

UNIVERSIDAD PRIVADA FRANZ TAMAYO



DEFENSA HITO 3 - TAREA FINAL

Estudiante: Univ. DAFNET LAYDA MAMANI LAURA

Asignatura: BASE DE DATOS II

Carrera: INGENIERÍA DE SISTEMAS

Sede: El Alto

Paralelo: BDA (1)

Docente: Lic. William Barra Paredes

fecha: 22/10/2022

GITHUB: <https://github.com/daf12/Base-de-Datos-II>

Parte teorica

1. DEFINA QUE ES LENGUAJE PROCEDURAL EN MYSQL.

El lenguaje procedural es programación a nivel de base de datos, para poder llegar a programar a nivel de base de datos tenemos que saber conceptos como, procedimientos integrados, funciones, etc., en MySQL o la base de datos que desee.

2. DEFINA QUE ES UNA FUNCIÓN EN MYSQL.

Una función en MySQL es un fragmento de código en lenguaje procedural que te retorna un valor.

3. ¿QUÉ COSAS CARACTERÍSTICAS DEBE DE TENER UNA FUNCIÓN? EXPLIQUE SOBRE EL NOMBRE, EL RETURN, PARAMETROS, ETC.

Al crear una función en MySQL se debe poner un nombre único para identificar la función, en el return decimos que valor vamos a devolver, los parámetros son datos que necesitamos para la función, pueden o no ser importantes dependiendo, en el return devolvemos un valor.

4. ¿CÓMO CREAR, MODIFICAR Y CÓMO ELIMINAR UNA FUNCIÓN? ADJUNTE UN EJEMPLO DE SU USO.

Para crear una función tenemos que poner "CREATE FUNCTION" {nombre de la función}, un return, BEGIN y END.

Para modificar una función tenemos que poner al lado de "CREATE OR REPLACE FUNCTION".

Para eliminar es "DROP FUNCTION"

5. PARA QUÉ SIRVE LA FUNCIÓN CONCAT Y COMO FUNCIONA EN MYSQL

- ¿CREAR UNA FUNCIÓN QUE MUESTRE EL USO DE LA FUNCIÓN CONCAT?
- LA FUNCIÓN DEBE CONCATENAR 3 CADENAS.

La función "Concat" te puede concatenar varias variables y demás.

Select CONCAT ('hola', ' dafnet') as Concat

6. PARA QUÉ SIRVE LA FUNCIÓN SUBSTRING Y COMO FUNCIONA EN MYSQL

- ¿CREAR UNA FUNCIÓN QUE MUESTRE EL USO DE LA FUNCIÓN SUBSTRING?

- LA FUNCIÓN RECIBE UN NOMBRE COMPLETO.

■ INPUT: XIMENA CONDORI MAR

- LA FUNCIÓN SOLO RETORNA EL NOMBRE.

■ OUTPUT: XIMENA

La función "SUBSTRING" sirve para contar una cadena de cierto lugar con un conteo determinado.

Select substr ('ximena condori mar', 2, 2) as Substr

7. PARA QUÉ 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.

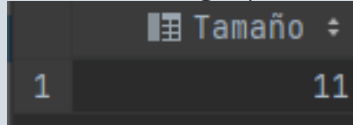
Compara cadenas y te retorna un número en binario.

Select strcmp('Hola', 'Hola') as Strcmp

8. PARA QUÉ SIRVE LA FUNCIÓN CHAR_LENGTH Y LOCATE Y COMO FUNCIONA EN MYSQL

○ ¿CREAR UNA FUNCIÓN QUE MUESTRE EL USO DE AMBAS FUNCIONES?

select char_length('Hola dafnet') as Tamaño



| | Tamaño |
|---|--------|
| 1 | 11 |

Select locate ('la', 'dafnet') as Locate

9. ¿CUAL ES LA DIFERENCIA ENTRE LAS FUNCIONES DE AGRESIÓN Y FUNCIONES CREADOS POR EL DBA? ES DECIR FUNCIONES CREADAS POR EL USUARIO.

Las funciones de agregación son las que están definidas en la base de datos y no se pueden utilizar en la clausula select.

Las funciones definidas por el usuario son las que las crea el DBA y se pueden utilizar en la clausula select.

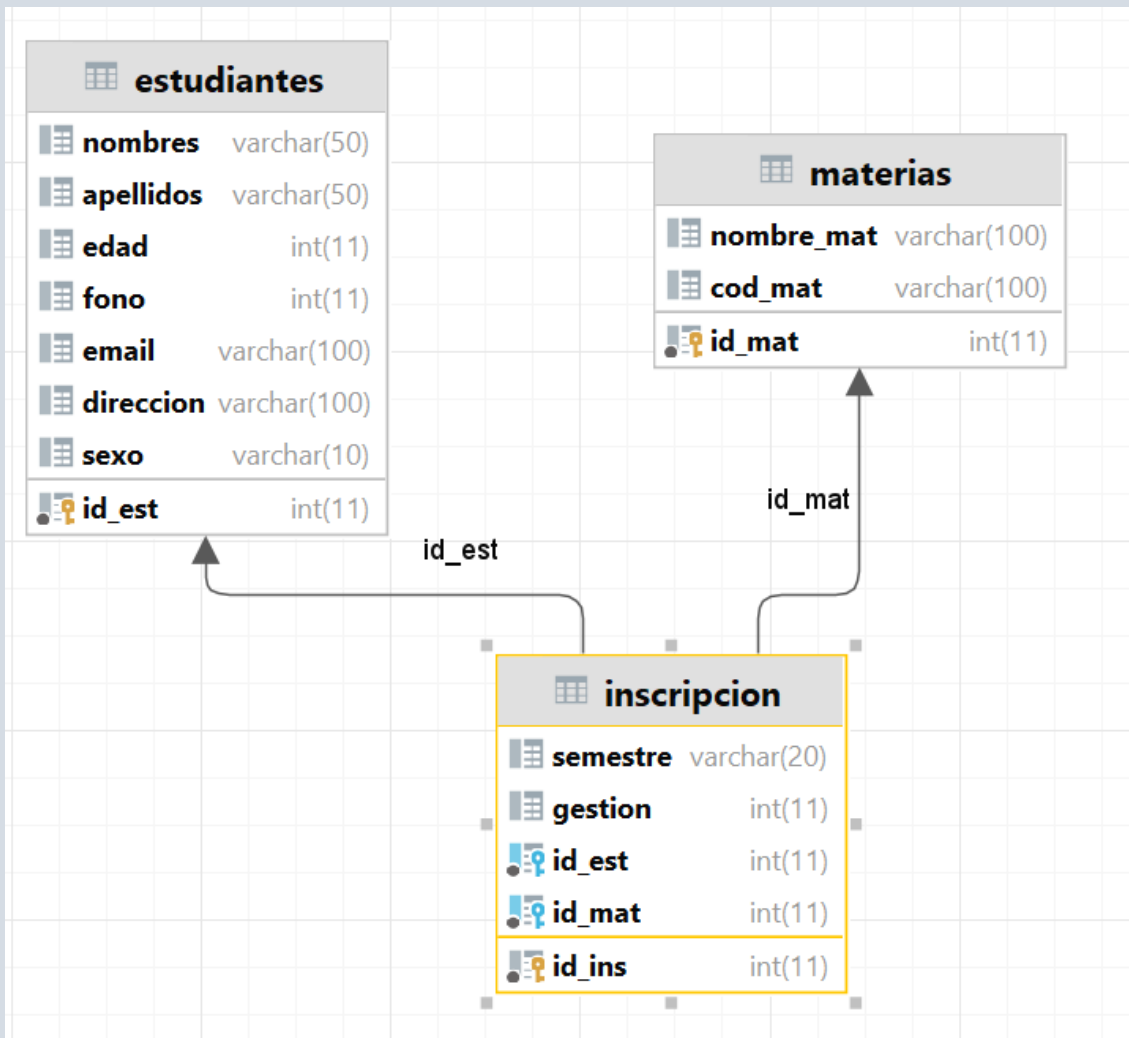
10.¿BUSQUE Y DEFINA A QUÉ SE REFERIRÁ CUANDO SE HABLA DE PARÁMETROS DE ENTRADA Y SALIDA EN MYSQL?

Los parámetros de entrada son las que se definen al inicio de la función, son variables que la función necesita.

Los parámetros de salida son aquellas variables que se utilizan en la función y se retornan con el "return

Parte practica

11. Crear la siguiente Base de datos y sus registros



```

create table estudiantes (
    nombres varchar (30),
    apellidos varchar (50),
    edad int,
    fono int,
    email varchar (100),
    direccion varchar (100),
    sexo varchar (10),
    id_est int not null auto_increment primary key
);
  
```

```

create table inscripcion (
    semestre varchar (20),
    gestion int,
    id_est int not null,
    id_mat int not null,
    foreign key (id_est) references estudiantes(id_est),
    foreign key (id_mat) references materias(id_mat)
);
  
```

```
create table materias (
  nombre_mat varchar (100),
  cod_mat varchar (100),
  id_mat int not null auto_increment primary key
);
```

```
insert into estudiantes (nombres, apellidos, edad, fono, email, direccion, sexo)
values ('Miguel', 'Gonzales Veliz', 20, 2832115, 'miguel@gmail.com', 'Av. 6 de Agosto', 'masculino'),
('Sandra', 'Mavir Uria', 25, 2832116, 'sandra@gmail.com', 'Av. 6 de Agosto', 'femenino'),
('Joel', 'Aduviri Mondar', 30, 2832117, 'joel@gmail.com', 'Av. 6 de Agosto', 'masculino'),
('Andrea', 'Arias Ballesteros', 21, 2832118, 'andrea@gmail.com', 'Av. 6 de Agosto', 'femenino'),
('Santos', 'Montes Valenzuela', 24, 2832119, 'santos@gmail.com', 'Av. 6 de Agosto', 'masculino');
```

```
insert into inscripcion (semestre, gestion, id_est, id_mat)
values ('1er Semestre', 2018, 1, 1),
('2do Semestre', 2018, 1, 2),
('1er Semestre', 2019, 2, 4),
('2do Semestre', 2019, 2, 3),
('2do Semestre', 2020, 3, 3),
('3er Semestre', 2020, 3, 1),
('4to Semestre', 2021, 4, 4),
('5to Semestre', 2021, 5, 5);
```

```
insert into materias (nombre_mat, cod_mat)
values ('Introduccion a la Arquitectura', 'ARQ-101'),
('Urbanismo y Diseño', 'ARQ-102'),
('Dibujo y Pintura Arquitectonico', 'ARQ-103'),
('Matematica discreta', 'ARQ-104'),
('Fisica Basica', 'ARQ-105');
```

12.Crear una función que genere la serie Fibonacci.

```

create function fibonacci(lim int)
returns text
begin
    declare dato int default 0;
    declare dato1 int default 1;
    declare dato2 int default 0;
    declare x int default 1;
    declare respuesta text;
    if lim >= 1
    then
        set respuesta = concat(dato, ',');
    end if;
    if lim >= 2
    then
        set respuesta = concat(respuesta, dato1, ',');
    end if;
    if lim >= 3
    then
        while x <= (lim - 2) do
            set dato2 = dato + dato1;
            set respuesta = concat(respuesta, dato2, ',');
            set dato = dato1;
            set dato1 = dato2;
            set x = x + 1;
        end while;
    end if;
    return respuesta;
end;

select fibonacci(lim: 5)

```

| | fibonacci |
|---|------------|
| 1 | 0,1,1,2,3, |

13. Crear una variable global a nivel BASE DE DATOS.

```

create function variableglobal ()
returns int
begin
    set @variableglobal = 1;
    return @variableglobal;
end;

select variableglobal();

```

```

set @variableglobal = 0;

```

| | variable | |
|---|----------|---|
| 1 | | 1 |

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

```

create function EdadMinima ()
returns int
begin
    declare est int default 0;
    set est = (select min(est.edad)
               from estudiantes as est);
    return est;
end;

select EdadMinima();

create function function1()
returns text
begin
    declare edad int default 0;
    declare cont int default 0;
    declare res text default '';

    set edad = EdadMinima();
    if (edad % 2 = 0)
    then

```

```

while (cont <= edad) do
    # PAR
    if(cont % 2 = 0)
        then
            set res = concat(res, cont, ', ');
        end if;
        set cont = cont + 1;
    end while;
else
    set cont = edad;
    while (cont >= 0) do
        if(cont % 2 = 1)
            then
                set res = concat(res, cont, ', ');
            end if;
            set cont = cont - 1;
        end while;
    end if;
    return res;
end;

```

```

function1()
1 0, 2, 4, 6, 8, 10, 12, 14, 16, 18, 20,

```

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

```

returns text
begin
    declare x int default 1;
    declare res text default '';
    declare lim int default char_length(cadena);
    declare letra char default '';
    declare a int default 0;
    declare e int default 0;
    declare i int default 0;
    declare o int default 0;
    declare u int default 0;
    while x <= lim
        do
            set letra = substr(cadena, x, 1);
            if letra = 'a'
                then
                    set
                        a = a + 1;
            else if letra = 'e'
                then
                    set
                        e = e + 1;

```



```

        else if letra = 'i'
            then
                set
                    i = i + 1;
        else if letra = 'o'
            then
                set
                    o = o + 1;
        else if letra = 'u'
            then
                set
                    u = u + 1;
        end if;
    end if;
end if;
end if;
end if;
set x= x + 1;
end while;

set res = concat( 'a:', a, ' ', 'e:', e, ' ', 'i:', i, ' ', 'o:', o, ' ', 'u:', u);
return (res);
end;

```

```
select ContarVocales( cadena: 'taller de base de datos') as Vocales;
```

| Vocales | |
|---------|----------------------|
| 1 | a:3 e:4 i:0 o:1 u: 0 |

16. Crear una función que recibe un parámetro INTEGER.

```

create or replace function usuario(credit_number INT)
returns text
begin
    declare respuesta text default '';
    set respuesta = (
        select case
            when credit_number > 50000 then 'PLATINIUM.'
            when credit_number >= 10000 && credit_number <= 50000 then 'GOLD.'
            when credit_number < 10000 then 'SILVER.'
        end);
    return respuesta;
end;

select usuario( credit_number: 80000000);

```

| `usuario(80000000)` | |
|---------------------|------------|
| 1 | PLATINIUM. |

17. Crear una función que reciba un parámetro TEXT

```

create or replace function descomponer(num text)
returns text
begin

    declare respuesta text default '';
    declare concatenar int default char_length(num);
    declare lim int default 1;
    declare x int default concatenar;

    repeat
        if concatenar >= lim
        then
            set respuesta = concat (substr(num, concatenar, x-1), ' ', ' ', respuesta);
            set concatenar = concatenar - 1;
        end if;
        until concatenar <= 0
        end repeat;
    return(respuesta);
end;

select descomponer( num: 'dbaii');

```

```

descomponer
1 dbai , baii , aii , ii , i ,

```