de datos	Intervalo	Almacenamiento
bigint	De -2^{63} (-9.223.372.036.854.775.808) a $2^{63}-1$ (9.223.372.036.854.775.807)	8 bytes
int	De -2^{31} (-2.147.483.648) a $2^{31}-1$ (2.147.483.647)	4 bytes
smallint	De -2^{15} (-32.768) a $2^{15}-1$ (32.767)	2 bytes
tinyint	De 0 a 255	1 byte



Numéricos exactos (II)

•decimal y numeric

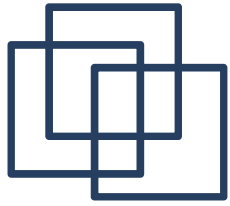
–Tipos de datos numéricos que tienen precisión y escala fijas

–**decimal**[(*p*[,*s*)]] y **numeric**[(*p*[,*s*)]]

–**P** indica el número total de cifras significativas y **s** el número de cifras decimales

•bit

–Tipo de datos entero que puede aceptar los valores 1, 0 o NULL

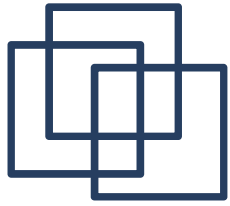


Numéricos exactos (III)

money y smallmoney

- Tipos de datos que representan valores monetarios o de moneda

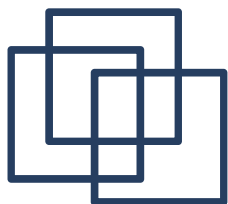
Tipo de datos	Intervalo	Almacenamiento
money	De -922.337.203.685.477,5808 hasta 922.337.203.685.477,5807	8 bytes
smallmoney	De - 214.748,3648 a 214.748,3647	4 bytes



Numéricos aproximados

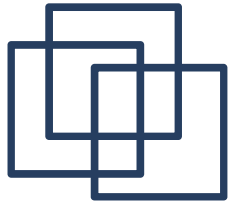
float y real

Tipo de datos	Intervalo	Almacenamiento
float	De $-1,79\text{E}+308$ a $-2,23\text{E}-308$, 0 y de $2,23\text{E}-308$ a $1,79\text{E}+308$	Depende del valor de n .
real	De $-3,40\text{E} + 38$ a $-1,18\text{E} - 38$, 0 y de $1,18\text{E} - 38$ a $3,40\text{E} + 38$	4 bytes



Fecha y hora

Tipo de datos	Intervalo	Precisión
datetime	Del 1 de enero de 1753 hasta el 31 de diciembre de 9999	Se redondea en incrementos de 0,000, 0,003 o 0.007 segundos
datetime2	Del 1 de enero del año 1 D.C. al 31 de diciembre de 9999	100 nanosegundos
datetimeoffset	Del 1 de enero del año 1 D.C. al 31 de diciembre de 9999	100 nanosegundos
smalldatetime	Del 1 de enero del año 1900 al 6 de junio de 2079	1 minuto
date	Del 1 de enero del año 1 D. C. al 31 de diciembre de 9999	1 día
time	De 00:00:00.0000000 a 23:59:59.9999999	100 nanosegundos



Cadenas de caracteres

- char (n)

- Cadena de longitud fija

- varchar (n)

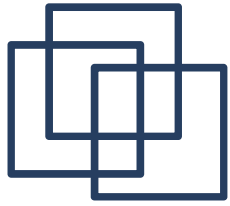
- Cadena de longitud variable

- n entre 1 y 8.000

- varchar (max)

- Hasta $2^{32}-1$ bytes

- Sustituye a text



Cadenas de caracteres Unicode

- `nchar (n)`

- Cadena de longitud fija

- `nvarchar (n)`

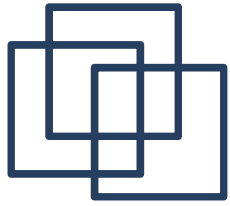
- Cadena de longitud variable

- n entre 1 y 4.000

- `nvarchar (max)`

- Hasta $2^{32}-1$ bytes

- Sustituye a text



Cadenas binarias

- binary (n)

- Datos binarios de longitud fija

- varbinary (n)

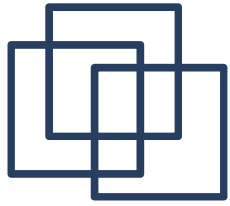
- Datos binarios de longitud variable

- n entre 1 y 8.000

- varbinary (max)

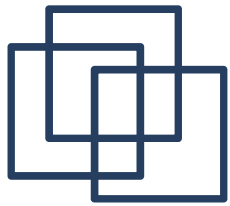
- Hasta $2^{32}-1$ bytes

- Sustituye a image



Otros tipos de datos

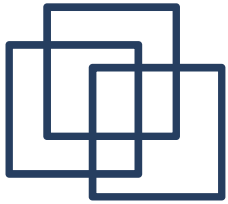
- cursor
- SQL variant
- table
- timestamp
- uniqueidentifier
- XML
- hierarchyid



Crear una tabla

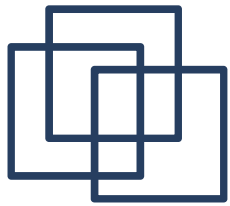
COCHES(MATRICULA, MARCA, MODELO,
CILINDRADA, PRECIO, FECHA_COMPRA)

```
CREATE TABLE COCHES (  
    Matricula char(7) Constraint  
    PK_Coches Primary Key  
    ,Marca VarChar(15)Not Null  
    ,Modelo VarChar(20) Not Null  
    ,Cilindrada decimal (2,1) Null  
    ,Precio SmallMoney Not Null  
    ,Fecha_Compra Date Null  
)
```



Otro ejemplo

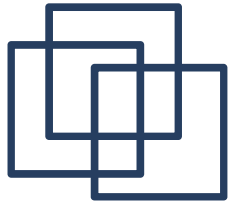
```
CREATE TABLE LM_VENTAS(  
Codigo_Venta SMALLINT NOT NULL  
,Fecha DATE NOT NULL  
,ID_Cliente INT NULL DEFAULT 1  
,Precio_Venta SMALLMONEY NOT NULL  
,CONSTRAINT PK_Ventas PRIMARY KEY  
(Codigo_venta)  
  
,CONSTRAINT FK_Ventas_Clientes FOREIGN KEY  
(ID_Cliente) REFERENCES LM_CLIENTES(ID) ON  
DELETE NO ACTION ON UPDATE CASCADE  
)
```



Columnas identificadoras

Si queremos que el servidor genere los valores de la PK automáticamente podemos usar columnas IDENTITY

```
CREATE TABLE Clientes(  
  IDCliente INT Not NULL IDENTITY (1,1)  
  ,Nombre NVarchar(20) Not NULL  
  ...  
  ,CONSTRAINT PK_Clientes PRIMARY KEY  
  (IDClientes)  
)
```



Columnas identificadoras

Otra forma es declarar nuestra columna con un tipo especial

```
CREATE TABLE Clientes(  
  IDCliente UniqueIdentifier Not NULL  
  ,Nombre NVarchar(20) Not NULL  
  ...  
  ,CONSTRAINT PK_Clientes PRIMARY KEY  
  (IDClientes)  
)
```

Los valores se generan con la función **NEWID()**