**Trabajo Práctico 2**

**Base de Datos 2**

**Integrantes:** Casermeiro Maria Silvia, Chiletti Emanuel, Vietto Santiago

Problema:

1. Crear una tabla 'tanques' que contenga:

nombre\_tanque VARCHAR(45) PK

nivel\_umbral\_delta DOUBLE

temperatura\_umbral\_delta DOUBLE

nivel\_umbral\_delta\_segundos INT

temperatura\_umbral\_delta\_segundos INT

1. Modificar tabla 'ultimos\_estados' para que contenga la FK a 'tanques'
2. Crear tabla 'alarmas\_tanques'

id BIGINT (PK AI)

fecha\_hora DATETIME3333

tanque VARCHAR(45)

tipo VARCHAR(20) Tipos posibles: 'nivel' o 'temperatura' (controlar y rechazar si los valores no son correctos)

delta DOUBLE

dif\_secs BIGINT

informado` TINYINT(4) NOT NULL DEFAULT 0

1. Crear un evento que se ejecute cada 10 segundos y que realice la siguiente tarea:
2. Verifique valores delta de temperatura y nivel por tanque en 'ultimos\_estados'
3. Si cualquier delta supera nivel\_umbral\_delta o temperatura\_umbral\_delta, según corresponda y además se supera la 'dif\_secs' en función de 'nivel\_umbral\_delta\_segundos'
4. 'temperatura\_umbral\_delta\_segundos' se debe insertar una fila en la tabla 'alarmas\_tanques' con los datos correspondientes. Para realizar este cálculo debe crear una función
5. Crear un procedimiento almacenado que permita ingresar el nivel y la temperatura de un tanque (en la tabla 'ultimos\_estados', sólo debe recibir como parámetros tanque, nivel y temperatura) y siempre establecer el último estado. Además anotará en una nueva tabla de valores históricos 'valores\_historicos': tanque, fecha\_hora, nivel y temperatura, pero solo a intervalos mínimos, medidos en segundos, establecidos en un nuevo campo por tanque denominado 'intervalo\_minimo\_de\_almacenamiento', por ejemplo:

Intervalo=30 segundos ('intervalo\_minimo\_de\_almacenamiento'=30)

Valores recibidos:

- Tanque, Fecha Hora, Nivel, Temperatura, Histórico

- 'TK 1', '2021-1-1 00:00:00', 400, 23 Si

- 'TK 1', '2021-1-1 00:00:11', 401, 23 No

- 'TK 1', '2021-1-1 00:00:21', 402, 24 No

- 'TK 1', '2021-1-1 00:00:32', 403, 25 Si

- 'TK 1', '2021-1-1 00:00:45', 404, 24 No

- 'TK 1', '2021-1-1 00:00:56', 405, 23 No

- 'TK 1', '2021-1-1 00:01:06', 406, 22 Si

- 'TK 1', '2021-1-1 00:01:16', 407, 21 No

1. Crear evento que llame al procedimiento almacenado anteriormente a intervalos de 7 segundos enviando datos aleatorios de nivel y temperatura (en un rango dado TIP: crear una función!!!!!) para un tanque en particular

**Resultados:**

10

**1)\_**

CREATE TABLE tanques (nombre\_tanque VARCHAR(45), nivel\_umbral\_delta DOUBLE, temperatura\_umbral\_delta DOUBLE, nivel\_umbral\_delta\_segundos INT, temperatura\_umbral\_delta\_segundos INT);

**2)\_**

ALTER TABLE ultimos\_estados ADD CONSTRAINT fk\_tanque FOREIGN KEY (tanque) REFERENCES tanques(nombre\_tanque);

**3)\_**

CREATE TABLE alarmas\_tanques (id BIGINT AUTO\_INCREMENT PRIMARY KEY, fecha\_hora DATETIME, tanque VARCHAR(45), tipo VARCHAR(45), delta DOUBLE, dif\_secs BIGINT, informado TINYINT(4) NOT NULL DEFAULT 0);

**4)\_**

Función crearAlerta:

CREATE DEFINER=`root`@`%` FUNCTION `crearAlerta`(nombre\_tanque VARCHAR(45)) RETURNS int

BEGIN

DECLARE nivel VARCHAR(45);

DECLARE temperatura VARCHAR(45);

SET nivel = (

select tanques.nombre\_tanque

from tanques JOIN ultimos\_estados ON tanques.nombre\_tanque = ultimos\_estados.tanque

WHERE ultimos\_estados.nivel\_delta > tanques.nivel\_umbral\_delta AND ultimos\_estados.dif\_secs > tanques.nivel\_umbral\_delta\_segundos AND tanques.nombre\_tanque = nombre\_tanque

);

SET temperatura = (

select tanques.nombre\_tanque

from tanques JOIN ultimos\_estados ON tanques.nombre\_tanque = ultimos\_estados.tanque

WHERE ultimos\_estados.temperatura\_delta > tanques.temperatura\_umbral\_delta AND ultimos\_estados.dif\_secs > tanques.temperatura\_umbral\_delta\_segundos AND tanques.nombre\_tanque = nombre\_tanque

);

CASE

WHEN nivel = nombre\_tanque

THEN

INSERT INTO alarmas\_tanques(fecha\_hora, tanque, tipo, delta, dif\_secs, informado) VALUES(NOW(),nivel,"nivel",1,0.5,true);

ELSE return 0;

END CASE;

CASE

WHEN temperatura = nombre\_tanque

THEN

INSERT INTO alarmas\_tanques(fecha\_hora, tanque, tipo, delta, dif\_secs, informado) VALUES(NOW(),temperatura,"temperatura",1,0.5,true);

ELSE return 0;

END CASE;

RETURN 1;

END

show processlist;

set global event\_scheduler=on;

DELIMITER $$

CREATE EVENT IF NOT EXISTS crearAlarma

ON SCHEDULE EVERY 10 SECOND

DO

BEGIN

SELECT crearAlerta(1);

END$$

CALL practico2.crearAlarma();

**5)\_**

(este trigger se corre antes que se actualice la tabla ultimos\_estados)

DELIMITER $$

[DROP TRIGGER](https://dev.mysql.com/doc/refman/8.0/en/drop-trigger.html) [IF](https://dev.mysql.com/doc/refman/8.0/en/if-statement.html) [EXISTS](https://dev.mysql.com/doc/refman/8.0/en/exists-and-not-exists-subqueries.html) `valores\_hisotricos`$$

[CREATE TRIGGER](https://dev.mysql.com/doc/refman/8.0/en/create-trigger.html) `valores\_hisotricos` BEFORE [UPDATE](https://dev.mysql.com/doc/refman/8.0/en/update.html) ON [ultimos\_estados](http://localhost:8090/?server=mysql&username=root&db=practico1&table=ultimos_estados)

FOR EACH ROW

[BEGIN](https://dev.mysql.com/doc/refman/8.0/en/commit.html)

IF [TIMESTAMPDIFF](https://dev.mysql.com/doc/refman/8.0/en/date-and-time-functions.html#function_timestampdiff)(SECOND,OLD.fecha\_hora,[NOW](https://dev.mysql.com/doc/refman/8.0/en/date-and-time-functions.html#function_now)()) > 30

THEN

INSERT INTO valores\_historicos (tanque,fecha\_hora,nivel,temperatura)

VALUES('1', '2021-1-1 00:00:00', '20','32');

END IF;

[END](https://dev.mysql.com/doc/refman/8.0/en/begin-end.html)$$

DELIMITER ;

store procedure para insertar en ultimos\_estados

DELIMITER //

DROP PROCEDURE IF EXISTS ultimos\_estados;

CREATE PROCEDURE ultimos\_estados(tanque,fecha\_hora,nivel,temperatura)

BEGIN

INSERT INTO ultimos\_estados(tanque,fecha\_hora,nivel,temperatura)

VALUES(‘1’,’2021-1-1 00:01:16’,’600’,’20’)

END;

DELIMITER ;

CREATE DEFINER=`root`@`%` PROCEDURE `agregarValoresHistoricos`(tanque VARCHAR(45), nivel DOUBLE, temperatura DOUBLE)

BEGIN

DECLARE intervalo INT;

IF EXISTS (SELECT ultimos\_estados.tanque FROM ultimos\_estados WHERE ultimos\_estados.tanque = tanque )

THEN

UPDATE ultimos\_estados set fecha\_hora = now(), nivel = nivel, temperatura = temperatura;

ELSE

INSERT INTO ultimos\_estados(fecha\_hora, tanque,nivel, temperatura, nivel\_delta, temperatura\_delta, dif\_secs) values (NOW(), tanque, nivel, temperatura, nivel\_delta, temperatura\_delta, dif\_secs);

END IF;

set intervalo = (

SELECT tanques.intervalo\_minimo\_de\_almacenamiento

FROM tanques

WHERE tanques.nombre\_tanque = tanque

);

IF SECOND(now()) = intervalo

THEN

INSERT INTO ultimos\_estados(tanque, fecha\_hora, nivel, temperatura) VALUES (tanque, NOW(), nivel, temperatura);

END IF;

END

CREATE DEFINER=`root`@`%` FUNCTION `datos\_aleatorios`() RETURNS varchar(45) CHARSET utf8mb4

BEGIN

DECLARE nivel DOUBLE;

DECLARE temperatura DOUBLE;

set nivel = (

SELECT ROUND(((20 - 1) \* RAND() + 1), 2)

);

set temperatura = (

SELECT ROUND(((20 - 1) \* RAND() + 1), 2)

);

RETURN concat(nivel,' ',temperatura) ;

END

**6)\_**

Proto ejer6

DELIMITER $$

[DROP TRIGGER](https://dev.mysql.com/doc/refman/8.0/en/drop-trigger.html) [IF](https://dev.mysql.com/doc/refman/8.0/en/if-statement.html) [EXISTS](https://dev.mysql.com/doc/refman/8.0/en/exists-and-not-exists-subqueries.html) `agregarValoresHistoricos`$$

[CREATE TRIGGER](https://dev.mysql.com/doc/refman/8.0/en/create-trigger.html) `agregarValoresHistoricos` BEFORE [UPDATE](https://dev.mysql.com/doc/refman/8.0/en/update.html) ON [ultimos\_estados](http://localhost:8090/?server=mysql&username=root&db=practico1&table=ultimos_estados)

FOR EACH ROW

[BEGIN](https://dev.mysql.com/doc/refman/8.0/en/commit.html)

IF [TIMESTAMPDIFF](https://dev.mysql.com/doc/refman/8.0/en/date-and-time-functions.html#function_timestampdiff)(SECOND,OLD.fecha\_hora,[NOW](https://dev.mysql.com/doc/refman/8.0/en/date-and-time-functions.html#function_now)()) > 30

THEN

INSERT INTO agregarValoresHistoricos(tanque, nivel, temperatura)

VALUES('1', datos\_aleatorios());

END IF;

[END](https://dev.mysql.com/doc/refman/8.0/en/begin-end.html)$$

DELIMITER ;

DELIMITER $$

CREATE EVENT IF NOT EXISTS agregarValoresHistoricos

ON SCHEDULE EVERY 7 SECOND

DO

BEGIN

SELECT agregarValoresHistoricos(1, ,datos\_aleatorios());

END$$

CALL practico2.agregarValoresHistoricos();