
	Bases de datos Tema 6		Práctica
	El lenguaje SQL II – Tratamiento de los datos		2

Tarea 1: Creación base de datos.

Ejecuta las siguientes instrucciones y resuelve las cuestiones que se plantean en cada paso:

```

SET AUTOCOMMIT = 0;
SELECT @@AUTOCOMMIT;

DROP DATABASE IF EXISTS tema6practica2;
CREATE DATABASE tema6practica2 CHARACTER SET utf8mb4;
USE tema6practica2;

CREATE TABLE productos(
    id INT UNSIGNED AUTO_INCREMENT PRIMARY KEY,
    nombre VARCHAR(100) NOT NULL,
    precio DOUBLE
);

INSERT INTO productos (id, nombre) VALUES (1, 'Primero');
INSERT INTO productos (id, nombre) VALUES (2, 'Segundo');
INSERT INTO productos (id, nombre) VALUES (3, 'Tercero');

```

1. Comprobar que se han insertado las filas de forma correcta.

```
SELECT * FROM productos;
```



Captura:

```

mysql> Select * from productos;
+----+-----+-----+
| id | nombre | precio |
+----+-----+-----+
| 1  | Primero | NULL   |
| 2  | Segundo | NULL   |
| 3  | Tercero | NULL   |
+----+-----+-----+
3 rows in set (0.00 sec)

```

2. Vamos a simular que perdemos la conexión con el servidor antes de que la transacción sean completada. Para simular que perdemos la conexión desde *MySQL Workbench* hay que cerrar la pestaña de conexión con el

	Bases de datos Tema 6		Práctica
	El lenguaje SQL II – Tratamiento de los datos		2

servidor. Si estás conectado al servidor desde la consola de MySQL sólo tienes que ejecutar el comando EXIT.

Volvemos a conectar con el servidor y ejecutamos las siguientes instrucciones:

```
USE tema6practica2;
SELECT * FROM productos;
```

Captura: ¿Qué devolverá la consulta?

```
mysql> USE tema6practica2;
Database changed
mysql> SELECT * FROM productos;
Empty set (0.00 sec)
```

Tarea 2: Transacciones.



Ejecuta las siguientes instrucciones y resuelve las cuestiones que se plantean en cada paso.

```
SET AUTOCOMMIT = 1;
SELECT @@AUTOCOMMIT;

DROP DATABASE IF EXISTS tema6practica2;
CREATE DATABASE tema6practica2 CHARACTER SET utf8mb4;
USE tema6practica2;

CREATE TABLE productos(
    id INT UNSIGNED AUTO_INCREMENT PRIMARY KEY,
    nombre VARCHAR(100) NOT NULL,
    precio DOUBLE
);

INSERT INTO productos (id, nombre) VALUES (1, 'Primero');
INSERT INTO productos (id, nombre) VALUES (2, 'Segundo');
INSERT INTO productos (id, nombre) VALUES (3, 'Tercero');
```

	Bases de datos Tema 6		Práctica
	El lenguaje SQL II – Tratamiento de los datos		2

1. ¿Qué devolverá esta consulta?.

```
SELECT * FROM productos;
```

Captura:

```
mysql> SELECT * FROM productos;
+----+-----+-----+
| id | nombre | precio |
+----+-----+-----+
| 1  | Primero | NULL   |
| 2  | Segundo | NULL   |
| 3  | Tercero | NULL   |
+----+-----+-----+
3 rows in set (0.00 sec)
```

2. Vamos a intentar deshacer la transacción actual.

```
ROLLBACK;
```

Captura:



```
mysql> ROLLBACK;
Query OK, 0 rows affected (0.00 sec)
```

3. ¿Qué devolverá esta consulta?

```
SELECT * FROM productos;
```

Justificar la respuesta:

El rollback solo revierte la ultima transacción realizada, al no tratarse de una transacción sino de una sentencia simple no es posible hacer el rollback

	Bases de datos Tema 6		Práctica
	El lenguaje SQL II – Tratamiento de los datos		2

Captura:

```
mysql> SELECT * FROM productos;
+----+-----+-----+
| id | nombre | precio |
+----+-----+-----+
| 1  | Primero | NULL   |
| 2  | Segundo | NULL   |
| 3  | Tercero | NULL   |
+----+-----+-----+
3 rows in set (0.00 sec)
```

4. Ejecutamos la siguiente transacción.

```
START TRANSACTION;
INSERT INTO productos (id, nombre) VALUES (4, 'Cuarto');



SELECT * FROM productos;
ROLLBACK;
```

5. ¿Qué devolverá esta consulta?

```
SELECT * FROM productos;
```

Justificar la respuesta:

La consulta no se ve afectada ya que aunque anteriormente hallamos introducido unos valores nuevos en la tabla, esta transacción se cancela debido al rollback y vuelve a ser como antes de la introducción de datos.

	Bases de datos Tema 6		Práctica
	El lenguaje SQL II – Tratamiento de los datos		2

Captura:

```
Alonso[(tema6practica2)]>SELECT * FROM productos;
+---+-----+-----+
| id | nombre | precio |
+---+-----+-----+
| 1  | Primero | NULL   |
| 2  | Segundo | NULL   |
| 3  | Tercero | NULL   |
+---+-----+-----+
3 rows in set (0.00 sec)
```

6. Ejecutamos la siguiente transacción.



```
INSERT INTO productos (id, nombre) VALUES (5, 'Quinto');
ROLLBACK;
```

7. ¿Qué devolverá esta consulta?.

```
SELECT * FROM productos;
```

Justificar la respuesta:

En este caso se mostrara la nueva columna creada ya que no es una transacción, por lo tanto el rollback no tiene efecto

	Bases de datos Tema 6		Práctica
	El lenguaje SQL II – Tratamiento de los datos		2

Captura:

```
Alonso[(tema6practica2)]>SELECT * FROM productos;
+----+-----+-----+
| id | nombre | precio |
+----+-----+-----+
| 1  | Primero | NULL   |
| 2  | Segundo | NULL   |
| 3  | Tercero | NULL   |
| 5  | Quinto  | NULL   |
+----+-----+-----+
4 rows in set (0.00 sec)
```

8. Desactivamos el modo AUTOCOMMIT y borramos el contenido de la tabla.

```
SET AUTOCOMMIT = 0;
SELECT @@AUTOCOMMIT;



DELETE FROM productos WHERE id > 0;
```

Captura:

```
Alonso[(tema6practica2)]>SET AUTOCOMMIT = 0;
Query OK, 0 rows affected (0.00 sec)

Alonso[(tema6practica2)]>SELECT @@AUTOCOMMIT;
+-----+
| @@AUTOCOMMIT |
+-----+
| 0             |
+-----+
1 row in set (0.00 sec)

Alonso[(tema6practica2)]>
Alonso[(tema6practica2)]>DELETE FROM productos WHERE id > 0;
Query OK, 4 rows affected (0.01 sec)
```

	Bases de datos Tema 6		Práctica
	El lenguaje SQL II – Tratamiento de los datos		2

9. Comprobamos que la tabla esta vacía.

```
SELECT * FROM productos;
```

Captura:



```
Alonso[(tema6practica2)]>SELECT * FROM productos;
Empty set (0.00 sec)
```

10. Insertamos dos filas nuevas

```
INSERT INTO productos (id, nombre) VALUES (6, 'Sexto');
INSERT INTO productos (id, nombre) VALUES (7, 'Séptimo');
SELECT * FROM productos;
```

Captura:

```
Alonso[(tema6practica2)]>SELECT * FROM productos;
+----+-----+-----+
| id | nombre | precio |
+----+-----+-----+
| 6  | Sexto  | NULL   |
| 7  | Séptimo | NULL   |
+----+-----+-----+
2 rows in set (0.00 sec)
```

	Bases de datos Tema 6		Práctica
	El lenguaje SQL II – Tratamiento de los datos		2

11. Hacemos un *ROLLBACK*.

```
ROLLBACK;
```

Captura:

```
Alonso[(tema6practica2)]>ROLLBACK;
Query OK, 0 rows affected (0.01 sec)
```



12. ¿Qué devolverá esta consulta?

```
SELECT * FROM productos;
```

Justificar la respuesta:

Debido al rollback se retira todo lo hecho anteriormente ya que estamos realizando una transacción y volvemos al mismo resultado de antes

Captura:

	Bases de datos Tema 6		Práctica
	El lenguaje SQL II – Tratamiento de los datos		2

```
Alonso[(tema6practica2)]>SELECT * FROM productos;
+----+-----+-----+
| id | nombre | precio |
+----+-----+-----+
| 1 | Primero | NULL |
| 2 | Segundo | NULL |
| 3 | Tercero | NULL |
| 5 | Quinto | NULL |
+----+-----+-----+
4 rows in set (0.00 sec)
```



13. Ejecutamos la siguiente transacción.

```
SET AUTOCOMMIT = 0;
START TRANSACTION;

CREATE TABLE fabricantes (id INT UNSIGNED);
INSERT INTO fabricantes (id) VALUES (1);

SELECT * FROM fabricantes;
ROLLBACK;
```

Captura:

	Bases de datos Tema 6		Práctica
	El lenguaje SQL II – Tratamiento de los datos		2

```

Alonso[(tema6practica2)]>SET AUTOCOMMIT = 0;
Query OK, 0 rows affected (0.00 sec)

Alonso[(tema6practica2)]>START TRANSACTION;
Query OK, 0 rows affected (0.00 sec)

Alonso[(tema6practica2)]>
Alonso[(tema6practica2)]>CREATE TABLE fabricantes (id INT UNSIGNED);
Query OK, 0 rows affected (0.04 sec)

Alonso[(tema6practica2)]>INSERT INTO fabricantes (id) VALUES (1);
Query OK, 1 row affected (0.00 sec)

Alonso[(tema6practica2)]>
Alonso[(tema6practica2)]>SELECT * FROM fabricantes;
+-----+
| id    |
+-----+
| 1     |
+-----+
1 row in set (0.00 sec)

Alonso[(tema6practica2)]>ROLLBACK;
Query OK, 0 rows affected (0.00 sec)

```

14. ¿Se puede hacer *ROLLBACK* de instrucciones de tipo DDL (*CREATE*, *ALTER*, *DROP*, *RENAME* y *TRUNCATE*)?

Justificar la respuesta:

No es posible puesto que el rollback solo será posible en el caso de las sentencias DML como es el caso de las transactions.