

# **BASE DE DATOS 2**

## **HITO 3**

### **PRESENTACION DE PREGUNTAS TEÓRICAS Y PRACTICAS**

**NOMBRE: JOEL REYNALDO**

# **OBJETIVOS A MOSTRAR**

- **MANEJO DE CONCEJOS**
- **PARTE PRACTICA**

# 1. Defina que es lenguaje procedural en MySQL

R.- Es el manejo de instrucciones o estructuras de control (programación) dentro de una base de datos

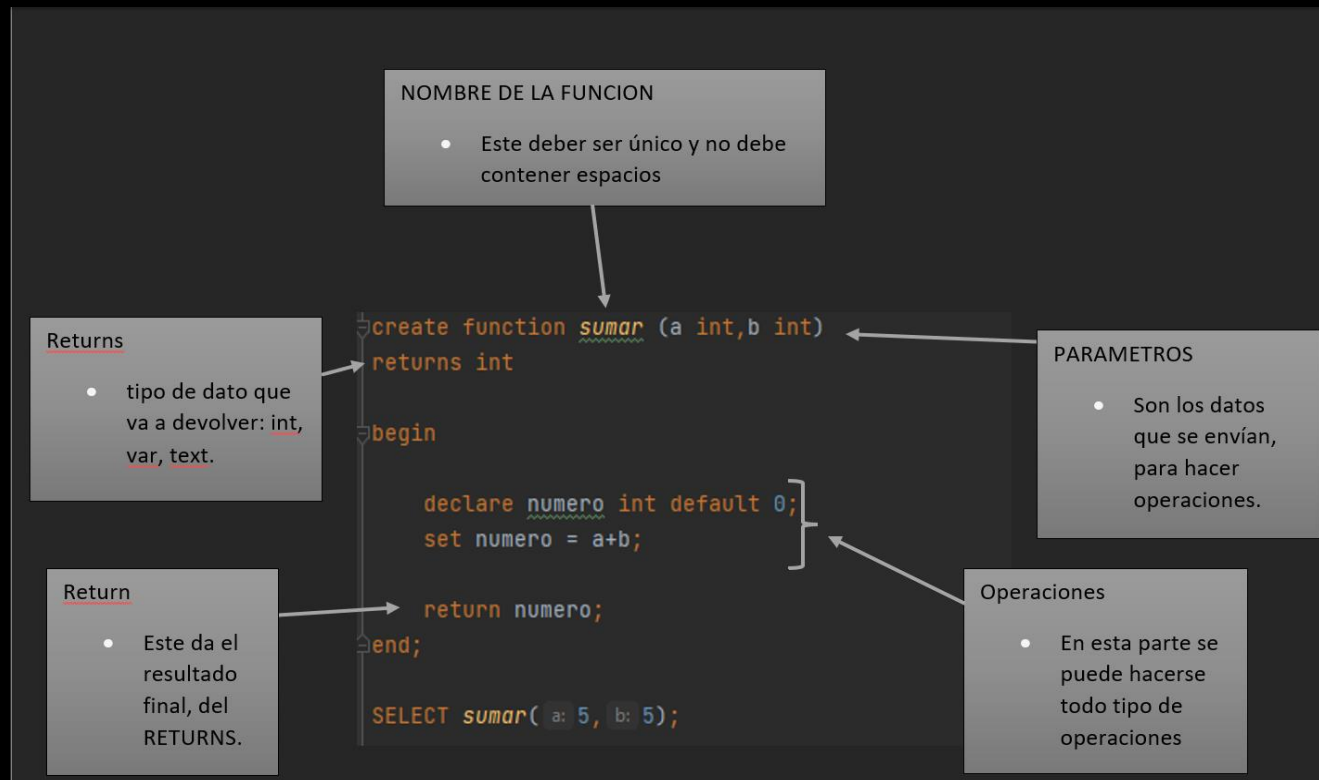
# 2. Defina que es una función en MySQL

R.- Es son funciones almacenadas que pueden modificar o realizar operaciones con los registros de la base datos, puedan o no pueden recibir o parámetros

# 3. ¿Qué cosas características debe de tener una función?

Explique sobre el: NOMBRE, EL RETURN, PARAMETROS,ETC

R.-



#### 4. ¿Cómo crear, modificar y como eliminar una función?

Para crear una función “CREATE FUNCTION”

R.-

```
create function sumar (a int,b int)
returns int

begin

    declare numero int default 0;
    set numero = a+b;

    return numero;
end;

SELECT sumar( a: 5, b: 5);
```

Para modificar una función se debe hacer uso de la sentencia “OR REPLACE”

```
create or replace function sumar (a int,b int)
returns int

begin

    declare numero int default 0;
    set numero = a+b;

    return numero;
end;

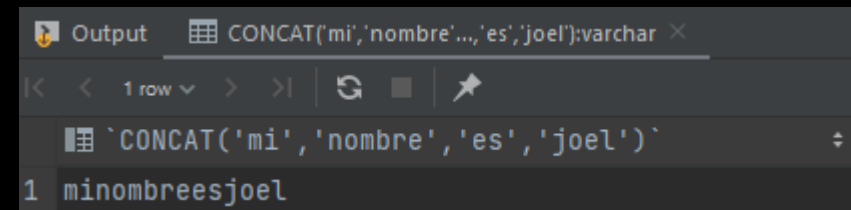
SELECT sumar( a: 5, b: 5);
```

5. ¿Para que sirve la función concat y como funciona en MySQL?

R.- ¿Crear una función que muestre el uso de las funciones CONCAT?

```
create or replace function CONCAT(A varchar(20),B varchar(20),C varchar (20))
returns text
begin
    declare nombres text default '';
    set nombres = concat(A, ' ', B, ' ', C);
    return nombres;
end;

select CONCAT('mi', 'nombre', 'es', 'joel');
```



The screenshot shows a database output window titled "Output" with a sub-header "CONCAT('mi','nombre'..., 'es','joel'):varchar". It displays a single row of data. The first column contains the SQL query: `CONCAT('mi','nombre','es','joel')`. The second column contains the result: "minombreesjoel".

1	minombreesjoel

6. Para que sirve la función substring y como funciona en MYSQL

R.- substring nos permite ver la posición de una cadena y su longitud= substrin(cadena,posición,longuitud)

¿crear una function que muestre el uso de las funciones SUBSTRING?

La función recibe un nombre completo

Input: Ximena Condori Mar

La función solo retorna el nombre

Output: Ximena

R.-

```
create or replace function pop(nombre varchar(50),posicion int,longuitud int)
returns text
begin
    declare cadena text default '';
    set cadena = substring(nombre,posicion,longuitud);
    return cadena;
end;

select pop( nombre: 'XIMENA CONDORI MAR', posicion: 1, longuitud: 7)
```

- 7. Para que sirve la función STRCMP y como funciona en MySQL
- Crear una función que muestre el uso de la función STRCMP
- La función debe comparar 3 cadenas. Y deberá determinar si dos de ellas son iguales

```
create or replace function igualar( a varchar(20),b varchar(20),c varchar (20))
returns text
begin
    declare respuesta text default '';
    if strcmp (a,b) = 0 then set respuesta = 'dos son de ellas son iguales';
    elseif strcmp (c,a) = 0 then set respuesta = 'dos son de ellas son iguales';
    elseif strcmp (c,b) = 0 then set respuesta = 'dos son de ellas son iguales';

    else
        set respuesta = 'tres de ellos son iguales';
    end if;
    return respuesta;
end;

select igualar( a: 'perro', b: 'perro', c: 'gato');
```

```
mysql> `igualar('perro','perro','gato')`
1 dos son de ellas son iguales
```

## 8. Para que sirve la función CHAR\_LENGTH Y LOCATE y Como funciona en MySQL

R.- **CHAR\_LENGTH:** nos permite contar la cantidad de letras y espacios de una cadena

**LOCATE:** este nos permite encontrar una palabra de una cadena

```
create or replace function lenght(a varchar (20))
returns text
begin
    declare respuesta text default '';
    set respuesta = char_length(a);
    return respuesta;
end;

select lenght(a: 'joel')
```

```
1 4
```

```
create or replace function loc(subcadena varchar (20), cadena varchar(20))
returns text
begin
    declare respuesta text default 0;
    set respuesta = locate(subcadena,cadena);
    return respuesta;
end;

select loc( subcadena: 'hola', cadena: 'mundo_hola');
```

```
1 7
```

## 9. Cual es la diferencia entre las funciones de agregación y funciones creados por el DBA

Es decir funciones creadas por el usuario

R.- **FUNCIONES DE AGRACION:** esta funcion no necesita ser creada

**FUNCIONES CREADAS POR EL USUARIO:** estas necesitan ser creadas,

Se crean haciendo uso de las funciones de agragacion.

10. ¿Busque y defina a que se referira cuando se habla de parametros de entrada y salida en MySQL?.

R.- PARAMETROS DE ENTRADA "IN": es un parámetro que entra dentro del procedimiento almacenado con el cual se harán operaciones.

PARAMETROS DE SALIDA "OUT": es un parámetro el cual devolverá un dato de salida al usuario.

PARAMETRO DE ENTRADA Y SALIDA "INOUT": es un parámetro que entrara dentro del procedimiento



## 11 Crear la siguiente base de datos y sus registros.

R.-

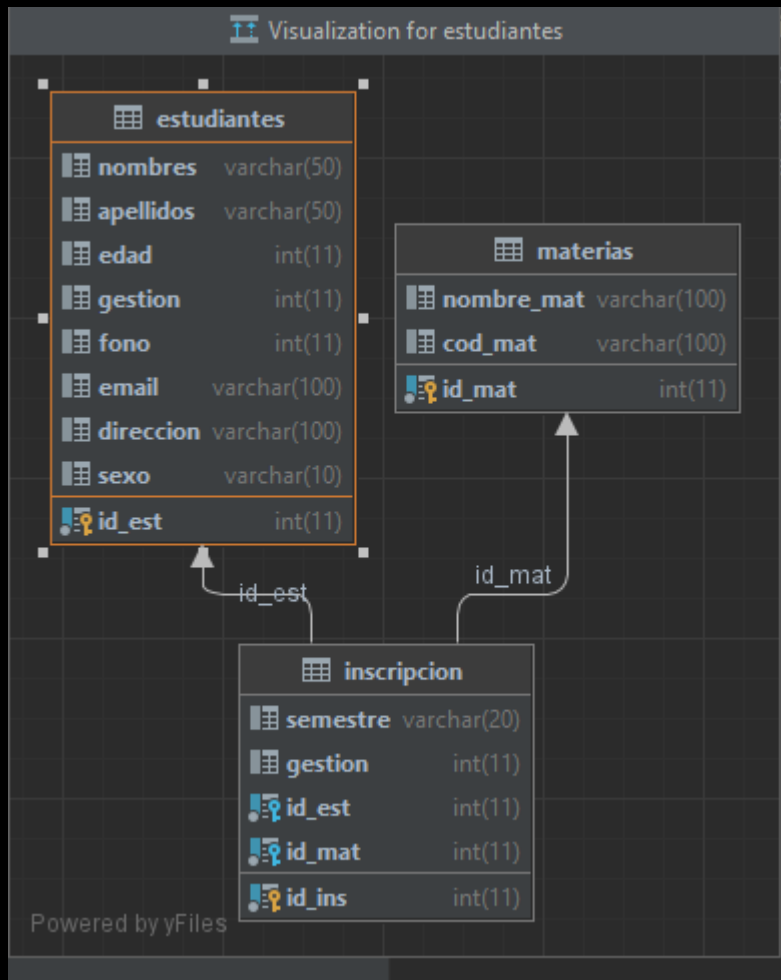
```
1  create database practica3;
2  use practica3;
3  create table estudiantes
4  (
5      id_est      int auto_increment primary key not null,
6      nombres     varchar(50),
7      apellidos   varchar(50),
8      edad        int(11),
9      gestion     int(11),
10     fono         int(11),
11     email        varchar(100),
12     direccion    varchar(100),
13     sexo         varchar(10)
14 );
15 create table materias
16 (
17     id_mat       int auto_increment primary key not null,
18     nombre_mat   varchar(100),
19     cod_mat      varchar(100)
20 );
21
22 create table inscripcion
23 (
24     id_ins       int auto_increment primary key not null,
25     semestre     varchar(20),
26     gestion      int(11),
27
28     id_est       int                not null,
29     id_mat       int                not null,
30
31     foreign key (id_est) references estudiantes (id_est),
32     foreign key (id_mat) references materias (id_mat)
33 );
```

```
35 INSERT INTO estudiantes (nombres, apellidos, edad, fono, email, direccion, sexo)
36 VALUES ('Miguel', 'Gonzales Veliz', 20, 2832115, 'miguel@gmail.com', 'Av. 6 de Agosto', 'masculino'),
37         ('Sandra', 'Mavir Uria', 25, 2832116, 'sandra@gmail.com', 'Av. 6 de Agosto', 'femenino'),
38         ('Joel', 'Aduviri Mondar', 30, 2832117, 'joel@gmail.com', 'Av. 6 de Agosto', 'masculino'),
39         ('Andrea', 'Arias Ballesteros', 21, 2832118, 'andrea@gmail.com', 'Av. 6 de Agosto', 'femenino'),
40         ('Santos', 'Montes Valenzuela', 24, 2832119, 'santos@gmail.com', 'Av. 6 de Agosto', 'masculino');
41
42 INSERT INTO materias (nombre_mat, cod_mat)
43 VALUES ('Introduccion a la Arquitectura', 'ARQ-101'),
44         ('Urbanismo y Diseno', 'ARQ-102'),
45         ('Dibujo y Pintura Arquitectonico', 'ARQ-103'),
46         ('Matematica discreta', 'ARQ-104'),
47         ('Fisica Basica', 'ARQ-105');
48
49 INSERT INTO inscripcion (semestre, gestion, id_est, id_mat)
50 values ('1er Semestre', 2018, 1, 1),
51        ('2do Semestre', 2018, 1, 2),
52        ('1er Semestre', 2019, 2, 4),
53        ('2do Semestre', 2019, 2, 3),
54        ('2do Semestre', 2020, 3, 3),
55        ('3er Semestre', 2020, 3, 1),
56        ('4to Semestre', 2021, 4, 4),
57        ('5to Semestre', 2021, 5, 5);
58
```

	id_est	nombres	apellidos	edad	gestion	fono	email	direccion	sexo
1	1	Miguel	Gonzales Veliz	20	<null>	2832115	miguel@gmail.com	Av. 6 de Agosto	masculino
2	2	Sandra	Mavir Uria	25	<null>	2832116	sandra@gmail.com	Av. 6 de Agosto	femenino
3	3	Joel	Adubiri Mondar	30	<null>	2832117	joel@gmail.com	Av. 6 de Agosto	masculino
4	4	Andrea	Arias Ballesteros	21	<null>	2832118	andrea@gmail.com	Av. 6 de Agosto	femenino
5	5	Santos	Montes Valenzuela	24	<null>	2832119	santos@gmail.com	Av. 6 de Agosto	masculino

WHERE			ORDER BY		
	id_ins	semestre	gestion	id_est	id_mat
1	1	1er Semestre	2018	1	1
2	2	2do Semestre	2018	1	2
3	3	1er Semestre	2019	2	4
4	4	2do Semestre	2019	2	3
5	5	2do Semestre	2020	3	3
6	6	3er Semestre	2020	3	1
7	7	4to Semestre	2021	4	4
8	8	5to Semestre	2021	5	5

	id_mat	nombre_mat	cod_mat
1	1	Introduccion a la Arquitectura	ARQ-101
2	2	Urbanismo y Diseno	ARQ-102
3	3	Dibujo y Pintura Arquitectonico	ARQ-103
4	4	Matematica discreta	ARQ-104
5	5	Fisica Basica	ARQ-105



## • 12 Crear una función que genere la serie Fibonacci.

- La función recibe un límite(number)
- La función debe de retornar una cadena.
- Ejemplo para **n=7. OUTPUT: 0, 1, 1, 2, 3, 5, 8,**
- Adjuntar el **código SQL generado y una imagen de su correcto funcionamiento.**

```
#12.Crear una función que genere la serie Fibonacci.
#◦ La función recibe un límite(number)
#◦ La función debe de retornar una cadena.
#◦ Ejemplo para n=7. OUTPUT: 0, 1, 1, 2, 3, 5, 8,
#◦ Adjuntar el código SQL generado y una imagen de su correcto funcionamiento.

CREATE OR REPLACE FUNCTION fibonacci(limite INT)
RETURNS TEXT
DETERMINISTIC
BEGIN
    DECLARE fib1 INT DEFAULT 0;
    DECLARE fib2 INT DEFAULT 1;
    DECLARE fib3 INT DEFAULT 0;
    DECLARE str VARCHAR(255) DEFAULT '0,1,';

    IF limite = 1 THEN
        RETURN fib1;
    ELSEIF limite = 2 THEN
        RETURN CONCAT(fib1, fib2);
    ELSE
        WHILE limite > 2 DO
            SET fib3 = fib1 + fib2;
            SET fib1 = fib2;
            SET fib2 = fib3;
            SET limite = limite - 1;
            SET str = CONCAT(str, fib3, ',');
        END WHILE;
        RETURN str;
    END IF;
END;

select fibonacci( limite: 7);
```

1 row	
	`fibonacci(7)`
1	0,1,1,2,3,5,8,

## 13. Crear una variable global a nivel BADE DE DATOS

```
set @limit=7;

CREATE OR REPLACE FUNCTION fibonacci1()
RETURNS text
DETERMINISTIC
BEGIN
    DECLARE fib1 INT DEFAULT 0;
    DECLARE fib2 INT DEFAULT 1;
    DECLARE fib3 INT DEFAULT 0;
    DECLARE str VARCHAR(255) DEFAULT '0,1,';

    IF @limit = 1 THEN
        RETURN fib1;
    ELSEIF @limit = 2 THEN
        RETURN CONCAT(fib1, fib2);
    ELSE
        WHILE @limit > 2 DO
            SET fib3 = fib1 + fib2;
            SET fib1 = fib2;
            SET fib2 = fib3;
            SET @limit = @limit - 1;
            SET str = CONCAT(str, fib3, ',');
        END WHILE;
        RETURN str;
    END IF;
END;

select fibonacci1();
```

1 row	
`fibonacci1()`	
1	0,1,1,2,3,5,8,

## 14. Crear una función no recibe parámetros (utilizar WHILE, REPEAT o LOOP)

```
create or replace function example2_1()
returns text
begin
    declare respuesta text default '';
    declare limite int;
    declare x int;

    select min(est.edad) into limite
    from estudiantes as est;

    if limite%2=0 then set x=2;
    while x<=limite do
        set respuesta= concat(respuesta,x,',');
        set x=x+2;
    end while;
    else
    set x=1;
    while x<=limite do
        set x=x+2;
        set respuesta= concat(respuesta,x,',');
    end while;
    end if;
    return respuesta;
end;

select example2_1();
```

📄 `example2\_1()`

1 2,4,6,8,10,12,14,16,18,20,

# 15. Crear una función que determina cuantas veces se repite las vocales.

```
create or replace function vowel_count(textoAIngresar varchar(50))
returns text
begin
    return concat(
        concat (' a: ', (LENGTH(textoAIngresar) - LENGTH(REPLACE(textoAIngresar, 'a', '')))/LENGTH('a')) ,
        concat (' e: ', (LENGTH(textoAIngresar) - LENGTH(REPLACE(textoAIngresar, 'e', '')))/LENGTH('e')) ,
        concat (' i: ', (LENGTH(textoAIngresar) - LENGTH(REPLACE(textoAIngresar, 'i', '')))/LENGTH('i')) ,
        concat (' o: ', (LENGTH(textoAIngresar) - LENGTH(REPLACE(textoAIngresar, 'o', '')))/LENGTH('o')) ,
        concat (' u: ', (LENGTH(textoAIngresar) - LENGTH(REPLACE(textoAIngresar, 'u', '')))/LENGTH('u'))
    );
end;

select vowel_count('bendito sea diabloooo');
```

```
`vowel_count('bendito sea diabloooo')`
1 a: 2.0000 e: 2.0000 i: 2.0000 o: 5.0000 u: 0.0000
```

## 16. Crear una función que recibe un parámetro INTEGER.

```
create or replace function creditoNumber(credit_number int)
RETURNS text
begin
    declare respuesta text default '';          # Y = AND      O = OR
    case
        when credit_number > 50000 then set respuesta = 'PLATINUM'; # Si es mayor a 50000 es PLATINIUM.
        when credit_number >= 10000 AND credit_number <= 50000 THEN SET respuesta = 'GOLD';      # Si es mayor igual a 10000 y menor igual a 50000 es GOLD.
        when credit_number < 10000 then set respuesta = 'SILVER';      # Si es menor a 10000 es SILVER
        else set respuesta = 'CASO DESCONOCIDO'; # CASO POR DEFECTO
    end case;
    return respuesta;
end;

SELECT creditoNumber( credit_number: 50000);
```

📄 `creditoNumber(50000)`

1 GOLD