



# Práctica de procedimientos y funciones



Pablo González Troyano

Gestión de Bases de Datos

1º ASIR

01//2021

# Índice

<b>Enunciado</b>	<b>3</b>
<b>Procedimientos</b>	<b>3</b>
Procedimiento 1	3
Procedimiento 2	4
Procedimiento 3	5
Procedimiento 4	5
<b>Funciones</b>	<b>7</b>
Función 1	7
Función 2	7

# Enunciado

Sobre la base de datos que ya tienes, crear, como mínimo:

- Dos procedimientos que contengan al menos una instrucción condicional (if) y/o un bucle (while):
  - Un procedimiento con parámetro/s de entrada
  - Un procedimiento con parámetro/s de entrada y salida.
- Dos funciones que contengan al menos una instrucción condicional (if) y/o un bucle (while).

Se valorará positivamente que algún procedimiento utilice alguna de las funciones creadas, o alguna función utilice alguno de los procedimientos creados.

## Procedimientos

### Procedimiento 1

*Versión mejorada del procedimiento ya entregado anteriormente.*

Se nos ha indicado que en el evento del pasado 2020-12-25 se ha confirmado un caso positivo en los asistentes al evento. El cliente positivo tiene el DNI 81678063A. Obtener su asiento y sus datos de contacto.

Localizar los asistentes cercanos (+-3 puestos) y obtener también sus datos.

```
DELIMITER $$
DROP PROCEDURE IF EXISTS CONTACTOS_COVID $$
CREATE PROCEDURE CONTACTOS_COVID(IN VAR_contacto VARCHAR(10),
                                IN VAR_fecha DATE,
                                OUT VAR_butaca_positivo INTEGER(2))
BEGIN
    DECLARE VAR_id_evento INT DEFAULT 0;
    DECLARE VAR_butaca_confinadx INT DEFAULT 0;
    SELECT ID_EVENTO
        FROM EVENTOS
        WHERE FECHA = VAR_fecha
        INTO VAR_id_evento;
    SELECT BUTACA
        FROM ASISTENCIA
        WHERE CLIENTE = VAR_contacto
        AND EVENTO = VAR_id_evento
        INTO VAR_butaca_confinadx;
    SELECT A.CLIENTE, C.NOMBRE, C.APELLIDOS, C.EMAIL
        FROM CLIENTES AS C, ASISTENCIA AS A
        WHERE A.EVENTO = VAR_id_evento
        AND BUTACA IN ((VAR_butaca_confinadx - 3),
```

```

        (VAR_butaca_confinadx - 2),
        (VAR_butaca_confinadx - 1),
        VAR_butaca_confinadx,
        (VAR_butaca_confinadx + 1),
        (VAR_butaca_confinadx + 2),
        (VAR_butaca_confinadx + 3))
    AND C.DNI_CLI = A.CLIENTE;
END$$
DELIMITER ;

```

```
CALL CONTACTOS_COVID('55820220W','2020-12-25',@butaca_conf);
```

CLIENTE	NOMBRE	APELLIDOS	EMAIL
05883847X	Felix	Medina Vargas	medinavargasfelix@correo.com
15676363T	Jose Carlos	Saez Castro	saezcastrojosecarlos@correo.com
26536647Y	Juan Carlos	Velasco Prieto	velascoprietojuancarlos@correo.com
55820220W	Ignacio	Molina Rodriguez	molinarodrigueznacion@correo.com
77557848P	Encarnacion	Rodriguez Cano	rodriguezcanoencarnacion@correo.com
86695737C	Francisco Jose	Saez Carmona	saezcarmonafranciscojose@correo.com
89716389J	Joan	Navarro Martinez	navarromartinezjoan@correo.com

```
7 rows in set (0.05 sec)
```

## Procedimiento 2

Crear un procedimiento que reciba el DNI de un cliente y muestre en mayúsculas las 3 primeras letras del nombre y las 4 últimas de sus apellidos. Es necesario de cara a los concursos, pues no se puede publicar el nombre completo.

```

DELIMITER $$
DROP PROCEDURE IF EXISTS CONCURSOS$$
CREATE PROCEDURE CONCURSOS (IN VAR_dni VARCHAR(10),
                            OUT VAR_nombreconcurso VARCHAR(8))
BEGIN
    SELECT CONCAT(LEFT(NOMBRE,3)," ",RIGHT(APELLIDOS,4))
    FROM CLIENTES
    WHERE DNI_CLI = VAR_dni INTO VAR_nombreconcurso;
END $$
DELIMITER ;

```

```
CALL CONCURSOS('57814348F',@ganador);
```

```
SELECT @ganador;
```

```
mysql> SELECT @ganador;
+-----+
| @ganador |
+-----+
| Mer Díez |
+-----+
1 row in set (0.00 sec)
```

## Procedimiento 3

Crear un procedimiento que reciba el DNI de un cliente y muestre en mayúsculas las 3 primeras letras del nombre y las 4 últimas de sus apellidos. Es necesario de cara a los concursos, pues no se puede publicar el nombre completo. Se deberá comprobar que el DNI introducido es correcto. Para hacerlo se utilizará la función DNI que se muestra más adelante. REVISAR ESTE

```
DELIMITER $$
DROP PROCEDURE IF EXISTS CONCURSOS2$$
CREATE PROCEDURE CONCURSOS2 (IN VAR_dni VARCHAR(10),
                             OUT VAR_nombreconcurso VARCHAR(8))
BEGIN
    IF VALIDADNI(VAR_dni) = 1 THEN
        SELECT CONCAT(LEFT(NOMBRE,3), " ", RIGHT(APELLIDOS,4))
        FROM CLIENTES
        WHERE DNI_CLI = VAR_dni INTO VAR_nombreconcurso;
    ELSE
        SELECT "Oye, colega... Introduce un DNI válido";
    END $$
DELIMITER ;
```

```
CALL CONCURSOS('57814348F',@ganador);
```

```
SELECT @ganador;
```

```
mysql> SELECT @ganador;
+-----+
| @ganador |
+-----+
| Mer Díez |
+-----+
1 row in set (0.00 sec)
```

## Procedimiento 4

```
DELIMITER $$
DROP PROCEDURE IF EXISTS ACTUALIZA_VENTAS;
CREATE PROCEDURE ACTUALIZA_VENTAS()
BEGIN
    DECLARE contador INT DEFAULT 0;
    DECLARE N_T INT DEFAULT 0;
    SELECT COUNT(*) FROM TIENDAS INTO N_T;
    WHILE (contador < N_T) DO
        SET contador = contador + 1;
        UPDATE TIENDAS
            SET VENTAS = (SELECT SUM(VENTAS)
                        FROM TRABAJADORES
                        WHERE OFICINA = contador)
            WHERE ID_OFI = contador;
    END WHILE;
END $$
```

```
SELECT * FROM TIENDAS;
```

ID_OFI	CIUDAD	REGION	VENTAS
1	Madrid	Centro	10308
2	Lugo	Oeste	4581
3	Madrid	Centro	3897
4	Avila	Centro	7256
5	Burgos	Norte	7978
6	Aviles	Norte	2952
7	Madrid	Centro	5707
8	Barcelona	Este	4717
9	Barcelona	Este	2237

```
9 rows in set (0.00 sec)
```

# Funciones

## Función 1

Se pide crear una función que valide los DNIs introducidos.

```
DELIMITER $$
DROP FUNCTION IF EXISTS VALIDADNI $$
CREATE FUNCTION VALIDADNI(DNI_Introducido VARCHAR(9))
RETURNS INT
DETERMINISTIC
BEGIN
    DECLARE LetrasCorrectas CHAR(23);
    DECLARE LetraIntroducida CHAR(1);
    DECLARE NumerosIntroducidos NUMERIC(8);
    DECLARE LetraCorrecta CHAR(1);
    DECLARE Resto INT;
    /*Aquí ponemos el orden de las letras según resto */
    SET LetrasCorrectas = 'TRWAGMYFPDXBNJZSQVHLCKE';
    SELECT RIGHT(DNI_Introducido,1) INTO LetraIntroducida;
    SELECT LEFT(DNI_Introducido,8) INTO NumerosIntroducidos;
    SELECT NumerosIntroducidos MOD 23 INTO Resto;
    SET LetraCorrecta = SUBSTRING(LetrasCorrectas,Resto+1,1);
    /*RETURN LetraCorrecta;*/
    IF LetraIntroducida = LetraCorrecta THEN
        RETURN 1;
    ELSE
        RETURN 0;
    END IF;
END $$
DELIMITER ;
SELECT VALIDADNI('57814348F');
```

```
mysql> SELECT VALIDADNI('57814348F');
+-----+
| VALIDADNI('57814348F') |
+-----+
| 1 |
+-----+
1 row in set (0.00 sec)
```

## Función 2

Si bien tenemos un procedimiento para averiguar los contactos cercanos, no sabemos en base a un DNI si ha estado en contacto. Nos dicen desde el Departamento de Atención al Cliente que están recibiendo muchas llamadas para preguntar. Desean saber en base al DNI del cliente que llama, la fecha y los casos confirmados (tienen nota de este último dato).

```

DELIMITER $$
DROP FUNCTION IF EXISTS ECONTACTO $$
CREATE FUNCTION ECONTACTO (VAR_contacto VARCHAR(10), /*Lo que tienen, positivo*/
                          VAR_fecha DATE, /*Fecha del evento*/
                          VAR_dnillama VARCHAR(10)) /*Id de la persona que llama,
para comparativa*/
RETURNS VARCHAR(25)
DETERMINISTIC
BEGIN
    DECLARE VAR_id_evento INT DEFAULT 0;
    DECLARE VAR_butaca_confinadx INT DEFAULT 0;
    DECLARE VAR_arraycontactos CHAR(63) DEFAULT NULL;
    IF VALIDADNI(VAR_dnillama) = 0 THEN
        RETURN "DNI NO VALIDO";
    END IF;
    SELECT ID_EVENTO
        FROM EVENTOS
        WHERE FECHA = VAR_fecha
        INTO VAR_id_evento;
    SELECT BUTACA
        FROM ASISTENCIA
        WHERE CLIENTE = VAR_contacto
        AND EVENTO = VAR_id_evento
        INTO VAR_butaca_confinadx;
    SELECT A.CLIENTE
        FROM ASISTENCIA AS A
        WHERE A.EVENTO = VAR_id_evento AND A.CLIENTE = VAR_dnillama
        AND BUTACA IN ((VAR_butaca_confinadx - 3),
                      (VAR_butaca_confinadx - 2),
                      (VAR_butaca_confinadx - 1),
                      (VAR_butaca_confinadx + 1),
                      (VAR_butaca_confinadx + 2),
                      (VAR_butaca_confinadx + 3))
        INTO VAR_arraycontactos;

    IF VAR_arraycontactos IS NOT NULL THEN
        RETURN "ES CONTACTO";
    ELSE
        RETURN "NO ES CONTACTO";
    END IF;
END$$

```

```

mysql> SELECT ECONTACTO('86695737C','2016-03-30','09707776M')$$
+-----+
| ECONTACTO('86695737C','2016-03-30','09707776M') |
+-----+
| ES CONTACTO |
+-----+
1 row in set (0.00 sec)

```