

The background is split into two main sections. The left section features a purple-to-blue gradient with a complex network of colorful lines (red, blue, green, yellow) forming a 3D-like geometric structure. Overlaid on this are white circuit-like patterns consisting of lines and small circles. The right section is a solid blue gradient with white circuit-like patterns in the corners.

TEMA 5: MODIFICANDO NUESTROS DATOS

BERNAT COSTA

BERNAT.COSTA@CESURFORMACION.COM

DML DATA MANIPULATION LANGUAGE

Consultar	Select
Insertar	Insert
Modificar	Update
Borrar	Delete

VAMOS A MODIFICAR DATOS

- Tendremos que tener mucho cuidado.
- Nos debe "temblar" el dedo antes de darle f5.
- No siempre tendremos un .bak para volver al punto anterior.

GUARDAR LA PARTIDA

BEGIN TRAN

Instrucción SQL para modificar/borrar/insertar datos

ROLLBACK TRAN

Si lanzamos estas lineas juntas, NO MODIFICAMOS la BBDD, pero sí vemos las lineas afectadas.

PROBAD SIEMPRE VUESTROS DELETES/UPDATES

A decorative graphic on the left side of the slide, consisting of a network of light blue lines and small circles, resembling a circuit board or a stylized tree structure.

1º TATUAJE DEL CURSO

NO QUIERO VER UN DELETE/UPDATE SIN UN BEGIN
TRAN/ROLLBACK TRAN

INSERTAR DATOS

Comando INSERT

3 formas de usarlo.

- Con valores puestos a pelo
- Insertando datos de otra consulta
- Insertando desde un csv.

INSERT CON VALORES A "PELO"

Muy útil si queremos insertar una fila, o insertar fila a fila.

INSERT INTO nombretabla (campo1,campo2,.....)

VALUES ('valor1','valor2',....)

Ejemplo:

INSERT INTO personas (personaid,apellido,nombre,direccion,ciudad,edad,altura)

VALUES('2','Costa','Miquel','calle falsa 23','Barcelona',37,183)

INSERT DESDE OTRA CONSULTA

Desde una SELECT...

```
INSERT INTO personas  
SELECT lastname,firstname, address,city,age from Person.Person
```

Insertará tantas filas, como resultados de la consulta (select...).

Muy útil para insertar datos masivamente.

¿Y LOS IDS IDENTITY'S?

- Si nuestra tabla tiene un campo autoincremental (normalmente el id) NO DEBEREMOS AÑADIRLO.
- "nos saltamos" ese campo.

EJEMPLO

Con Identity

- INSERT INTO personas (apellido,nombre,direccion,ciudad,edad)
VALUES('costa','Miquel','calle','Barcelona',37,183)

Sin Identity

- INSERT INTO personas (personaid,apellido,nombre,direccion,ciudad,edad)
VALUES('2','costa','Miquel','calle','Barcelona',37,183)

INSERTAR DESDE UN CSV

```
BULK INSERT equipos FROM '/var/opt/mssql/backup/futbol/clubs.csv'  
WITH (  
    FIELDTERMINATOR = ',' , ROWTERMINATOR = '0x0a'  
)
```

La ruta después del FROM le indicamos la ruta DENTRO del contenedor o servidor donde va a encontrar el CSV.

MODIFICAR UN CAMPO

A diferencia de con el DDL, no modificaremos un campo de int a varchar

Cambiaremos el contenido una tabla.

También podemos hacerlo de varias formas:

- Modificar un dato a pelo
- Modificar un dato desde una consulta

MODIFICAR DATOS A "PELO"

- Comando **UPDATE** tabla **SET** campo='nuevovalor' **WHERE** ...
- Ejemplo:

```
UPDATE personas SET ciudad = 'Barcelona'  
WHERE ciudad='BCN'
```

IMPORTANTE PONER EL WHERE. Sino, modificaríamos TODAS las filas de la tabla poniendo ciudad='Barcelona'

UPDATE DESDE UNA CONSULTA

- Podemos enlazar una consulta para modificar un campo desde un valor de esa consulta.
- Ejemplo:

```
UPDATE pedidos
SET SubTotal =
(
  SELECT sum(cantidad*PrecioVentaUnitario)
  FROM pedidos p
  INNER JOIN PedidosLinea l on p.Id=l.IdPedido
  WHERE p.id=1
)
WHERE id=1
```

MAS EJEMPLOS DE UPDATES

UPDATE PEDIDOS

SET total= subtotal+gastosEnvio

WHERE id=32

UPDATE menu

SET descripcion = REPLACE(descripcion,'baso','vaso')

WHERE descripcion like '%baso%'

UPDATE PedidosLinea **SET** iva=t.IVA

FROM PedidosLinea l

INNER JOIN Menu m ON m.id=l.IdMenu

INNER JOIN TipoAlimento t on t.id = m.Tipo

WHERE 1=1

BORRADO DE DATOS

- El temible DELETE.
- Sirve para borrar datos.
- Borraremos tantos datos como queramos y le indiquemos en el where.
- NO TE OLVIDES DE PONER EL WHERE EN EL DELETE

DELETE FROM personas **WHERE** ciudad='Barcelona'

COMANDO IN EN EL WHERE

- Es muy útil en el DELETE el where con un IN y una subconsulta
- Podemos localizar todos los Id's con una subconsulta y ponerla en un IN para borrar varios registros automáticamente.

- **DELETE FROM** menu **WHERE** id **IN** (
SELECT id FROM menu m
LEFT JOIN pedidoslinea l on l.idmenu=m.id
WHERE l.idpedido IS NULL
)

2º TATUAJE DEL CURSO

- JAMÁS ME OLVIDO DE PONER EL WHERE EN EL DELETE FROM
- Si quiero borrar la tabla entera, escribiré:
DELETE FROM personas WHERE 1=1
- SIEMPRE PONDRÉ EL WHERE.

EN EL UPDATE,
también escribiré
SIEMPRE
el where.

SIEMPRE.

AUNQUE SEA
 $1=1$

EL WHERE TAMBIÉN EN EL UPDATE





Y POR SI NO OS HA QUEDADO CLARO...

NO TE OLVIDES DE PONER EL WHERE EN EL DELETE FROM

HIMNO DE LA ASIGNATURA

- https://www.youtube.com/watch?v=i_cVJglz_Cs



Y CON TODO ESTO QUE PODEMOS HACER?

- Podemos enlazar consultas con inserts
- Con deletes
- Con updates.
- Podemos usar todo el potencial de las consultas, para modificar datos.



EJERCICIO

- Borra las películas de netflix que no ve nadie
- Borra del Arepazo, los elementos del menú que nadie compra
- Borra la bbdd de las olimpiadas, los jugadores españoles que no han ganado ni una medalla.



EN EL GITHUB DE LA ASIGNATURA....

- Mirar el script `nba.sql` para crear la BBDD de la NBA.
- Analizarlo y interpretar lo que hace.



That's all Folks!