

Unidad de Trabajo 5
Tratamiento de los datos

Manipulación de datos

IES Palomeras Vallecas
Curso 2020/2021
Profesor: Alberto Ruiz

Tratamiento de los datos

- Hasta ahora hemos estudiado cómo recuperar los datos que queremos
- Ahora veremos cómo manipular los datos:
 - Insertando nuevos registros (filas): INSERT
 - Modificando algún dato de registros ya existentes: UPDATE
 - Eliminando registros: DELETE

Insertar nuevos registros

Utilizamos INSERT INTO:

```
INSERT INTO ciclo (Código, Nombre, Tipo) VALUES  
( 'SMR', 'Sistemas Microinformáticos y Redes', 'Medio');
```

El orden de los campos será el que indiques entre paréntesis:

```
INSERT INTO ciclo (Tipo, Nombre, Código) VALUES  
( 'Medio', 'Sistemas Microinformáticos y Redes', 'SMR');
```

Si no se indican los campos se asume que se incluirán valores para todos, en el orden en que se definieron al crear la tabla:

```
INSERT INTO ciclo VALUES  
( 'SMR', 'Sistemas Microinformáticos y Redes', 'Medio');
```

Puedes utilizar **NULL** como valor (sin comillas, de lo contrario será el texto “NULL”, siempre que el campo no esté marcado como obligatorio (NOT NULL))

```
INSERT INTO estudiantes VALUES  
( '23423545J', 'Sonia García', 'España', '2008-04-12', NULL, 'Normal');
```

Insertar nuevos registros

Es posible insertar más de una fila de una vez:

```
INSERT INTO ciclo VALUES  
( 'SMR', 'Sistemas Microinformáticos y Redes', 'Medio'),  
( 'ASIR', 'Administración de Sistemas Informáticos en Red', 'Superior');
```

- También es posible insertar datos que vienen de otra tabla
- Las dos tablas deben tener campos compatibles (el mismo número, el mismo orden, los mismos tipos de datos)

```
INSERT INTO ciclo  
SELECT * FROM nuevosCiclos
```

Modificar registros

```
UPDATE empleados  
SET ciudad = "Madrid";
```

```
UPDATE empleados  
SET sueldo = sueldo * 2, horas = horas - 1;
```

- Podemos usar filtros con WHERE:

```
UPDATE empleados  
SET sueldo = sueldo * 2  
WHERE antigüedad > 20
```

Eliminar registros

```
DELETE  
FROM ciclo;
```

- DELETE sin filtro es peligroso porque borra todos los datos de la tabla
 - Nunca se elimina la tabla, sólo los datos (se queda vacía)

```
DELETE  
FROM ciclo  
WHERE tipo = "Medio";
```

```
DELETE  
FROM estudiante  
WHERE NIF= "0123532A";
```

Eliminar o modificar un registro

- Siempre que quieras modificar o eliminar un registro, **localízalo por su clave primaria**
 - Si utilizas otro campo, puede que haya confusión y modifiques o borres el registro equivocado

```
DELETE
FROM jugadores
WHERE Nombre = "Marcus Williams";
```

Código	Nombre	Procedencia	Altura	Peso	Posición	NombreEquipo
35	Marcus Williams	Arizona	6-7	207	F	Clippers
473	Marcus Williams	Connecticut	6-3	205	G	Nets

```
DELETE
FROM jugadores
WHERE Código=35;
```

```
UPDATE jugadores
SET Peso=210
WHERE Código=35;
```

UPDATE y DELETE en subconsultas

- Es posible actualizar o borrar registros de una tabla filtrando a través de una subconsulta
- Ejemplo: Borrar los representantes de ventas que no tengan clientes:

```
DELETE
FROM empleados
WHERE Puesto='Representante Ventas'
      AND CodigoEmpleado NOT IN (SELECT CodigoEmpleadoRepVentas
                                   FROM clientes);
```

- El borrado se hace en la tabla "empleados", y se usa la tabla "clientes" para filtrar

UPDATE y DELETE en subconsultas

- **No** es posible actualizar o borrar registros en la misma tabla que se está consultando para filtrar.
- Ejemplo: **Borrar los clientes con límite de crédito 0:**

```
DELETE
FROM clientes
WHERE CodigoCliente IN (SELECT CodigoCliente
                        FROM clientes
                        WHERE LimiteCredito=0);
```

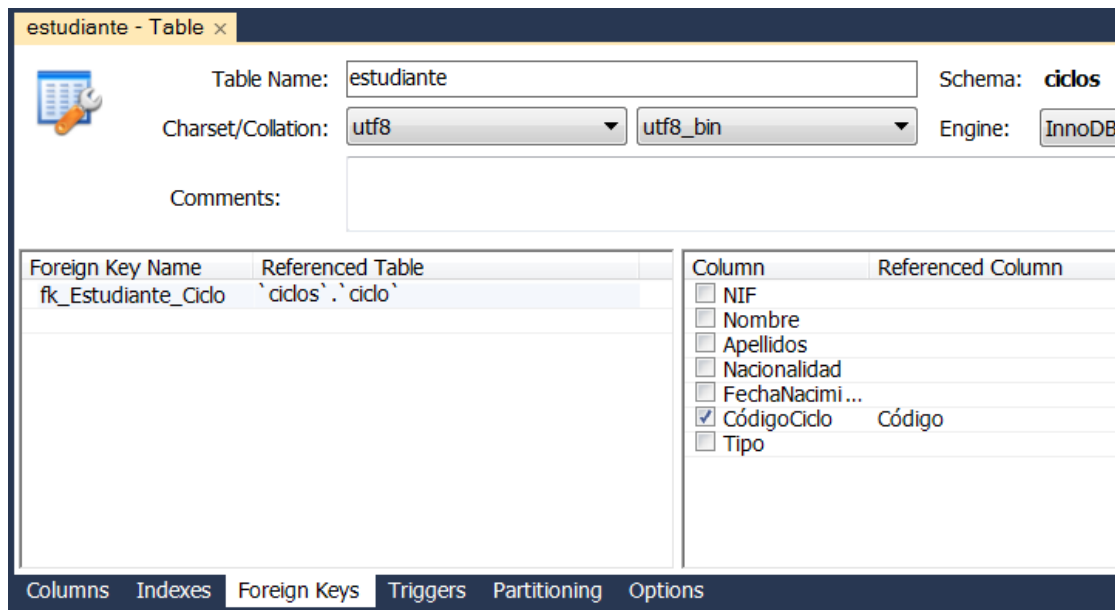
```
DELETE
FROM empleados
WHERE Puesto='Representante Ventas'
      AND CodigoEmpleado NOT IN (SELECT CodigoEmpleadoRepVentas
                                FROM clientes);
```

UPDATE y DELETE de claves ajenas

- El borrado y la modificación de campos que se usan como clave ajena es conflictivo y no siempre se podrá hacer, ya que afecta a otras tablas.
- Ejemplos:
 - Si eliminamos un cliente pero tenía pagos pendientes, ¿qué pasa con esos pagos?
 - Si eliminamos un producto pero aparece en la tabla de pedidos, ¿qué pasa con esos pedidos?
 - Si cambiamos el código de un ciclo,
 - ¿Qué pasa con el campo de la tabla estudiante en la que se indica el código del ciclo que estudia?
 - ¿Y qué pasa en la tabla donde se asocian códigos de ciclo a códigos de módulo?

Recordando la clave ajena

- Si analizas la estructura de la tabla "estudiante", verás una pestaña de claves ajenas (Foreign Keys).
- Debes localizar en la imagen los tres datos fundamentales:
 - Referenced table: "ciclo"
 - Column "CódigoCiclo"
 - Referenced Column "Código"



Reglas de comportamiento

- En esa ventana verás las reglas de comportamiento, que deben indicarse siempre para claves ajenas
 - ON DELETE: qué hacer si se **borra** el código de un ciclo en la tabla "ciclo"
 - ON UPDATE: qué hacer si se **modifica** el código de un ciclo en la tabla "ciclo"

The screenshot shows the MySQL Table Editor for the 'estudiante' table in the 'ciclos' schema. The table is using the InnoDB engine with utf8 charset and utf8_bin collation. The 'Foreign Keys' tab is selected, showing a foreign key named 'fk_Estudiante_Ciclo' that references the 'ciclo' table in the 'ciclos' schema. The foreign key is defined on the 'CodigoCiclo' column of the 'estudiante' table, which references the 'Codigo' column of the 'ciclo' table. The 'Foreign Key Options' section shows the 'On Update' and 'On Delete' actions set to 'RESTRICT'. A mouse cursor is hovering over the 'CASCADE' option in the 'On Delete' dropdown menu.

Foreign Key Name	Referenced Table
fk_Estudiante_Ciclo	`ciclos`.`ciclo`

Column	Referenced Column
<input type="checkbox"/> NIF	
<input type="checkbox"/> Nombre	
<input type="checkbox"/> Apellidos	
<input type="checkbox"/> Nacionalidad	
<input type="checkbox"/> FechaNacimi...	
<input checked="" type="checkbox"/> CódigoCiclo	Código
<input type="checkbox"/> Tipo	

Foreign Key Options

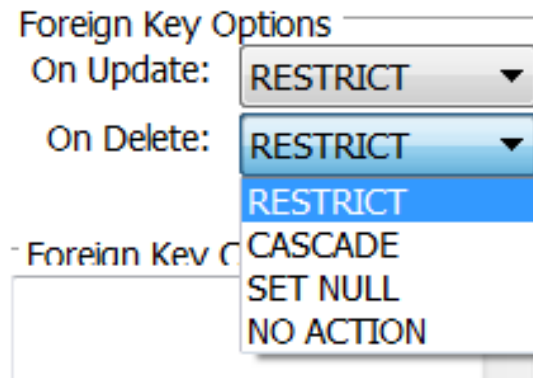
On Update: RESTRICT

On Delete: RESTRICT

Foreign Key Options: RESTRICT, CASCADE, SET NULL, NO ACTION

Reglas de comportamiento

- Posibles valores para las reglas de comportamiento:
 - RESTRICT o NO ACTION
 - SET NULL
 - CASCADE



Reglas de comportamiento

- **RESTRICT o NO ACTION:** no se permite el borrado o la modificación de claves primarias que están siendo utilizadas como clave ajena en esta tabla. Es la opción más conservadora.
- **SET NULL:** al desaparecer o modificar la clave primaria, se asigna NULL a la clave ajena
 - Si borro un ciclo o cambio su código, el estudiante que lo estudiaba tendrá NULL en el campo CódigoCiclo
 - ¡Esto sólo funcionará si el campo CódigoCiclo permite nulos!
- **CASCADE:** se propaga el borrado o la modificación a la clave ajena
 - Si borro un ciclo, se **eliminan** todos los estudiantes que lo estudiaban
 - Si modifico un código de ciclo, se modifica el campo CódigoCiclo en los estudiantes que lo estudiaban

Reglas de comportamiento

- Es habitual escoger valores diferentes para borrado y modificación. Por ejemplo:

Foreign Key Options

On Update:

On Delete:

- Si **modifico** el código de "ASI" a "ASIR" con UPDATE:
 - El cambio se propaga en cascada, y el campo CódigoCiclo de los estudiantes que estudian el ciclo cambia automáticamente a "ASIR" para mantener la coherencia de los datos
- Si **elimino** el ciclo ASIR con DELETE:
 - No lo permito si existen alumnos que lo estudian. En ese caso se obtendrá el error *"no se cumple una restricción de clave ajena"*

DELETE

FROM ciclo

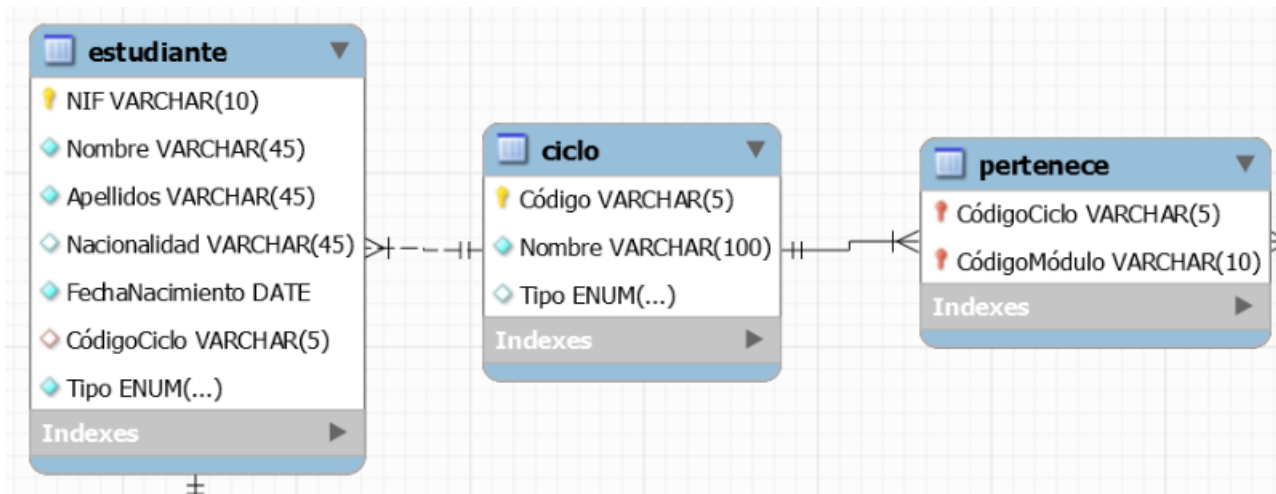
WHERE código = "ASIR"

Message

Error Code: 1451. Cannot delete or update a parent row: a foreign key constraint fails

Reglas de comportamiento

- ¿Dónde defino qué ocurre al eliminar o modificar el código de un ciclo?



- En la tabla **estudiante**, al definir la clave ajena CódigoCiclo
- En la tabla **pertenece**, al definir la clave ajena CódigoCiclo

¡En la tabla ciclo no tienes que hacer nada!