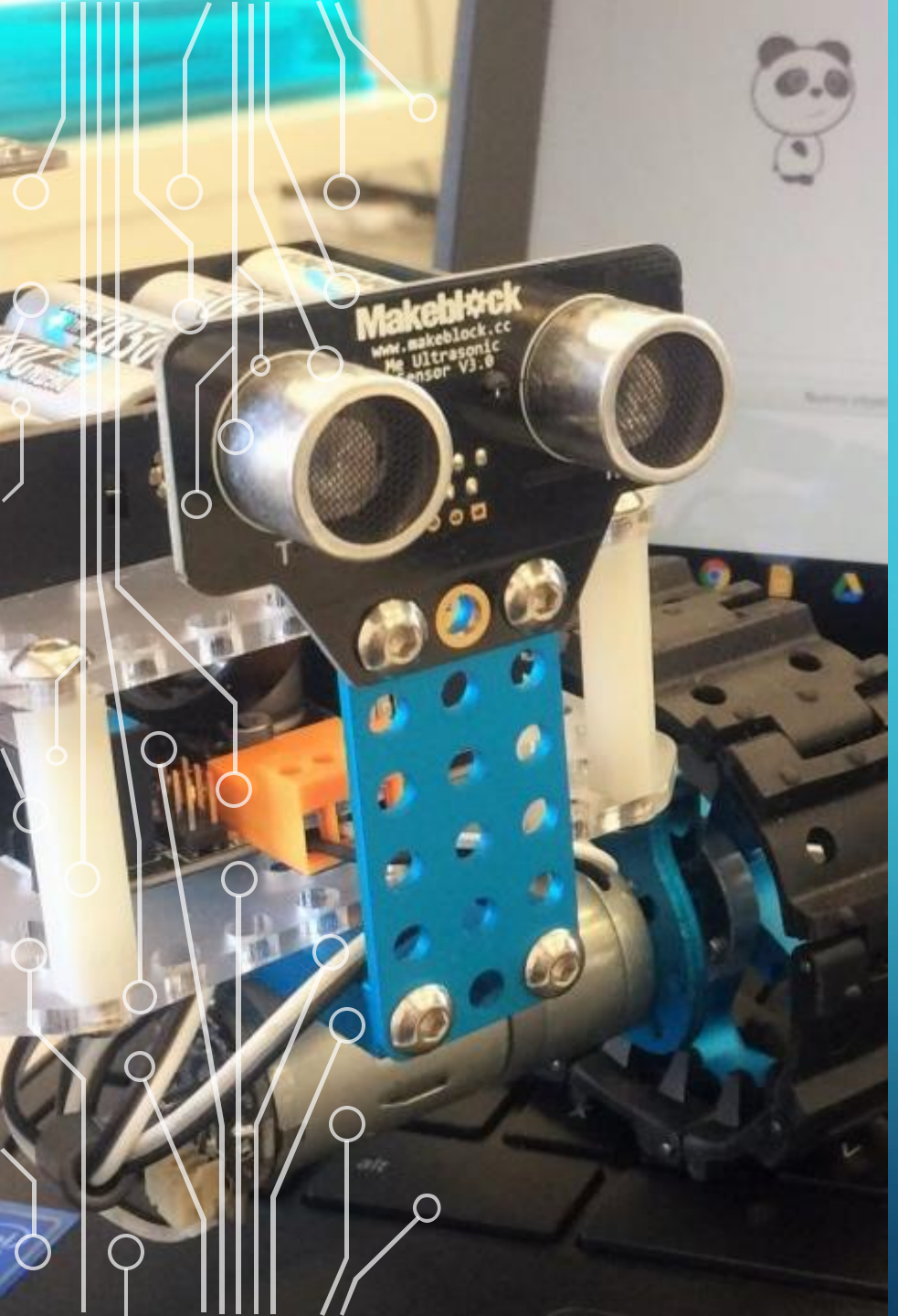


TEMA 6: T-SQL PROGRAMANDO CON SQL

BERNAT COSTA

BERNAT.COSTA@CESURFORMACION.COM

¿QUÉ ES T-SQL?

T-SQL es una extensión a SQL que nos dará herramientas para incluir :

- programación procidemental (condicionales,bucles..)
- Variables locales.
- Manejo de strings (replace...),
- procesamiento de fechas (DateAdd, day(),month(),getdate()…)
- Operaciones Matemáticas .

¿QUÉ PODEMOS HACER CON T-SQL?

Guiones (Scripts)

Métodos o funciones

Triggers

¿ES ESTO LO MISMO QUE PL/SQL?

- En muchas ofertas de trabajo se pide PL/SQL. ¿Es esto lo mismo?
- NO. PL/SQL es el lenguage que se usa en ORACLE
- Nosotros aprenderemos a usar T-SQL, que es el "PL/SQL de Microsoft"
- Es muy parecido, aunque con algunas diferencias.

T-SQL O EL TRANSACT-SQL PERMITE:

Tipos de datos

- Los que ya teniamos, int, datetime,varchar...

Variables

- Nombre que le damos a cierto valor

Estructuras de control

- Condicionales
- Bucles

Gestión de excepciones

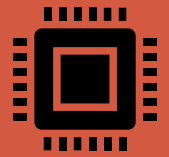
- Que hacer si ocurre un error

Funciones predefinidas

- ALgunas nos las da SQL Server, otras las podremos "hacer" nosotr@s.

SIN EMBARGO NO PERMITE

Crear interfaces de usuario.



Crear aplicaciones ejecutables, serán aplicaciones para lanzar desde el motor de bbdd.

BLOQUES DE CÓDIGO

En T-SQL los bloques de código se traducen por:

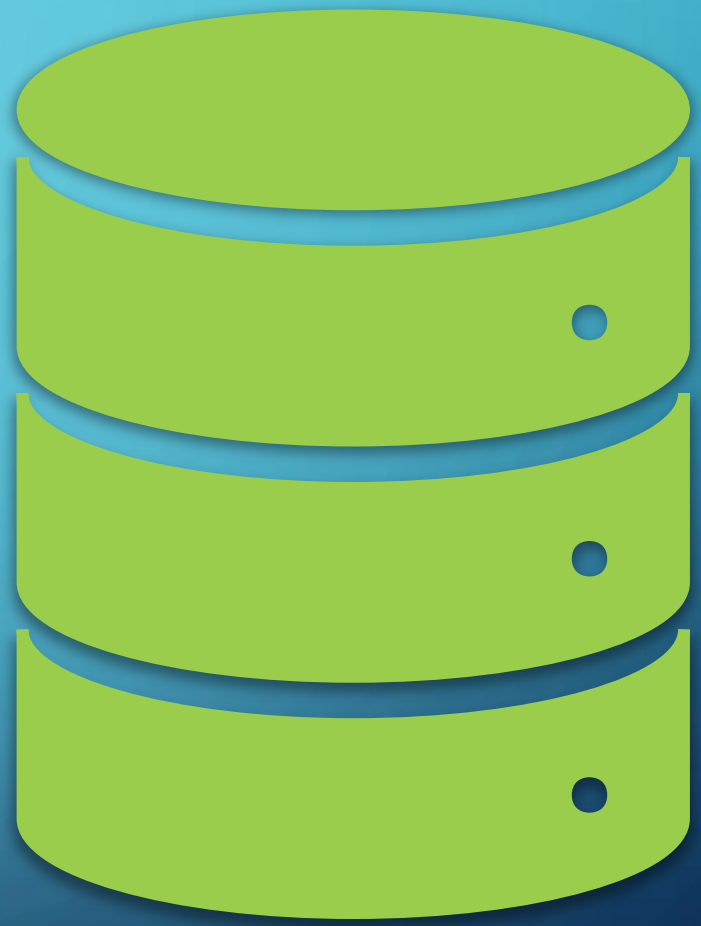
BEGIN

...

END

Dentro podremos ponerle todas las instrucciones que queramos.

Son nuestras {} de Java.



DECLARACIÓN DE VARIABLES

- Las variables se tienen que declarar y setear.
- el nombre empieza SIEMPRE con @
- Se declaran con el comando

DECLARE @nombre as varchar(100)

- Las variables son volatiles. No se guardan en la BBDD. Se crean y destruyen al ejecutar nuestros scripts o métodos.

USAR UNA VARIABLE

Las variables se pueden setear con Select o con set.

```
DECLARE @fecha as DateTime

SET @fecha=getdate()

SELECT @fecha=getdate()

PRINT @fecha
```

SETEAR DESDE UNA CONSULTA

Podemos setear una variable desde una consulta.

```
DECLARE @idmax as int
SET @idmax =
(SELECT max(id)
FROM menu)
```

```
DECLARE @idmax as int
SELECT
@idmax=max(id)
FROM menu
```

¿CÓMO DEVOLVEMOS UN VALOR?

- Podemos guardarlo en una tabla con un update.
- Podemos "pintarlo" en la pantalla.

Si tenemos:

```
DECLARE @fecha as DATETIME
SET @fecha = getdate()
```

Podemos hacer:

PRINT @fecha
Para devolver
en formato texto

SELECT @fecha
Para devolver
una tabla

ESTRUCTURAS DE CONTROL

CONDICIONALES

BUCLES

CONDICIONALES

CASE WHEN

Se puede poner en medio de una consulta

IF ELSE

Para definir dos bloques diferenciados con BEGIN END

¿QUÉ ES UN CONDICIONAL?

Es una estructura de control que nos servirá para decidir entre 2 o más opciones según una condición.

Si se cumple esto, haz esto, sino, haz esto otro.

En programación también tenemos el SWITCH. Aquí no lo vamos a usar. Es prescindible.

CASE WHEN THEN ELSE END

```
CREATE  TABLE Personas (nombre varchar(100),apellido  varchar(100),edad  int))
INSERT INTO  Personas SELECT  'Pedro','Rodriguez',41

INSERT INTO  Personas SELECT  'Adam','Rodriguez',10
INSERT INTO  Personas SELECT  'Pau','Rodriguez',7

INSERT INTO  Personas SELECT  'Laia','Rodriguez',4
```

```
SELECT nombre, apellido,

CASE WHEN edad>18 THEN 'SI' ELSE 'NO'  END as mayorEdad

FROM Personas
```

IF ELSE

```
DECLARE @num as INT
SET @num=5
BEGIN

IF @num>4
  PRINT 'es mayot que 4'
ELSE
  PRINT 'NO es mayor que 4'
```

Si dentro de la condición necesitamos escribir más de una linea, lo encapsulamos con BEGIN END

```
DECLARE @num as INT
SET @num=5
BEGIN

IF @num>4
  BEGIN
    DECLARE @num2 as INT
    SET @num2=@num
    PRINT 'es mayor que 4'
  END
ELSE
  BEGIN
    Print 'NO es mayor que 4'
  END
```

BUCLES

- Un bucle es un fragmento de código que se repite hasta que nosotros decidamos.
- En programación, hay muchos tipos de bucles
  - While, for, do while...

•Nosotros solo vamos a ver el WHILE. Todo programa que se puede hacer con un bucle se puede hacer con un while.

WHILE

- El WHILE consta de dos partes:
  - Condición de salida
  - Lineas que se van a repetir hasta que se cumpla la condición de salida.
    - Estas lineas, deben contener una modificación de la variable que se comprueba para salir.

WHILE condicion

BEGIN

....

END

EJEMPLO: SCRIPT PARA CALCULAR EL FACTORIAL DE 6

```
DECLARE @resultado as int =1
DECLARE @num as INT
SET @num=6

BEGIN
    WHILE @num<>0   --condición de salida
    BEGIN
        SET @resultado = @resultado * @num
        SET @num = @num-1    -- linea para modificar la condición
    END
    PRINT @resultado
END
```

SQLCMD, SQL SERVER DESDE LA LINEA DE COMANDOS

- **sqlcmd** -S localhost -U alumnadobbdd -P 123456Ab$ -i fichero.sql

docker exec -it sql-server-db /opt/mssql-tools/bin/sqlcmd -S localhost –U sa -P 12345Ab## -Q "USE AREPAZO; SELECT nombre,precioventa from menu "

- Nos servirá para lanzar scripts desde la consola de windows o linux contra SQL Server.
  - -S -> server name
  - -U -> user name
  - -P -> password
  - -i -> ruta fichero script.sql


That's all Folks!