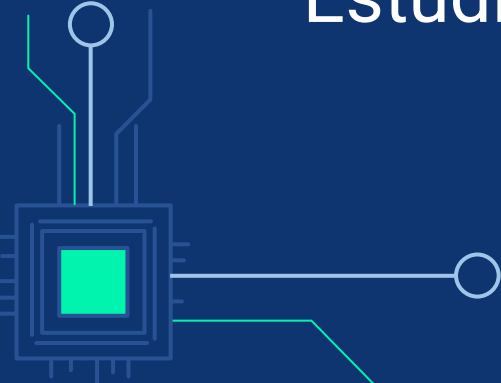




PROCESUAL BASE DE DATOS II

Estudiante: Mijael Jhonatan Rojas Arias



The background of the image features a person's face and hands interacting with a futuristic digital interface. The person's face is partially visible at the top, and their hands are shown at the bottom, touching a glowing, circular digital element. The interface is composed of various blue and white geometric shapes, lines, and patterns, including a large circular gauge or dial in the center. The overall aesthetic is high-tech and modern.

MANEJO DE CONCEPTOS

¿A QUE SE REFIERE CUANDO SE HABLA DE BASE DE DATOS RELACIONALES?

R.- Son la que su pueden relacionar entre si, son SQL y utilizan tablas.

¿A QUE SE REFIERE CUANDO SE HABLA DE BASES DE DATOS NO RELACIONALES?

R.- Son las que no son SQL y utilizan documentos como ser JSON.

¿QUÉ ES MYSQL Y MARÍADB? EXPLIQUE SI EXISTEN DIFERENCIAS O SON IGUALES, ETC

R.- MariaDB es una copia de la MySQL cuando esta empezó a ser a paga.

¿QUE SON LAS FUNCIONES DE AGREGACION?

R.- Son las funciones que están predeterminadamente y que solo se puede ejecutar en la clausula select.

¿QUÉ LLEGARÍA SER XAMPP?

R.- Es un servidor que nos proporciona MySQL.

¿CUÁL ES LA DIFERENCIA ENTRE LAS FUNCIONES DE AGREGACIÓN Y FUNCIONES CREADOS POR EL DBA? ES DECIR FUNCIONES CREADAS POR EL USUARIO

R.- Las funciones de agregación ya están definidas y solo se ejecutan en la clausula select, las funciones creadas por el DBA son aquellas que se pueden ejecutar en la clausula select y where.

¿PARA QUE SIRVE EL COMANDO USE?

R.- Con el comando “use” se utiliza para entrar a una Base de Datos.

¿QUÉ ES DML Y QUE ES DDL?

R.- DDL es Data Definition Language que define la estructura que almacenara datos, DML es Data Manipulation Languaje que permite realizar acciones a los datos en la base de datos.

¿QUÉ COSAS CARACTERÍSTICAS DEBE TENER UNA FUNCIÓN?

EXPLIQUE SOBRE EL NOMBRE, RETURN, PARÁMETROS, ETC

R.- El nombre es la forma de identificar la función creada, el return es el datos que estas retornando, el returns especifica el dato que vas a retornar, los parámetros son los datos que ingresan a la función.

¿CÓMO CREAR, MODIFICAR Y COMO ELIMINAR UNA FUNCIÓN?

R.- Para crear una función se utiliza el “CREATE FUNCTION”

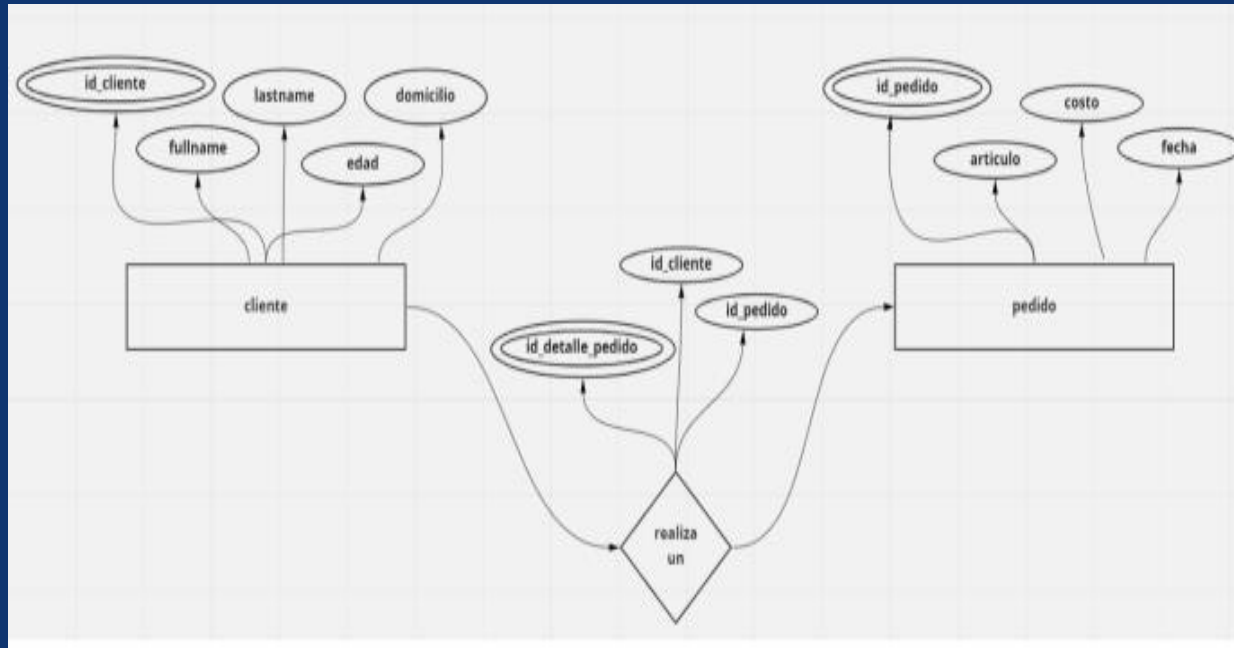
Para modificar una función se utiliza el “CREATE OR REPLACE FUNCTION.”

Para eliminar una función se utiliza el “DROP FUNCTION (nombre de la función)”



PARTE PRACTICA

CREAR LAS TABLAS Y 2 REGISTROS PARA CADA TABLA PARA EL SIGUIENTE MODELO ER



Tablas

```
create table cliente (  
    id_cliente int primary key not null,  
    fullname varchar (30) not null,  
    lastname varchar (30) not null,  
    edada int not null,  
    domicilio varchar (20) not null  
);
```

```
create table pedido (  
    id_pedido int auto_increment primary key not null,  
    articulo varchar (20) not null,  
    costo int not null,  
    fecha date not null  
);
```

```
create table detalle_pedido (  
    id_detalle_pedido int auto_increment primary key not null,  
    id_cliente int not null,  
    id_pedido int not null,  
    foreign key (id_cliente) references cliente (id_cliente),  
    foreign key (id_pedido) references pedido (id_pedido)  
);
```

Registros

```
insert into cliente (id_cliente, fullname, lastname, edada, domicilio)  
values (123451, 'nombre1', 'apellido1', 21, 'domicilio1'),  
      (123452, 'nombre2', 'apellido2', 22, 'domicilio2'),  
      (123453, 'nombre3', 'apellido3', 23, 'domicilio3'),  
      (123454, 'nombre4', 'apellido4', 24, 'domicilio4'),  
      (123455, 'nombre5', 'apellido5', 25, 'domicilio5');
```

```
insert into pedido (articulo, costo, fecha)  
values ('articulo1', 22, '1/2/21'),  
      ('articulo2', 22, '17/2/21'),  
      ('articulo3', 45, '4/3/21'),  
      ('articulo4', 45, '25/5/21'),  
      ('articulo5', 32, '26/12/21');
```

```
insert into detalle_pedido (id_cliente, id_pedido)  
values (123451, 1),  
      (123452, 2),  
      (123453, 3),  
      (123454, 4),  
      (123455, 5);
```

CREAR UNA CONSULTA SQL EN BASE AL EJERCICIO ANTERIOR

```
select cli.fullname, cli.lastname, ped.articulo, ped.costo
from cliente as cli
inner join detalle_pedido as dp on cli.id_cliente = dp.id_cliente
inner join pedido as ped on dp.id_pedido = ped.id_pedido
where ped.costo >= 32
```

	■ fullname	÷	■ lastname	÷	■ articulo	÷	■ costo	÷
1	nombre3		apellido3		articulo3		45	
2	nombre4		apellido4		articulo4		45	
3	nombre5		apellido5		articulo5		32	

CREAR UN FUNCIÓN QUE COMPARE DOS CÓDIGOS DE MATERIA.

Mostrar los nombres y apellidos de los estudiantes inscritos en la materia ARQ-105, adicionalmente mostrar el nombre de la materia.

```
create or replace function comparaMaterias (codigo varchar (100), materia varchar (10))  
returns boolean  
begin  
    declare flag boolean;  
    set flag = 0;  
    if (codigo = materia)  
    then  
        set flag = 1;  
    end if;  
    return (flag);  
end;
```

```
select est.id_est, est.nombres, est.apellidos, mat.nombre_mat, mat.cod_mat  
from estudiantes as est  
inner join inscripcion as ins on est.id_est = ins.id_est  
inner join materias as mat on ins.id_mat = mat.id_mat  
where comparaMaterias(codigo: mat.cod_mat, materia: 'ARQ-105');
```

	id_est	nombres	apellidos	nombre_mat	cod_mat
1	5	Santos	Montes Valenzuela	Fisica Basica	ARQ-105

CREAR UNA FUNCIÓN QUE PERMITA OBTENER EL PROMEDIO DE LAS EDADES DEL GÉNERO MASCULINO O FEMENINO DE LOS ESTUDIANTES INSCRITOS EN LA ASIGNATURA ARQ-104.

```
create function avg_edad_estudiantes(genero varchar(15))
  returns integer
begin
  return (select avg(est.edad)
          from estudiantes as est
          inner join inscripcion as ins on est.id_est = ins.id_est
          inner join materias as mat on ins.id_mat = mat.id_mat
          where mat.cod_mat = 'ARQ-104' and est.sexo = genero);
end;
```

```
select avg_edad_estudiantes(genero: 'femenino') as promedioEdad
```

	promedioEdad
1	23

CREAR UNA FUNCIÓN QUE PERMITA CONCATENAR 3 CADENAS

```
create function concatenar (par1 varchar (20), par2 varchar (20), par3 varchar (20))  
returns varchar (70)  
begin  
    declare concatenar varchar (70);  
    set concatenar = concat(par1, ' - ', par2, ' - ', par3);  
    return (concatenar);  
end;
```

```
select concatenar( par1: 'pepito', par2: 'perez', par3: 50) as concatenar ~
```

	concatenar
1	pepito - perez - 50

CREAR UNA FUNCIÓN DE ACUERDO A LO SIGUIENTE:

Mostrar el nombre, apellidos y el semestre de todos los estudiantes que estén inscritos. Siempre y cuando la suma de las edades del sexo femenino o masculino sea par y mayores a cierta edad.

```
create function suma_genero_mayor_edad (genero varchar (20), edad int)
returns int
begin
    return (select sum(est.edad)
            from estudiantes as est
            where est.sexo = genero and est.edad > edad);
end;
```

```
select est.nombres, est.apellidos, ins.semestre
from estudiantes as est
inner join inscripcion as ins on est.id_est = ins.id_est
inner join materias as mat on ins.id_mat = mat.id_mat
where suma_genero_mayor_edad ( genero: 'masculino', edad: 22) % 2 = 0 and est.sexo = 'masculino' and est.edad > 22;
```

	nombres	apellidos	semestre
1	Joel	Adubiri Mondar	2do Semestre
2	Joel	Adubiri Mondar	3er Semestre
3	Santos	Montes Valenzuela	5to Semestre

CREAR UNA FUNCIÓN DE ACUERDO A LO SIGUIENTE:

Crear una función sobre la tabla estudiantes que compara un nombre y apellidos. (si existe este nombre y apellido mostrar todos los datos del estudiante).

```
create or replace function compararNombres (nombre varchar (20), apellido varchar (20))
returns boolean
begin
    declare flag boolean;
    declare nombre_apellido varchar(100);
    set nombre_apellido = (select est.nombres
                           from estudiantes as est
                           where est.nombres = nombre and est.apellidos = apellido);
    if (nombre_apellido is not null)
    then
        set flag = 1;
    else
        set flag = 0;
    end if;
    return (flag);
end;
```

```
select est.*
from estudiantes as est
where compararNombres( nombre: 'Sandra', apellido: 'Mavir Uria') and est.nombres = 'Sandra' and est.apellidos = 'Mavir Uria';
```

id_est	nombres	apellidos	edad	gestion	fono	email	direccion	sexo
1	2 Sandra	Mavir Uria	25	<null>	2832116	sandra@gmail.com	Av. 6 de Agosto	femenino



THANKS!