

Verónica González Amor.

Entrega Actividades UF2177

Resuelve las siguientes consultas en la base de datos que se crea con este script.

```
DROP SCHEMA IF EXISTS supermarket;
CREATE SCHEMA supermarket;
USE supermarket;
CREATE TABLE drink (
    id INT AUTO_INCREMENT,
    name VARCHAR (255),
    expiration_date DATE,
    is_available BOOL,
    PRIMARY KEY (id)
);
INSERT INTO drink (name, expiration_date, is_available) VALUES
('coffee', '2025-06-20', TRUE),
('water', '2040-12-31', TRUE ),
('beer', '2025-03-30', FALSE);
SELECT * FROM drink;
CREATE TABLE request (
    id INT AUTO_INCREMENT,
    vendor VARCHAR(255),
    drink_id INT,
    quantity INT,
    PRIMARY KEY (id),
    FOREIGN KEY (drink_id) REFERENCES drink(id)
);
INSERT INTO request (vendor, drink_id, quantity) VALUES
('Ana', 1, 100),
('Ana', 2, 100),
('Carlos', 3, 150),
('Ana', 1, 50),
('Carlos', 3, 75),
('David', 2, 120);
,
```

Pregunta 1: Crea un Procedimiento para mostrar si una bebida está caducada o no según su ID usando **IF**.

```
/* Creamos el PROCEDIMIENTO*/
DELIMITER $$
CREATE PROCEDURE verificar_caducidad(IN bebida_id INT)
BEGIN
/*Se declara una variable para almacenar la fecha de caducidad
* obtenida de la tabla drink*/
    DECLARE fecha_caducidad DATETIME;
/*Obtener la fecha de caducidad segun el ID*/
    SELECT expiration_date INTO fecha_caducidad
    FROM drink
    WHERE id = bebida_id;
/*Comparamos la fecha de caducidad con la fecha actual*/
    IF fecha_caducidad < CURRENT_TIMESTAMP THEN
        SELECT CONCAT('La bebida con ID ', bebida_id, 'está caducada.');
    ELSE
        SELECT CONCAT('La bebida con ID ', bebida_id, 'no está caducada.');
    END IF;
END$$
DELIMITER ;
/*llamar al Procedimiento*/
CALL verificar_caducidad(1);
CALL verificar_caducidad(2);
CALL verificar_caducidad(3);
,
```

Pregunta 2: Usa WHILE para mostrar los números del 6 al 10.

```
/*Usa WHILE para mostrar los números del 6 al 10.*/
```

```
DELIMITER $$
CREATE PROCEDURE mostrar_numeros()
BEGIN
    -- Declarar variables
    DECLARE numero INT DEFAULT 6;
    DECLARE resultado VARCHAR(255) DEFAULT '';
    -- Bucle WHILE
    WHILE numero <= 10 DO
        SET resultado = CONCAT(resultado, numero, ', ');
        SET numero = numero + 1;
    END WHILE;
    -- Mostrar el resultado final
    SELECT resultado AS 'Números del 6 al 10';
END$$
DELIMITER ;
CALL mostrar_numeros();`
```

Pregunta 3: Insertar 5 bebidas de prueba usando WHILE.

```
DELIMITER $$
CREATE PROCEDURE insertar_bebidas_prueba()
BEGIN
    /*Declaramos una variable para controlar el bucle*/
    DECLARE contador INT DEFAULT 1;
    /*WHILE*/
    WHILE contador <= 5 DO
        INSERT INTO drink(name, expiration_date, is_available)
        VALUES (CONCAT('Bebida_prueba_', contador),
        DATE_ADD(CURRENT_DATE, INTERVAL contador YEAR), TRUE);
    /*DATE_ADD=función que modifica una fecha añadiendo o restando un
    * intervalo de tiempo
    * CURRENT_DATE= fecha actual del sistema
    * INTERVAL contador YEAR= indica que queremos sumar un número de años
    * igual al valor de la variable contador*/
    SET contador = contador + 1;
    END WHILE;
END$$
DELIMITER ;
CALL insertar_bebidas_prueba();
SELECT * FROM drink;
`
```

Pregunta 4: Insertar 3 filas en la tabla request con los siguientes datos dinámicos usando LOOP.

vendedor: "Loop_1", "Loop_2", "Loop_3"

drink_id: siempre 1

quantity: 10, 20, 30 (progresivo con i * 10)

```
/*Pregunta 4: Insertar 3 filas en la tabla request con los siguientes datos dinámicos usando LOOP.
```

```
vendedor: "Loop_1", "Loop_2", "Loop_3"
```

```
drink_id: siempre 1
```

```
quantity: 10, 20, 30 (progresivo con i * 10)*/
```

```
DELIMITER $$
CREATE PROCEDURE insertar_request_loop()
BEGIN
    /*Declaramos la variable*/
    DECLARE i INT DEFAULT 1;
```

```
/*LOOP*/
loop_bucle: LOOP
/*Insertamos filas en la tabla request DINÁMICA*/
INSERT INTO request (vendor, drink_id, quantity)
VALUES (CONCAT('Loop_', i), 1, i * 10);
/*Incrementa el índice*/
SET i = i + 1;
/*Condición de salida: cuando i > 3, salir del bucle*/
IF i > 3 THEN
    LEAVE loop_bucle;
END IF;
END LOOP;

END$$
DELIMITER ;
CALL insertar_request_loop();
SELECT * FROM request;`
```