

Instrucciones

- Descargue el proyecto del examen del siguiente repositorio: <https://github.eii.us.es/IISSI-2223/examen-lab-iissi.git>.
- Prepare el proyecto creando la BD ModeloExamen en HeidiSQL, revisando settings.py, y ejecutando "silence createdb", "silence createapi", "silence createtests" y finalmente "silence run".
- Implemente la solución a los ejercicios en los archivos **sql/solucion.sql** (para código sql) y **tests/solucion.http** (para peticiones http). Estos ejercicios están basados en unos requisitos adicionales que se suministran a continuación.
- Incluya capturas de pantalla HeidiSQL/vsCode que muestre el resultado de ejecutar el ejercicio al final de este documento.
- Debe subir a la actividad de EV este documento con las capturas de pantallas, y los archivos sql/solucion.sql y tests/solucion.http. Antes de subir la actividad, avise a un profesor para que valide la misma.
- Es necesario completar un nivel para que se valoren los siguientes niveles.

Catálogo de requisitos

RI-1-01 Requisito de información

Como: Profesor de la asignatura

Quiero: Tener disponible información sobre **Multas**: Una multa es una penalización a un usuario por alguna causa. Sus atributos son: el usuario multado, la causa de la multa, un entero representando el nivel de la multa, la fecha de la multa, y si es moderada o grave. Todos los atributos son obligatorios.

Para: Evaluar al estudiante

RN-1-01 Primera regla de negocio

Como: Profesor de la asignatura

Quiero: Un usuario no puede tener varias multas el mismo día.

Para: Evaluar al estudiante

RN-1-02 Segunda regla de negocio

Como: Profesor de la asignatura

Quiero: El nivel debe estar entre 0 y 10 (inclusive).

Para: Evaluar al estudiante

RN-1-03 Tercera regla de negocio







Como: Profesor de la asignatura

Quiero: Una multa grave no puede tener un nivel inferior a 3.

Para: Evaluar al estudiante

PA- 1-01 Inserción de nuevas entidades

Como: Profesor de la asignatura

Quiero:  Insertar Multa con causa "Multa 1" del usuario con ID=2, de nivel 3, moderada, del 05/08/2020.
 Insertar Multa con causa "Multa 2" del usuario con ID=2, de nivel 3, moderada, del 06/09/1997.
 Multa con causa "Multa 3" del usuario con ID=1, de nivel 1 moderada, del 04/06/2016.
 Insertar Multa con causa "Multa 4" del usuario con ID=1, de nivel 1, grave, del 02/01/2090. (RN-1-03)
 Insertar Multa con causa "Multa 5" del usuario con ID=20, de nivel 10, grave, del 04/02/2020. (referencia no existente)
 Insertar Multa con causa "Multa 6" del usuario con ID=2, de nivel 1, moderada, del 05/08/2020. (RN-1-01)

Para: Evaluar al estudiante.

Nivel C. Máximo 7 puntos

1.(2)- Implemente los requisitos proporcionados (RI/RN) en MariaDB.

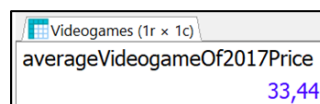
2.(1)- Use Insert para implementar las pruebas de aceptación. **Incluya captura de pantalla con los datos de la tabla después de las pruebas.**

Use Delete para eliminar el elemento con ID=1. **Incluya captura de pantalla con los datos de la tabla después de la llamada.**

Use Update para actualizar los elementos con IDs 2 y 3 para que tengan un nivel de 7. **Incluya captura de pantalla con los datos de la tabla después de las llamadas.**

3.(0.5)- Escriba y realice peticiones HTTP para obtener el listado de usuarios de la base de datos.

4.(1)- Cree una consulta que devuelva el precio medio de los videojuegos lanzados en 2017:



Videogames (1r x 1c)	
averageVideogameOf2017Price	33,44

5.(1.5)- Cree un disparador que, antes de borrarse una Multa, compruebe si el nivel es igual al máximo nivel de las multas existentes en la tabla, mostrando un mensaje de error con el valor que se intenta borrar. Pruebe el disparador. **Incluya captura de pantalla con los datos de la tabla después de ejecutar la llamada.**

6.(0.5)- Cree una función que, a partir del identificador de una Multa, devuelva la cantidad de multas en la tabla con el mismo nivel que dicha Multa (incluyéndose a ella misma). Use la función para obtener el resultado a partir del elemento con ID=2. Incluya capturas de pantallas.

(0.5) Cree un procedimiento que, a partir del identificador de una Multa, la borre solo si el resultado de la función anterior es igual a 2.

Nivel B. Máximo 9 puntos

7.(1)- Cree una consulta que devuelva la valoración media de los videojuegos de cada plataforma, ordenados por la valoración, de menor a mayor.

Videogames (7r x 2c)	
idPlatform	avgScore
8	1,0
6	6,3333
3	7,0
2	7,2
4	8,75
7	9,0
1	9,5714

8.(1)- Implemente las pruebas de aceptación como peticiones POST HTTP. **Recuerde limpiar los datos de la tabla insertados los apartados anteriores.** Las peticiones POST se pueden realizar con o sin autenticación. Incluya capturas de pantallas.

Nivel A. Máximo 10 puntos

9.(0.5)- Cree una consulta que devuelva el nombre de la plataforma con más videojuegos a partir de 2005, y ese número de videojuegos:

Platforms (1r x 2c)	
name	numberOfVideogames
PC	4

10.(0.5)- Cree un procedimiento **pAddTwoPenalties(...)** que, dentro de una transacción, inserte dos Multas a partir de los datos suministrados de los dos.

Realice dos llamadas: una que inserte dos correctamente, y una en la que el segundo rompa alguna restricción y aborte la transacción. Incluya capturas de pantallas

Capturas de pantalla

Inserte aquí las capturas de pantalla COMPLETA que muestre la ejecución de los ejercicios. Para realizar una captura de pantalla puede usar la tecla “impr pant” del teclado y pulsar “ctrl+v” para pegar la captura de pantalla.

Insert sin errores

```
19  CONSTRAINT invalidNivel CHECK (nivel >= 0 AND nivel <= 10),
20  CONSTRAINT invalidMultaGrave CHECK (esGrave = 0 OR (esGrave = 1 AND nivel >= 3))
21  );
22
23
24  DELIMITER //
25  CREATE OR REPLACE TRIGGER RN101_triggerFecha
26  BEFORE INSERT ON multas
27  FOR EACH ROW
28  BEGIN
29      DECLARE tieneMulta INT;
30      SET tieneMulta = (SELECT COUNT(*)
31                      FROM multas
32                      WHERE idUser = new.idUser AND fecha = new.fecha);
33      IF (tieneMulta > 0) THEN
34          SIGNAL SQLSTATE '45000' SET MESSAGE_TEXT =
35              'Solo puede haber una multa a cada usuario por dia';
36      END IF;
37  END //
38  DELIMITER ;
39
40  INSERT INTO multas (multaId, idUser, causa, nivel, fecha, esGrave) VALUES
41      (1, 2, 'Multa 1', 3, '2020-08-05', 0),
42      (2, 2, 'Multa 2', 3, '1997-09-06', 0),
43      (3, 1, 'Multa 3', 1, '2016-06-04', 0);
44  -- (4, 1, 'Multa 4', 1, '2090-01-02', 1);
45  -- (5, 20, 'Multa 5', 10, '2020-02-04', 1);
46  -- (6, 2, 'Multa 6', 1, '2020-08-05', 0);
47
48
49
50
```

305 /* Filas afectadas: 0 Filas encontradas: 0 Advertencias: 0 Duración para 3 de 4 consultas: 0,078 seg. */
306 /* Respuestas al examen de Laboratorio. ** Nombre del alumno: ** Grupo al que pertenece: */ -- A continuación debe escribir
307 CREATE TABLE multas (multaId INT NOT NULL AUTO_INCREMENT, idUser INT NOT NULL, causa VARCHAR(200) NOT NULL, nivel INT NOT NULL,
308 CREATE OR REPLACE TRIGGER RN101_triggerFecha BEFORE INSERT ON multas FOR EACH ROW BEGIN DECLARE tieneMulta INT;
309 INSERT INTO multas (multaId, idUser, causa, nivel, fecha, esGrave) VALUES (1, 2, 'Multa 1', 3, '2020-08-05', 0), (2, 2, 'Multa 2', 3, '1997-09-06', 0),
310 -- (4, 1, 'Multa 4', 1, '2090-01-02', 1); -- (5, 20, 'Multa 5', 10, '2020-02-04', 1); -- (6, 2, 'Multa 6', 1, '2020-08-05', 0);
311 /* Filas afectadas: 3 Filas encontradas: 0 Advertencias: 0 Duración para 5 consultas: 0,109 seg. */

error 1

HeidiSQL 12.3.0.6589

Host: 127.0.0.1 Base de datos: modeloexamen tables.sql solucion.sql*

Filter: Expresión regular

310 -- (4, 1, 'Multa 4', 1, '2090-01-02',1); -- (5, 20, 'Multa 5', 10, '2020-02-04',1); -- (6, 2, 'Multa 6', 1, '2020-08-05',0);
311 /* Filas afectadas: 3 Filas encontradas: 0 Advertencias: 0 Duración para 5 consultas: 0,109 seg. */
312 /* Respuestas al examen de Laboratorio. ** Nombre del alumno: ** Grupo al que pertenece: */ -- A continuación debe escribir
313 CREATE TABLE multas (multaId INT NOT NULL AUTO_INCREMENT, idUser INT NOT NULL, causa VARCHAR(200) NOT NULL, nivel INT NOT NULL,
314 CREATE OR REPLACE TRIGGER RN101_triggerFecha BEFORE INSERT ON multas FOR EACH ROW BEGIN DECLARE tieneMulta INT;
315 INSERT INTO multas (multaId, idUser, causa, nivel, fecha, esGrave) VALUES (1, 2, 'Multa 1', 3, '2020-08-05',0), (2, 2, 'Mult
316 /* Error de SQL (4025): CONSTRAINT 'invalidMultGrave' failed for 'modeloexamen'.multas` */

r44 : c1 (1,4 KiB) Conectado: 0 MariaDB 11.1.2 Activo durante: 33 días, 2 00:00:07.2: Ejecutando consul

Error 2

HeidiSQL 12.3.0.6589

Host: 127.0.0.1 Base de datos: modeloexamen tables.sql solucion.sql*

Filter ...

Column...

Funcion...

Palabra...

Frage...

Historia...

Perfi...

Pará...

Error de SQL (1452): Cannot add or update a child row: a foreign key constraint fails ('modeloexamen'.multas, CONSTRAINT 'multas_ibfk_1' FOREIGN KEY ('idUser') REFERENCES 'usersvideogames' ('idUser'))

Aceptar

Encontrar ayuda acerca de este error (=> ecosia.org)

```

19  CONSTRAINT invalidNivel CHECK (nivel >= 0 AND nivel <= 10),
20  CONSTRAINT invalidMultasGrave CHECK (esGrave = 0 OR (esGrave = 1 AND nivel >= 3))
21  );
22
23
24  DELIMITER //
25  CREATE OR REPLACE TRIGGER RN101_triggerFecha
26  BEFORE INSERT ON multas
27  FOR EACH ROW
28  BEGIN
29      DECLARE tieneMultas INT;
30      SET tieneMultas = (SELECT COUNT(*)
31                          FROM multas
32                          WHERE idUser = new.idUser AND fecha <= ROWTIME);
33      IF (tieneMultas > 0) THEN
34          SIGNAL SQLSTATE '45000' SET MESSAGE_TEXT = 'Solo puede haber una multa a la vez';
35      END IF;
36  END //
37  DELIMITER ;
38
39
40  INSERT INTO multas (multaId, idUser, causa, nivel, fecha, esGrave) VALUES
41  (1, 2, 'Multa 1', 3, '2020-08-05', 0),
42  (2, 2, 'Multa 2', 3, '1997-09-06', 0),
43  (3, 1, 'Multa 3', 1, '2016-06-04', 0),
44  -- (4, 1, 'Multa 4', 1, '2090-01-02', 1);
45  (5, 20, 'Multa 5', 10, '2020-02-04', 1);
46  -- (6, 2, 'Multa 6', 1, '2020-08-05', 0);
47
48
49
50

```

316 /* Error de SQL (4025): CONSTRAINT 'invalidMultasGrave' failed for 'modeloexamen'.multas */

317 /* Filas afectadas: 0 Filas encontradas: 0 Advertencias: 0 Duración para 3 de 5 consultas: 0,062 seg. */

318 /* Respuestas al examen de Laboratorio. ** Grupo al que pertenece: */ -- A continuación debe escribir:

319 CREATE TABLE multas (multaId INT NOT NULL AUTO_INCREMENT, idUser INT NOT NULL, causa VARCHAR(200) NOT NULL, nivel INT NOT NULL, fecha DATE, esGrave INT NOT NULL);

320 CREATE OR REPLACE TRIGGER RN101_triggerFecha BEFORE INSERT ON multas FOR EACH ROW BEGIN DECLARE tieneMultas INT;

321 INSERT INTO multas (multaId, idUser, causa, nivel, fecha, esGrave) VALUES (1, 2, 'Multa 1', 3, '2020-08-05', 0), (2, 2, 'Multa 2', 3, '1997-09-06', 0), (3, 1, 'Multa 3', 1, '2016-06-04', 0),

322 /* Error de SQL (1452): Cannot add or update a child row: a foreign key constraint fails ('modeloexamen'.multas, CONSTRAINT 'multas_ibfk_1' FOREIGN KEY ('idUser') REFERENCES 'usersvideogames' ('idUser'))

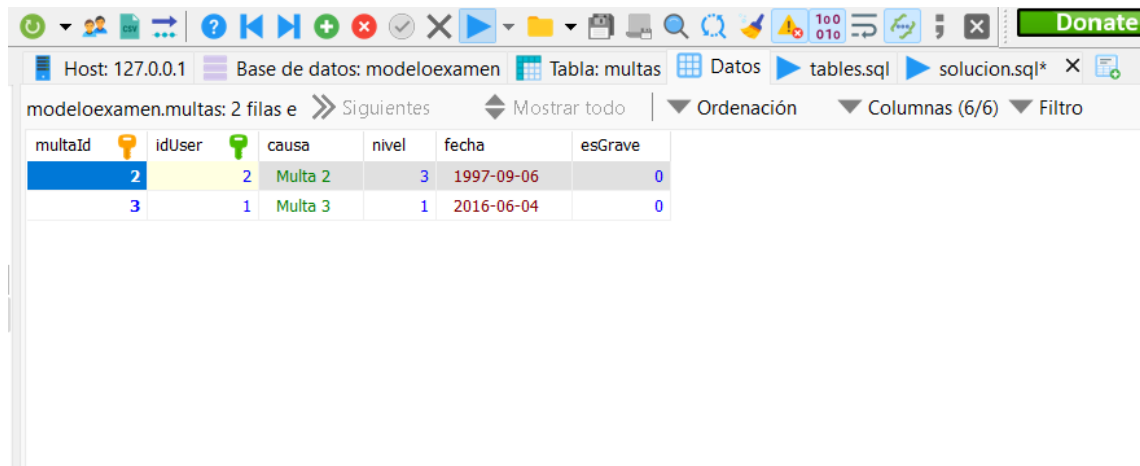
Ejecutar consulta(s) SQL... r45: c1 (1,4 KiB) Conectado: 0 MariaDB 11.1.2 Activo durante: 33 días, 2 00:00:21.2: Ejecutando consulta

Error 3

Datos de la tabla

multaId	idUser	causa	nivel	fecha	esGrave
1	2	Multa 1	3	2020-08-05	0
2	2	Multa 2	3	1997-09-06	0
3	1	Multa 3	1	2016-06-04	0

Datos tras borrado

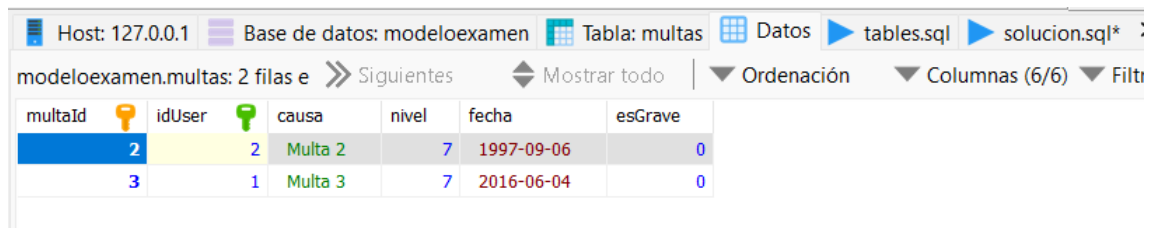


Host: 127.0.0.1 Base de datos: modeloexamen Tabla: multas Datos tables.sql solucion.sql*

modeloexamen.multas: 2 filas e >> Sigüientes Mostrar todo Ordenación Columnas (6/6) Filtro

multaId	idUser	causa	nivel	fecha	esGrave
2	2	Multa 2	3	1997-09-06	0
3	1	Multa 3	1	2016-06-04	0

Updates

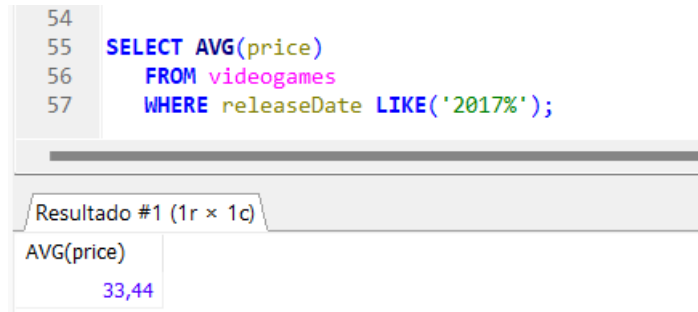


Host: 127.0.0.1 Base de datos: modeloexamen Tabla: multas Datos tables.sql solucion.sql*

modeloexamen.multas: 2 filas e >> Sigüientes Mostrar todo Ordenación Columnas (6/6) Filtro

multaId	idUser	causa	nivel	fecha	esGrave
2	2	Multa 2	7	1997-09-06	0
3	1	Multa 3	7	2016-06-04	0

Consulta 1



```
54
55 SELECT AVG(price)
56 FROM videogames
57 WHERE releaseDate LIKE('2017%');
```

Resultado #1 (1r x 1c)

AVG(price)
33,44

Trigger

```
39
40 INSERT INTO multas (multaId, idUser, causa, nivel, fecha, esGrave) VALUES
41 (1, 2, 'Multa 1', 3, '2020-08-05',0),
42 (2, 2, 'Multa 2', 3, '1997-09-06',0),
43 (3, 1, 'Multa 3', 1, '2016-06-04',0);
44 -- (4, 1, 'Multa 4', 1, '2090-01-02',1);
45 -- (5, 20, 'Multa 5', 10, '2020-02-04',1);
46 -- (6, 2, 'Multa 6', 1, '2020-08-05',0);
47
48 DELETE FROM multas
49 WHERE multaId = 1;
50
51
52 DELIMITER //
53 CREATE OR REPLACE TRIGGER RN001_triggerBorradoMulta
54 BEFORE DELETE ON multas
55 FOR EACH ROW
56 BEGIN
57     DECLARE maximaMulta INT;
58     SET maximaMulta = (SELECT MAX(nivel)
59                        FROM multas);
60     IF (maximaMulta = OLD.nivel) THEN
61         SET @error_message = CONCAT('Nivel incorrecto pues es maximo, nivel= ', maximaMulta);
62         SIGNAL SQLSTATE '45000' SET MESSAGE_TEXT = @error_message;
63     END IF;
64 END //
65 DELIMITER ;
66
67 DELETE FROM multas
68 WHERE nivel = 7;
69
```

IISSI_USER: Error

Error de SQL (1644): Nivel incorrecto pues es maximo, nivel= 7

Aceptar

Encontrar ayuda acerca de este error (=> [ecosia.org](https://www.ecosia.org/))