**SELECT**

Es la función más utilizada en bases de datos ya que ella nos permite obtener el dato preciso o un conjunto de datos que el cliente desea saber, por ejemplo se cuenta con una tabla con dueños y otra tabla de mascotas, donde se desea saber cuales son las mascotas del dueño ‘Pedro Perez’ o saber Cual es el dueño de la mascota ‘Rocinante’. Ej.

Dada las siguiente base de datos llamada laVeterinaria:

**PROPIETARIO**

|  |  |
| --- | --- |
| Propietario | |
| cedulaProp | nombreProp |
| 13999888 | Raul Fulton |
| 63444555 | Laura Jimenez |
| 110177788 | Silvia Santos |

CREATE TABLE Propietario (

cedulaProp int not null PRIMARY KEY,

nombreProp varchar(55) NOT null );

-----------

INSERT into propietario(cedulaProp,nombreProp)

VALUES(13999888,"Raul Fulton"),

(63444555,"Laura Jimenez"),

(110177788, "Silvia Santos");

**RAZA**

|  |  |
| --- | --- |
| Raza | |
| idRaza | nombreRaza |
| 1 | Canino |
| 2 | Fenilo |
| 3 | Roedor |
| 4 | Lagarto |

CREATE TABLE Raza (

idRaza int not Null AUTO\_INCREMENT PRIMARY key,

nombreRaza varchar(55) not null

);

----

INSERT into raza(idRaza,nombreRaza)

VALUES(null,"Canino"),

(null,"Felino"),

(null, "Roedor"),

(null,"Lagarto");

**VETERINARIA**

|  |  |  |
| --- | --- | --- |
| Veterinaria | | |
| idVet | nombreVet | direccionVet |
| 1 | Hacienda | Cra 6 No 4 – 89 |
| 2 | Scott | Cll 5 No. 23 – 4 |
| 3 | Animalisto | Cra 9 No 3 – 45 |

CREATE TABLE Veterinaria (

idVet int not null AUTO\_INCREMENT PRIMARY KEY,

nombreVet varchar(55)not null ,

direccionVet varchar (55)not null );

--

INSERT into veterinaria(idVet,nombreVet,direccionVet)

VALUES(null,"Hacienda","Cra 6 No 4-89"),

(null,"Scott","Cll 5 No.23-4"),

(null, "Animalismo","Cra 9 No 3-45");

**MASCOTA**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Mascota | | | | | |
| idMascota | nombreMascota | idRazaMascota | pesoMascot | CumpleMascot | propietarioMascota |
| 1 | Tony | 1 | 1010 | 2005-03-18 | 13999888 |
| 2 | Rintintin | 1 | 2205 | 2000-07-23 | 13999888 |
| 3 | Lassie | 1 | 4300 | 2002-02-13 | 13999888 |
| 4 | Paspartuk | 2 | 701 | 2008-10-16 | 63444555 |
| 5 | Stuart | 3 | 122 | 2011-02-17 | 110177788 |
| 6 | Carraman | 2 | 405 | 2007-01-30 | 63444555 |

CREATE table mascota(

idMascota int not null AUTO\_INCREMENT PRIMARY key,

nombreMascota varchar (50) not null ,

idRaza int not null,

pesoMascot int not null,

cumpleMascot date not null,

propietarioMascot int NOT null,

FOREIGN KEY (idRaza)REFERENCES raza(idRaza),

FOREIGN KEY (propietarioMascot)REFERENCES propietario(cedulaProp) );

--

INSERT INTO mascota (nombreMascota, idRaza, pesoMascot, cumpleMascot, propietarioMascot) VALUES

('Tony', 1, 1010, '2005-03-18',13999888),

('Rintintin', 1,2205, '2000-07-23',13999888),

('Lassie', 1, 4300, '2002-02-13',13999888),

('Paspartuk', 2, 701, '2008-10-16', 63444555),

('Stuart', 3, 122, '2007-01-30', 110177788),

('Carraman', 2, 405, '2007-01-30', 63444555);

**ATENCION**

|  |  |  |  |
| --- | --- | --- | --- |
| Atencion | | | |
| idAtencion | idMasAte | idServiAte | idVetAte |
| 1 | 1 | 1 | 1 |
| 2 | 3 | 2 | 1 |
| 3 | 6 | 2 | 3 |
| 4 | 4 | 3 | 2 |

CREATE TABLE atencion (

idAtencion int NOT null AUTO\_INCREMENT PRIMARY KEY ,

idMasAte int not null,

idServiAte int not null,

idVetAte int not null,

FOREIGN key (idMasAte)REFERENCES mascota(idMascota),

FOREIGN KEY (idServiAte)REFERENCES servicio (idServicio),

FOREIGN KEY (idVetAte)REFERENCES veterinaria(idVet));

--

INSERT into atencion (idAtencion ,idMasAte,idServiAte,idVetAte)VALUES(null,1,1,1),

(null,3,2,1),

(null,6,2,3),

(null,4,3,2);

**SERVICIO**

|  |  |
| --- | --- |
| Servicio | |
| idServicio | descripcionServicio |
| 1 | Peluqueria |
| 2 | Infertilizacion de 90 dias |
| 3 | Consulta medica |

CREATE TABLE servicio(

idSeervicio int not null AUTO\_INCREMENT PRIMARY KEY ,

descripcionServicio varchar (55) not null

);

----

INSERT into servicio (idServicio,descripcionServicio)VALUES

(null,"Peluqueria"),

(null,"Infertilizacion de 90 dias"),

(null, "Consulta medica");

3.2 Realizar las anteriores tablas usando el comando CREATE TABLE (tenga en cuenta que algunas tablas tienen llave foranea), realizar las siguientes consultas y copiar el resultado en word como SQLveterinaria, posteriormente subirlos a la plataforma la evidencia.

Se desea saber cuales son todos los datos de los propietarios registrados

SELECT **\*** FROM Propietario;

* SELECT **\*** FROM Propietario;

Se desea saber cuales son los nombres y direcciones de todas las veterinarias registradas

SELECT nombreVet, direccionVen FROM Veterinaria;

* SELECT nombreVet, direccionVet FROM Veterinaria;

Se desea saber cuales son los nombres y pesos de las mascotas donde el peso sea **mayor** a 1000

SELECT nombreMascota, pesoMascot FROM Mascota WHERE pesoMascot **>** 1000;

* SELECT nombreMascota, pesoMascot FROM Mascota WHERE pesoMascot > 1000;

Se desea saber cuales son los nombres de las mascotas cuyo peso sea **entre** 700 **y** 1500

SELECT nombreMascota FROM Mascota WHERE pesoMascot **BETWEEN** 700 **AND** 1500;

* SELECT nombreMascota FROM Mascota WHERE pesoMascot BETWEEN 700 AND 1500;

Se desea saber cuales son los nombre de mascotas **ordenados por** peso de manera **descentente**

SELECT nombreMascota FROM Mascota **ORDER BY** pesoMascot **DESC;**

* SELECT nombreMascota FROM Mascota ORDER BY `Mascota`.`nombreMascota` DESC

Se desea saber cuales son los nombres de mascotas del propietario ‘Raul Fulton’

SELECT nombreMascota FROM Mascota WHERE propietarioMascota **IN** (SELECT cedulaProp FROM Propietario WHERE nombreProp **LIKE** ‘Raul Fulton’);

* SELECT nombreMascota FROM Mascota WHERE propietarioMascot IN (SELECT cedulaProp FROM Propietario WHERE nombreProp LIKE "Raul Fulton");

Se desea saber cuales son los nombres de mascotas **junto con** su respectivo propietario.

SELECT Mascotas.nombreMascota, Propietario.nombreProp FROM Mascotas **INNER JOIN** Propietario **ON** Mascota.propietarioMascota = Propietario.cedulaProp

* SELECT mascota.nombreMascota, propietario.nombreProp FROM mascota INNER JOIN propietario ON mascota.propietarioMascot = propietario.cedulaProp;

Se desea saber cuales el nombre de la mascota junto con la raza

SELECT Mascota.nombreMascota, Raza.nombreRaza FROM Mascota INNER JOIN Raza ON Mascota.idRazaMascota = Raza.idRaza;

* SELECT mascota.nombreMascota, raza.nombreRaza FROM mascota INNER JOIN raza ON mascota.idRaza = raza.idRaza;

Se desea saber los datos de la atencion prestada a las mascotas

SELECT Mascota.nombreMascota, Servicio.descripcionServicio, Veterinaria.nombreVet FROM Atencion INNER JOIN Mascota ON Atencion.idMasAte = Mascota.idMascota

INNER JOIN Servicio ON Atencion.idServAte = Servicio.idServicio

INNER JOIN Veterinaria ON Atencion.idVetAte = Veterinaria.idVeterinaria;

* SELECT mascota.nombreMascota, servicio.descripcionServicio, veterinaria.nombreVet FROM atencion INNER JOIN mascota ON atencion.idMasAte = mascota.idMascota

INNER JOIN servicio ON atencion.idServiAte = servicio.idServicio

INNER JOIN veterinaria ON atencion.idVetAte = veterinaria.idVet;

Se desea saber la **cantidad** de Roedores registrados

SELECT **COUNT**(\*) FROM Mascota WHERE idRazaMascota IN(SELECT idRaza FROM Raza WHERE nombreRaza LIKE ‘Roedor’ );

* SELECT COUNT(\*)FROM mascota where idRaza IN (SELECT idRaza FROM raza WHERE nombreRaza LIKE"Roedor");

Se desea saber cual es el **promedio** de peso de todas las mascotas agrupados por raza mostrando el nombre de la raza

SELECT Raza.nombreRaza, **AVG**(Mascota.pesoMascot) FROM Mascota

INNER JOIN Raza ON Mascota.idRazaMascota = Raza.idRaza

GROUP BY Mascota.idRazaMascota

* SELECT raza.nombreRaza,AVG (mascota.pesoMascot) FROM mascota INNER JOIN raza on mascota.idRaza =raza.idRaza GROUP BY mascota.idRaza

Se desea saber cual es la mascota **más** pesada:

SELECT pesoMascota, nombreMascota FROM Mascota WHERE pesoMascota = (SELECT **MAX**(pesoMascota) FROM Mascota);

* SELECT pesoMascot, nombreMascota

FROM mascota

WHERE pesoMascot = (SELECT MAX(pesoMascot) FROM mascota);

Otra manera de averiguarlo es ordenando **desc**endentemente los pesos y seleccionando el primer registro:

SELECT pesoMascota, nombreMascota FROM Mascota ORDER BY pesoMascota **DESC** LIMIT 1

* SELECT pesoMascot,nombreMascota FROM mascota ORDER BY pesoMascot DESC limit 1

Se desea saber cual es la mascota con el peso menor.

SELECT pesoMascota, nombreMascota FROM Mascota WHERE pesoMascota = (SELECT **MIN**(pesoMascota) FROM Mascota);

* SELECT pesoMascot,nombreMascota FROM mascota WHERE pesoMascot= (SELECT MIN(pesoMascot) FROM mascota);

Se desea saber cual es la **sumatoria** de pesos por raza

SELECT Raza.nombreRaza, **SUM**(pesoMascota) FROM Mascota

INNER JOIN Raza ON Mascota.idRazaMascota = Raza.idRaza

GROUP BY Mascota.idRazaMascota

* SELECT Raza.nombreRaza, SUM(pesoMascot) FROM Mascota

INNER JOIN Raza ON Mascota.idRaza = Raza.idRaza

GROUP BY Mascota.idRaza

3.3 Dada las siguiente base de datos llamado sistemabancario (crearla en phpmyadmin):

Diagrama

Descripción generada automáticamente

Las siguientes scripts y los resultados debe guardarlo archivo word llamado sqlbancario.doc y subir la evidencia al LMS

a. Cree las tablas usando el comando CREATE TABLE tenga en cuenta el uso de REFERENCES.

-CREATE TABLE cliente (

idCliente int not null AUTO\_INCREMENT PRIMARY KEY,

documentoCliente varchar(16)not null,

nombreCliente varchar(45) not null,

cedulaCliente varchar (14)not null,

direccionCliente varchar (50) not null

);

- CREATE TABLE tipoCuenta (

idTipoCuenta tinyint not null AUTO\_INCREMENT PRIMARY KEY,

descripcionTipoCuenta varchar(45) not null

);

- CREATE TABLE cuenta (

numeroCuenta int not null AUTO\_INCREMENT primary key ,

saldoCuenta int not null,

idTipoCuenta tinyint not null ,

idCliente int not null,

foreign key (idTipoCuenta) references tipocuenta(idTipoCuenta),

foreign key (idCliente) references cliente (idCliente)

);

- create table prestamo (

idPrestamo int not null AUTO\_INCREMENT primary key ,

montoPrestamo int not null,

fechaPrestamo date not null,

idCliente int not null,

mesesPago tinyint not null,

foreign key (idCliente) references cliente(idCliente)

);

- create table pagos (

idPagos int not null AUTO\_INCREMENT primary key ,

valorPago int not null,

fechaPago date not null ,

idPrestamo int not null,

foreign key (idPrestamo) references prestamo (idPrestamo)

);

b. Genere la vista de relaciones usando la opcion Estructura para cada tabla.

Interfaz de usuario gráfica

Descripción generada automáticamente con confianza media

c. Inserte 3 Clientes, 3 tipos de cuenta (ahorro, corriente, rendimax), 5 Cuentas, 4 Prestamos y 3 pagos.

- insert into cliente (idCliente,documentoCliente,nombreCliente,celularCliente,direccionCliente)

values (null,100000,"hector Mancipe",3234402236,"velez"),

(null,100001,"Aleja Mancipe",3208209285,"velez"),

(null,100002,"Manuel Gonzalez",3208209285,"velez"

);

- insert into tipocuenta (idTipoCuenta,descripcionTipoCuenta)

values (null,"ahorro"),

(null,"corriente"),

(null,"rendimax"

);

-insert into cuenta (numeroCuenta,saldoCuenta,idTipoCuenta,idCliente) VALUES(null,300,2,1),

(null,1000000,1,2),

(null,230000,3,3),

(null,200000,1,1),

(null,400000,2,2)

;

-insert into prestamo (idPrestamo,montoPrestamo,fechaPrestamo,idCliente,mesesPago)

VALUES(null,100000,'2024-01-15',1,4),

(null,50000,'2024-03-18',1,1),

(null,230000,'2024-02-13',2,12),

(null,670000,'2023-10-05',3,9)

;

-insert into pagos(idPagos,valorPago,fechaPago,idPrestamo)

VALUES (null,1000,'2024-03-9',1),

(null,15000,'2024-02-9',2),

(null,1000,'2024-04-9',3)

;

3.3.1 En un archivo word escriba las siguientes consultas y los resultados de cada una:

a. Se desea saber todos los datos de los clientes registrados

-[SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/select.html) \*FROM cliente

Interfaz de usuario gráfica

Descripción generada automáticamente

b. Se desea saber el numero de cuenta y que tipo de cuenta es de las cuentas registradas

[SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/select.html) cuenta.numeroCuenta, tipoCuenta.descripcionTipoCuenta FROM cuenta INNER JOIN tipoCuenta ON cuenta.idTipoCuenta = tipoCuenta.idTipoCuenta;

Tabla

Descripción generada automáticamente

c. Se desea saber el nombre del cliente, su cuenta y el tipo de cuenta que posee.

[SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/select.html) cliente.nombreCliente,cuenta.numeroCuenta,tipocuenta.descripcionTipoCuenta FROM cuenta INNER JOIN cliente on cuenta.idCliente = cliente.idCliente INNER JOIN tipocuenta ON cuenta.idTipoCuenta = tipocuenta.idTipoCuenta;

Tabla

Descripción generada automáticamente

d. Se desea saber cual es el valor de pago más alto que hay registrado

[SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/select.html) valorPago,idPagos from pagos WHERE valorPago=([SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/select.html) [MAX](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/aggregate-functions.html%23function_max)(valorPago) from pagos);

Interfaz de usuario gráfica

Descripción generada automáticamente con confianza media

e. Se desea saber cual es el prestamo más pequeño que hay registrado

[SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/select.html) montoPrestamo,idPrestamo FROM prestamo WHERE montoPrestamo=([select](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/select.html) [MIN](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/aggregate-functions.html%23function_min)(montoPrestamo) from prestamo);

Interfaz de usuario gráfica

Descripción generada automáticamente

f. Se desea saber cual el monto del prestamo y nombre de cliente que tiene el prestamos más alto.

[SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/select.html) cliente.nombreCliente, prestamo.montoPrestamo FROM cliente inner join prestamo on cliente.idCliente = prestamo.idCliente WHERE montoPrestamo = ( [select](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/select.html) [MAX](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/aggregate-functions.html%23function_max)(montoPrestamo) from prestamo);

Imagen que contiene Texto

Descripción generada automáticamente

g. Cuales son las cuentas que tienen un saldo mayor a 40000

[select](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/select.html) cliente.nombreCliente,cuenta.saldoCuenta FROM cliente inner join cuenta on cliente.idCliente= cuenta.idCliente WHERE saldoCuenta > 400000;

Imagen que contiene Tabla

Descripción generada automáticamente

h. Cuales son los clientes que tienen el monto de prestamos > que el saldo de Cuenta

[select](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/select.html) cliente.nombreCliente,cuenta.saldoCuenta,prestamo.montoPrestamo FROM cliente inner join cuenta on cliente.idCliente= cuenta.idCliente INNER JOIN prestamo on cliente.idCliente= prestamo.idCliente WHERE montoPrestamo >saldoCuenta;

Tabla

Descripción generada automáticamente

i. Se desea saber la totalidad de saldo que cada cliente tiene de sus respectivas cuentas.

[select](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/select.html) cliente.nombreCliente,cuenta.saldoCuenta FROM cliente inner join cuenta on cliente.idCliente= cuenta.idCliente GROUP BY cliente.nombreCliente;

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

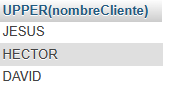
3.5 Cree tabla de clientes, libros, canción, autor, finca, mascota, película, estudiante para ejecutar las siguientes consultas y guardar el resultado en un archivo Word.

Funciones de texto:

Mostrar los nombres de los clientes en MAYUSCULAS

SELECT **UPPER**(nombreCliente) FROM Clientes:

[SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/select.html) UPPER(nombreCliente) FROM Cliente;



Mostrar el titulo del libro en minúscula

SELECT **LOWER**(tituloLibro) FROM Libros

[SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/select.html) LOWER(nombreLibro) FROM Libros;



Colocar un nombres a las columnas de una consulta:

SELECT codCan **AS** ‘Codigo Cancion’, artCan **AS** ‘Artista’, durCan **AS** ‘Duracion’ FROM Canción

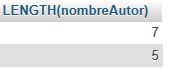
[SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/select.html) codCan AS "Codigo Cancion", artCan AS "Artista", durCan AS "Duracion" FROM cancion;



Saber la cantidad de letras (incluye espacios, números, símbolos) del nombre de un Autor

SELECT **LENGTH**(nombreAutor) FROM Autor:

[SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/select.html) LENGTH(nombreAutor) FROM autor;



Obtener las 7 primeras letras (incluyes espacios, número, símbolos) del nombre de una finca:

SELECT **LEFT**(nombreFinca,7) FROM Finca

[SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/select.html) [LEFT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/string-functions.html%23function_left)(nombreFinca,7) FROM Finca;

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente con confianza media

Obtener las ultimas 5 letras (incluye espacios, números, símbolos) del nombre de una mascota

SELECT **RIGHT**(nombreMascota) FROM Mascota;

[SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/select.html) [RIGHT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/string-functions.html%23function_right)(nombreMascota,5) FROM mascota;



Obtiene una sección de 5 letras (incluye espacios, números, símbolos) desde el tercer carácter

SELECT **SUBSTRING** (tituloPelicula, 3, 5 );

[SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/select.html) SUBSTRING(nombreLibro, 2, 4) from libros;

Texto

Descripción generada automáticamente con confianza media

Obtiene la unión de varios textos en un solo campo

SELECT **CONCAT** (nombreEstudiante, apellidoEstudiante) FROM Estudiante;

[SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/select.html) CONCAT (idCliente,nombreCliente) FROM cliente;

Patrón de fondo

Descripción generada automáticamente con confianza media

Patrones de búsqueda con **LIKE**

Obtener el estudiante que se llame Maria Moreno

SELECT nombreEstudiante FROM Cliente WHERE nombreEstudiante **LIKE** ‘Maria Moreno’

[SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/select.html) nombreCliente FROM Cliente WHERE nombreCliente [LIKE](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/string-comparison-functions.html%23operator_like) 'david';

Imagen que contiene Diagrama

Descripción generada automáticamente

Obtener todos los clientes que comienzan con ‘Ar’

SELECT nombreClientes FROM Clientes WHERE nombreClientes **LIKE** ‘Ar%’;

[SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/select.html) nombreLibro FROM libros WHERE nombreLibro [LIKE](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/string-comparison-functions.html%23operator_like) 'el%';

Interfaz de usuario gráfica

Descripción generada automáticamente

Obtener todos las canciones que terminan con ‘ces’

SELECT nombreCancion FROM Canciones WHERE nombreCancion **LIKE** ‘%ces’

[SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/select.html) nombreLibro FROM libros WHERE nombrelibro [LIKE](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/string-comparison-functions.html%23operator_like) '%ito'

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Obtener todas las películas que inician con ‘Los’ y termina con ‘res’

SELECT nombrePelicula FROM Pelicula WHERE nombrePelicula **LIKE** ‘Los%res’

[SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/select.html) nombreLibro FROM libros WHERE nombrelibro [LIKE](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/string-comparison-functions.html%23operator_like) 'el%ito';

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Obtener todos los actores que tengan en su nombre ‘ardo’

SELECT nombreActor FROM Actores WHERE nombreActor **LIKE** ‘%ardo%’

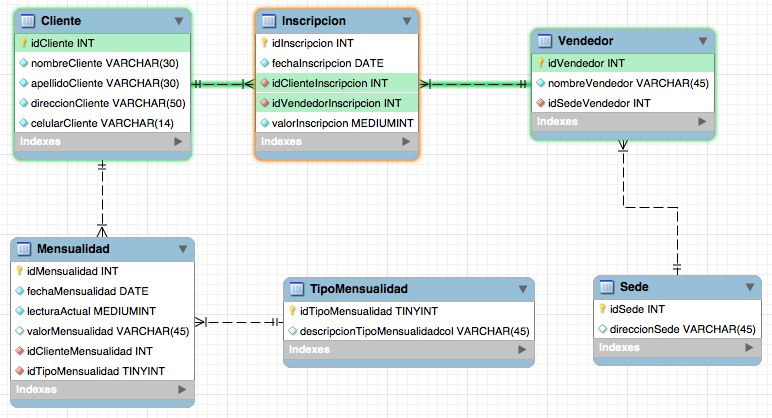
[SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/select.html) nombreLibro FROM libros WHERE nombrelibro [LIKE](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/string-comparison-functions.html%23operator_like) '%ito%';

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

(Generalmente se realiza una conversión a mayúsculas o minúsculas )

3.6 Dada la base siguiente base de datos llamada acueducto: (crearla en phpmyadmin )



[CREATE](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/create-table.html) [TABLE](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/create-table.html) cliente ( idCliente int [not](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/logical-operators.html%23operator_not) null AUTO\_INCREMENT primary key, nombreCliente varchar(30) [not](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/logical-operators.html%23operator_not) null, apellidoCliente varchar(30) [not](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/logical-operators.html%23operator_not) null, direccionCliente varchar(30) [not](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/logical-operators.html%23operator_not) null, celularCliente varchar(30)[not](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/logical-operators.html%23operator_not) null );

[INSERT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/insert.html) into cliente (idCliente ,nombreCliente,apellidoCliente,direccionCliente,celularCliente) [VALUES](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/miscellaneous-functions.html%23function_values) (null,'hector','mancipe','velez','3202343434'), (null,"manuel","mancipe","velez","3244566576"), (null,"jesus","gonzalez","tunja","32448796576"), (null,"David","mancipe","velez","3244566576"), (null,"julio","lopez","tunja","3244987676");

[CREATE](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/create-table.html) [table](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/create-table.html) sede( idSede int [not](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/logical-operators.html%23operator_not) null AUTO\_INCREMENT primary key, direccionSede varchar(45));

[insert](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/insert.html) into sede (idSede,direccionSede) [values](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/miscellaneous-functions.html%23function_values)(null,'velez'), (null,'tunja'), (null,'bogota'), (null,'boyaca'), (null,'velez');

[create](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/create-table.html) [TABLE](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/create-table.html) vendedor ( idVendedor int [not](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/logical-operators.html%23operator_not) null AUTO\_INCREMENT PRIMARY KEY, nombreVendedor varchar(45) [not](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/logical-operators.html%23operator_not) null, idSede int [not](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/logical-operators.html%23operator_not) null, foreign key (idSede) REFERENCES sede(idSede));

[INSERT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/insert.html) into vendedor(idVendedor,nombreVendedor,idSede) [VALUES](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/miscellaneous-functions.html%23function_values)(null,'romulo',1), (null,'argemiro',2), (null,'jose',3), (null,'ricardo',4), (null,'reinaldo',5);

[create](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/create-table.html) [table](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/create-table.html) inscripcion ( idInscripcion int [not](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/logical-operators.html%23operator_not) null AUTO\_INCREMENT primary key , fechaInscripcion date [not](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/logical-operators.html%23operator_not) null, idCliente int [not](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/logical-operators.html%23operator_not) null, idVendedor int [not](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/logical-operators.html%23operator_not" \t "mysql_doc) null,valorInscripcion not null foreign key (idCliente) REFERENCES cliente (idCliente), foreign key (idVendedor) REFERENCES vendedor (idVendedor));

[insert](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/insert.html) into inscripcion (idInscripcion,fechaInscripcion,idCliente,idVendedor,valorInscripcion) [VALUES](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/miscellaneous-functions.html%23function_values)(null,"23-03-2012",1,1,2300), (null,"24-08-2010",2,2,89000), (null,"23-09-2015",3,3,2000), (null,"11-04-2012",4,4,22300), (null,"23-08-2022",5,5,15000);

[create](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/create-table.html) [table](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/create-table.html) tipoMensualidad ( idTipoMensualidad tinyint [not](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/logical-operators.html%23operator_not) null AUTO\_INCREMENT PRIMARY KEY ,descripcionMensualidad varchar(45) )

[INSERT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/insert.html) into tipomensualidad (idTipoMensualidad,descripcionMensualidad) [VALUES](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/miscellaneous-functions.html%23function_values)(null,"consumo"), (null,"reconexion"), (null,"ampliacion"), (null,"divicion");

[create](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/create-table.html) [TABLE](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/create-table.html) mensualidad (idMensualidad int [not](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/logical-operators.html%23operator_not) null AUTO\_INCREMENT PRIMARY key , fechaMensualidad date [not](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/logical-operators.html%23operator_not) null , lecturaActual mediumint [not](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/logical-operators.html%23operator_not) null, valorMensualidad int [not](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/logical-operators.html%23operator_not) null, idCliente int [not](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/logical-operators.html%23operator_not) null, idTipoMensualidad tinyint [not](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/logical-operators.html%23operator_not) null, foreign key (idCliente) references cliente(idCliente), FOREIGN key (idTipoMensualidad) REFERENCES tipomensualidad(idTipoMensualidad))

[insert](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/insert.html) into mensualidad(idMensualidad,fechaMensualidad,lecturaActual,valorMensualidad,idCliente,idTipoMensualidad) [VALUES](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/miscellaneous-functions.html%23function_values) (null,'2010-04-03',100,30000,1,1), (null,'2010-11-05',1000,40000,2,2), (null,'2015-04-03',3000,80000,3,3), (null,'2019-07-09',1900,30000,4,4), (null,'2020-09-08',2300,8000,5,2);

Insertar en cada tablas 5 registros, para tipo de mensualidad (consumo, reconexion, ampliacion, division)

Las siguientes scripts y resultados de consulta guardarlos en archivo Word llamado sqlacueducto.doc y subir la evidencia al LMS

a. Obtener la lista de todos los clientes donde aparezca el nombre en minúsculas y apellido en mayúsculas

[SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/select.html) \* FROM cliente WHERE LOWER(nombreCliente) = nombreCliente [AND](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/logical-operators.html%23operator_and) UPPER(apellidoCliente) = apellidoCliente;

Tabla

Descripción generada automáticamente

b. Obtener el nombre del vendedor y las tres primeras de letras del vendedor de todas las inscripciones

[SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/select.html) nombreVendedor, [LEFT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/string-functions.html%23function_left)(nombreVendedor, 3) FROM inscripcion INNER JOIN vendedor ON inscripcion.idVendedor = vendedor.idVendedor;

Tabla

Descripción generada automáticamente

d. Obtener por cada tipo de mensualidad la cantidad de mensualidades efectuadas

[SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/select.html) tipomensualidad.descripcionMensualidad, [COUNT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/aggregate-functions.html%23function_count)(mensualidad.idMensualidad) FROM mensualidad JOIN tipoMensualidad ON mensualidad.idTipoMensualidad = tipomensualidad.idTipoMensualidad GROUP BY tipomensualidad.descripcionMensualidad;

Patrón de fondo

Descripción generada automáticamente

e. Obtener el cliente y la dirección de la sede donde hizo la inscripción

[SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/select.html) cliente.nombreCliente,cliente.direccionCliente,sede.direccionSede FROM inscripcion INNER JOIN cliente ON inscripcion.idCliente=cliente.idCliente INNER JOIN vendedor ON inscripcion.idVendedor=vendedor.idVendedor INNER JOIN sede on vendedor.idSede=sede.idSede;

Tabla

Descripción generada automáticamente

f. Buscar las lecturas de mensualidades cuyo tipo termine en ‘ion’

[SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/select.html) mensualidad.idMensualidad, mensualidad.fechaMensualidad, mensualidad.lecturaActual, mensualidad.valorMensualidad, mensualidad.idCliente, tipomensualidad.descripcionMensualidad FROM mensualidad INNER JOIN tipomensualidad on mensualidad.idTipoMensualidad=tipomensualidad.idTipoMensualidad WHERE tipomensualidad.descripcionMensualidad [LIKE](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/string-comparison-functions.html%23operator_like) '%ion';

Tabla

Descripción generada automáticamente

**4. ACTIVIDADES DE EVALUACIÓN**

Crear consultar SQL para su proyecto formativo una por cada tabla que tenga subir al LMS en archivo Word las consultas y los resultados de las mismas y la base de datos exportada.

[SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/select.html) cliente.nombreCliente,factura.idFactura,detalleventa.idFacturaDetalle FROM factura INNER JOIN cliente on factura.idCliente=cliente.idCliente INNER JOIN detalleventa ON factura.idFactura= detalleventa.idFacturaDetalle;