

BASE DE DATOS II

TAREA DE DEFENSA HITO 3



NOMBRE: APAZA LARA KEVIN MICHAEL
CÓDIGO: SIS13814908
DOCENTE: WILLIAM RODDY BARRA PAREDES
FECHA: Martes 5 de abril, 2022



Defina que es lenguaje procedural en MySQL.

La delegación de tareas específicas a una función

Defina que es una función en MySQL.

Las funciones son piezas de código que reciben datos de entrada, realizan operaciones con ellos y luego devuelven un resultado.

¿Qué cosas características debe de tener una función? Explique sobre el nombre, el return, parametros, etc.

¿Qué cosas características debe de tener una función? Explique sobre el nombre, el return, parametros, etc.

Nombre: con el cual se identifica a la función

Returns: Especifica que devolverá la función

return: Un valor a retornar

Parámetros: En caso de que se requieran

Begin y end: En donde va el código

¿Cómo crear, modificar y cómo eliminar una función?
Adjunte un ejemplo de
su uso.

crear: create function nombre_funcion
eliminar: drop function if exist
modificar: create or replace function

```
create or replace function comparar_cadenas(cadena1 varchar(20), cadena2 varchar(20))  
returns text  
begin  
    declare respuesta text default '';  
    if cadena1 = cadena2 then  
        set respuesta='cadenas iguales';  
    else  
        set respuesta='cadenas diferentes';  
    end if;  
    return respuesta;  
end;  
  
select comparar_cadenas('anamanaguchi','anamanaguchi');
```

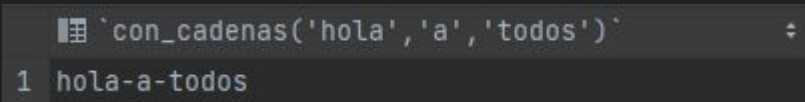
Para qué sirve la función CONCAT y como funciona en MYSQL

- ¿Crear una función que muestre el uso de la función CONCAT?
- La función debe concatenar 3 cadenas.

El concat es usado para concatenar cadenas (unirlas)

```
create function con_cadenas(cadena1 text, cadena2 text, cadena3 text) returns text
begin
    declare respuesta text default '';
    set respuesta=concat(cadena1,'-',cadena2,'-',cadena3);
    return respuesta;
end;
```

```
select con_cadenas('hola','a','todos');
```



The screenshot shows a MySQL query result. The query is `con_cadenas('hola','a','todos')`. The result is a single row with the value `hola-a-todos`.

	con_cadenas('hola','a','todos')
1	hola-a-todos

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

```
create function uso_substring(cadena1 text)
returns text
begin
    declare respuesta text default '';
    set respuesta=substring(cadena1,1,6);
    return respuesta;
end;

select uso_substring('Ximena Condori Mar');
```

El substring es utilizado para definir un rango de letras en una cadena.

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.

Usado para comparar cadenas para ver si son iguales o diferentes si son iguales será 0 si es uno o menos uno serán diferentes.

```
create function uso_substrih(cadena1 text, cadena2 text, cadena3 text) returns text
begin
    declare respuesta text default '';
    declare palabra1 int;
    declare palabra2 int;
    set palabra1= strcmp(cadena1,cadena2);
    set palabra2=strcmp(cadena2,cadena3);
    if palabra2=palabra1 then
        set respuesta = 'iguales';
    else set respuesta = 'no iguales';
    end if;
    return respuesta;
end;
```

```
select uso_substrih('hola','a','todos');
```

```
1 `uso_substrih('hola','a','todos')`
1 no iguales
```


Para qué sirve la función CHAR_LENGTH y LOCATE y como funciona en MYSQL

- ¿Crear una función que muestre el uso de ambas funciones?

char_length usado para averiguar la cantidad de caracteres que posee un texto y locate para verificar en qué posición existe lo buscado.

```
create function uso_char_lenght(palabra text) returns text
begin
    declare respuesta text default '';
    declare cantidad int;
    declare num text;
    set cantidad=char_length(palabra);
    set num=locate('+51',palabra);
    set respuesta = concat(palabra,' tiene ',cantidad,' letras y es numero de telefono',num);
    return respuesta;
end;

select uso_char_lenght('mi numero es +51 72 222');
```

¿Cual es la diferencia entre las funciones de agregación y funciones creados por el DBA? Es decir funciones creadas por el usuario.

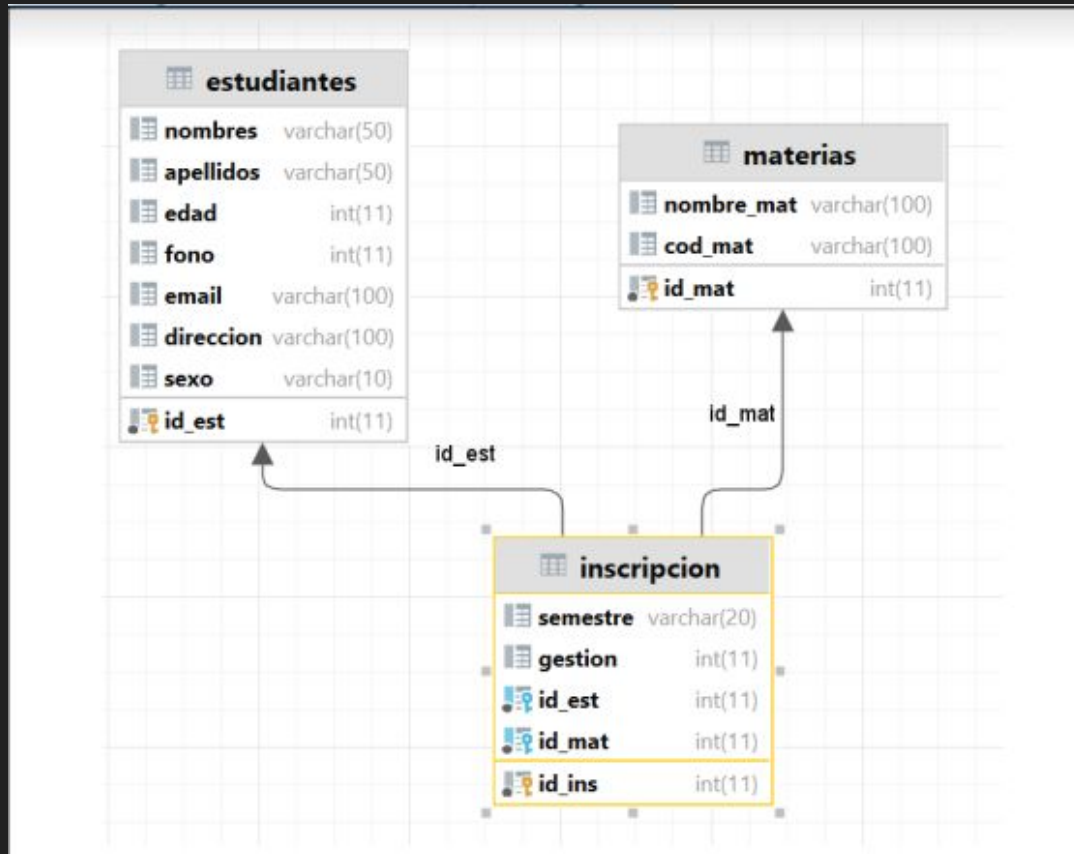
La diferencia radica en que las funciones del gestor se pueden ejecutar en la cláusula where y las creadas por el DBA no.

10.¿Busque y defina a qué se referirá cuando se habla de parámetros de entrada y salida en MySQL?

- Es decir IN INOUT, etc

Parámetros de entrada para los procedimientos almacenados.

Crear la siguiente Base de datos y sus registros.

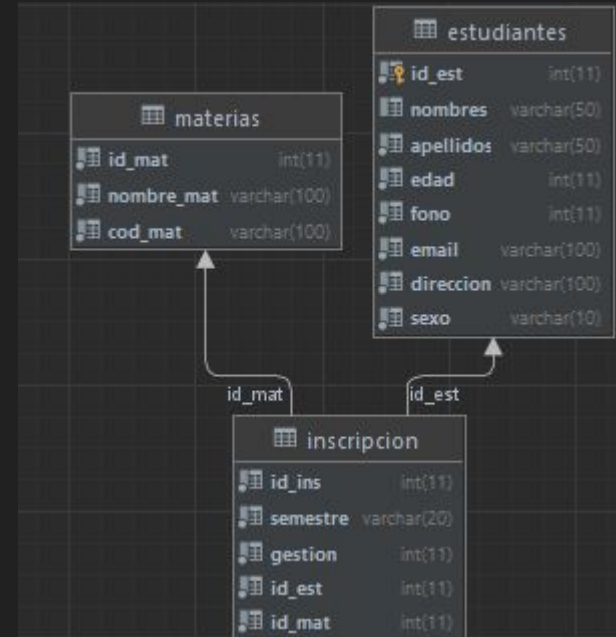


```
create database tarea_hito_3;
use tarea_hito_3;
```

```
CREATE TABLE estudiantes(
    id_est INTEGER AUTO_INCREMENT PRIMARY KEY,
    nombres VARCHAR(50) NOT NULL,
    apellidos VARCHAR(50) NOT NULL,
    edad INTEGER NOT NULL,
    fono INTEGER NOT NULL,
    email VARCHAR(100) NOT NULL ,
    direccion VARCHAR(100) NOT NULL ,
    sexo VARCHAR(10)
);
```

```
CREATE TABLE materias(
    id mat INTEGER AUTO INCREMENT PRIMARY KEY ,
    nombre mat VARCHAR(100) NOT NULL ,
    cod_mat VARCHAR(100) NOT NULL
);
```

```
CREATE TABLE inscripcion(
    id_ins INTEGER AUTO_INCREMENT PRIMARY KEY ,
    semestre VARCHAR(20) NOT NULL ,
    gestion INTEGER NOT NULL ,
    id est INTEGER NOT NULL ,
    id mat INTEGER NOT NULL ,
    FOREIGN KEY (id est) REFERENCES estudiantes (id_est) ,
    FOREIGN KEY (id_mat) REFERENCES materias (id_mat)
);
```



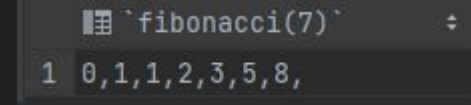
Crear una función que genere la serie Fibonacci.

```
drop function if exists fibonacci;
create function fibonacci(limite int) returns text
begin
    declare respuesta text default '';
    declare suma int default 0;
    declare x int default 0;
    declare y int default 1;
    declare cont int default 0;

    while cont<limite do

        set respuesta = concat(respuesta,x,',');
        set y=x+y;
        set x=y-x;
        set cont=cont+1;
    end while;
    return respuesta;
end;

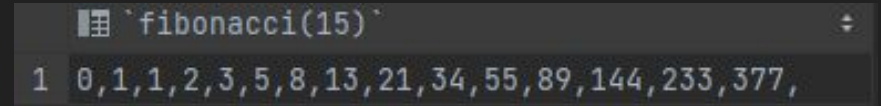
select fibonacci(7);
```



Query: ``fibonacci(7)``

1	0,1,1,2,3,5,8,
---	----------------

```
select fibonacci(15);
```



Query: ``fibonacci(15)``

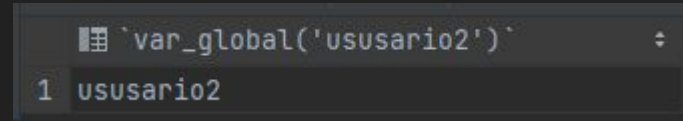
1	0,1,1,2,3,5,8,13,21,34,55,89,144,233,377,
---	---

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

```
set @userAll='ADMIN';

drop function if exists
var_global;
CREATE FUNCTION var_global(DATA
text)
returns text
begin
    set @userAll=data;
    return @userAll;
end;

select var_global('ususuario2');
```



The screenshot shows a SQL query result in a dark-themed interface. The query is `var_global('ususuario2')`. The result is displayed in a table with one row and one column, showing the value `ususuario2`.

	var_global('ususuario2')
1	ususuario2

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

```
create function edad_min()
returns integer
begin
    declare respuesta integer;
    set respuesta=(
        select min(est.edad)
        from estudiantes as est
    );
    return respuesta;
end;
```

```
drop function if exists
numeros_de_edad;
```

```
create function numeros_de_edad() returns text
begin
    declare x integer default 0;
    declare respuesta text default '';
    declare y integer;
    set y = edad_min();
    if y%2=0 then

        edad par: loop
            if x > y then
                leave edad_par;
            end if;

            set respuesta=concat(respuesta,x,',');
            set x=x+2;
            iterate edad_par;
        end loop;
    else

        repeat
            set respuesta=concat(respuesta,y,',');
            set y=y-2;
        until y<=0
        end repeat;
    end if;
    return respuesta;
end;
```

```
select numeros_de_edad();
```

	id_est	nombres	apellidos	edad
1	1	Miguel	Gonzales Veliz	20
2	2	Sandra	Mavir Uria	25
3	3	Joel	Adubiri Mondar	30
4	4	Andrea	Arias Ballesteros	21
5	5	Santos	Montes Valenzuela	24

```
select numeros_de_edad();|
```

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

```
create function contar_letrass(textoo text) returns text
begin
```

```
    declare respuesta text default '';
    declare y int default 0;
    declare x char;
    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;
```

```
select contar_letras('Palabra');
```

FALLA

```
    set y=char_length(textoo);
    repeat
        set y=y+1;
        set x=substring(textoo,y,1);
        case x
            when 'a' then set a=a+1;
            when 'e' then set e=e+1;
            when 'i' then set i=i+1;
            when 'o' then set o=o+1;
            when 'u' then set u=u+1;
            else set x=0;
        end case;
    until y<textoo
end repeat;
set respuesta = concat('a:',a,' e:',e,' i:',i,' o:',o,' u:',u);
return respuesta;
end;
```

16. Crear una función que recibe un parámetro INTEGER.

```
drop function if exists determinar_tipo;

create function determinar_tipo(credit_number integer) returns text
begin
    declare respuesta text default '';

    case credit_number
        when credit_number > 50000 then set respuesta = 'PLATINIUM';
        when credit_number >= 10000 and credit_number <= 50000 then
set respuesta = 'GOLD';
        when credit_number < 10000 then set respuesta = 'SILVER';
        else set respuesta = credit_number;
    end case;

    return respuesta;
end;

select determinar_tipo(5000);
select determinar_tipo(30000);
select determinar_tipo(5000);
```

FALLA