

Práctica 17B

1.- Corrige los errores y ejecuta.

```
# *****
# Procedimiento de prueba Practica 17B *
#   Uso de comentarios                *
# *****
CREATE PROCEDURE bucle(p1 INT)
BEGIN
-- Declaramos las Variables
DECLARE x INT Default 1;
# Inicio del Bucle
lazo: LOOP
    set p1 = p1 + 1;
    x = x+1;

    IF p1 < 10 THEN ITERATE lazo;
    END IF
    LEAVE lazo; /* Salida del bucle */
END LOOP lazo;
    Select x;
END
```

```
MariaDB [p17]> DELIMITER //
MariaDB [p17]> CREATE PROCEDURE bucle(p1 INT)
-> BEGIN
->     DECLARE x INT DEFAULT 1;
->     lazo: LOOP
->         SET p1 = p1 + 1;
->         SET x = x+1;
->         IF p1 < 10 THEN ITERATE lazo;
->         END IF;
->     LEAVE lazo;
->     END LOOP lazo;
->     SELECT x;
-> END//
Query OK, 0 rows affected (0.04 sec)

MariaDB [p17]> DELIMITER ;
```

Práctica 17B

2.- En el ejercicio anterior explica que sucede al pasar los valores 5, 8 y 20 (por separado ya que solo tiene un parámetro de entrada).

Repite el bucle hasta que p1 sea 10

```
MariaDB [p17]> CALL bucle(5);
+-----+
| x      |
+-----+
|      6 |
+-----+
1 row in set (0.00 sec)
```

```
MariaDB [p17]> CALL bucle(8);
+-----+
| x      |
+-----+
|      3 |
+-----+
1 row in set (0.00 sec)
```

El resultado es la suma de x (1), mas 1, esta suma se realiza antes de comparar si el numero es menor que 10

```
MariaDB [p17]> CALL bucle(20);
+-----+
| x      |
+-----+
|      2 |
+-----+
1 row in set (0.00 sec)

Query OK, 0 rows affected (0.00 sec)
```

3.- Di que tipos de comentarios hemos usado?

Comentario en línea --

Comentario en línea #

Práctica 17B

4.- Crea en la base de datos **Test** la tabla **primitiva** con un único campo **Num** entero y clave principal.

```
MariaDB [test]> USE test; CREATE TABLE primitiva (  
Database changed  
    ->     num INT(2) PRIMARY KEY  
    -> );  
Query OK, 0 rows affected (0.04 sec)
```

Luego **crea, comenta y comprueba** los procedimientos siguientes:

```
Create procedure primi()  
Begin  
Declare n int(2);  
Declare c int(1);  
Truncate primitiva;  
repeat  
    Set n= Floor(rand()*49)+1;  
    Call numero(n);  
    Select count(*) into c from primitiva;  
    Select c;  
    Until c=6  
End repeat;  
Select * from primitiva;  
End //
```

```
MariaDB [test]> DELIMITER //  
MariaDB [test]> CREATE PROCEDURE primi()  
    -> BEGIN  
    ->     -- Declaracion de variables  
    ->     DECLARE n INT(2);  
    ->     DECLARE c INT(1);  
    ->     -- Borra los datos de la tabla primitiva  
    ->     TRUNCATE primitiva;  
    ->     -- Muestra por pantalla valores aleatorios del 1 al 50 y se para si es 6  
    ->     REPEAT  
    ->         SET n = Floor(rand()*49)+1;  
    ->         CALL numero(n);  
    ->         SELECT COUNT(*) INTO c FROM primitiva;  
    ->         SELECT c;  
    ->         UNTIL c=6  
    ->     END REPEAT;  
    ->     SELECT * FROM primitiva;  
    -> END //  
Query OK, 0 rows affected (0.00 sec)
```

```
+-----+  
| num |  
+-----+  
| 3 |  
| 9 |  
| 13 |  
| 15 |  
| 21 |  
| 44 |  
+-----+  
6 rows in set (0.05 sec)  
Query OK, 0 rows affected (0.06 sec)
```

Práctica 17B

```
Create procedure numero(a int(2))
Begin
Declare x int(2);
Select num into x from primitiva where num = a;
If x is null then
    Insert into primitiva values(a);
End if;
End //
```

```
MariaDB [test]> DELIMITER //
MariaDB [test]> CREATE PROCEDURE numero(a INT(2))
-> BEGIN
->     -- Declara un entero x
->     DECLARE x INT(2);
->     -- Guarda el valor de a en x
->     SELECT num INTO x FROM primitiva WHERE num = a;
->     -- Si el valor de x no existe en la tabla lo inserta
->     IF x IS NULL THEN
->         INSERT INTO primitiva VALUES(a);
->     END IF;
-> END //
Query OK, 0 rows affected (0.00 sec)
```

```
MariaDB [test]> DELIMITER ;
```

```
MariaDB [test]> CALL numero(3);
Query OK, 1 row affected (0.01 sec)
```

```
MariaDB [test]> SELECT * FROM primitiva;
+-----+
| num |
+-----+
| 3 |
+-----+
1 row in set (0.00 sec)
```

Práctica 17B

5.- Explica que sucede y porqué si ejecutamos:

Call numero(35);

Call numero(85);

Call numero(125);

Call numero(35);

Select * from primitiva;

```
MariaDB [test]> Call numero(35);
Query OK, 1 row affected (0.00 sec)

MariaDB [test]> Call numero(85);
Query OK, 1 row affected (0.00 sec)

MariaDB [test]>
MariaDB [test]> Call numero(125);
Query OK, 1 row affected (0.01 sec)

MariaDB [test]>
MariaDB [test]> Call numero(35);
Query OK, 1 row affected (0.00 sec)

MariaDB [test]> Select * from primitiva;
+-----+
| num |
+-----+
| 3 |
| 9 |
| 13 |
| 15 |
| 21 |
| 35 |
| 44 |
| 85 |
| 125 |
+-----+
9 rows in set (0.00 sec)
```

Introduce los números en la tabla, si estos no existen

Práctica 17B

6.- Haz un Procedimiento que haga **lo mismo EXACTAMENTE** que el del ejercicio 1 pero usando un bucle **WHILE**.

```
MariaDB [test]> DELIMITER //
```

```
MariaDB [test]> CREATE PROCEDURE buclewhile(p1 INT)
```

```
    -> BEGIN
```

```
    ->     DECLARE x INT DEFAULT 1;
```

```
    ->     WHILE p1 < 10 DO
```

```
    ->         SET p1 = p1+1;
```

```
    ->         SET x = x+1;
```

```
    ->     END WHILE;
```

```
    ->     SELECT x;
```

```
    -> END//
```

```
Query OK, 0 rows affected (0.00 sec)
```

```
MariaDB [test]> DELIMITER ;
```