

Actividad | #3 | Consultas

Ingeniería en Desarrollo de Software



TUTOR: Miguel Angel Rodriguez Vega

ALUMNO: Edgar Enrique Cuamea Ochoa

FECHA: 10 de septiembre del 2024

Contenido

| | |
|--|----|
| Introducción..... | 3 |
| Descripción..... | 4 |
| Justificación. | 5 |
| Desarrollo. | 6 |
| Creación base de datos..... | 6 |
| Creacion de tablas..... | 6 |
| Llaves foraneas. | 9 |
| Insercion de datos. | 10 |
| Consultas..... | 11 |
| Contar registros en una tabla..... | 11 |
| Duplicados de registros..... | 11 |
| Ordenar de manera alfabetica. | 12 |
| Ordenar de mayor a menor una tabla. | 12 |
| Selección de campos..... | 13 |
| Mostrar registros de año especifico..... | 14 |
| Ver datos de una tabla..... | 14 |
| Conclusion..... | 17 |
| Referencias. | 18 |

Introducción.

En esta actividad usaremos lenguaje escrito para la creación de a base de datos utilizando como base las actividades anteriores ya que conocemos la estructura de la base de datos así como las llaves primarias y foráneas y como es que podemos crear las tablas en PhpMyAdmin por lo que esta vez lo haremos nuevamente pero por comandos, se documentara el comando utilizado y la función que realiza ya que al tener que crear la base de datos de manera local, esto hará que el hacerlo por comandos sea mas fácil pues ya se conoce la estructura de las bases de datos, al tener que hacer las operaciones por comandos también tendremos que tener cuidado con la sintaxis que tengamos ya que un solo error hará que el comando no funcione además de que podemos borrar toda la base de datos por error si no tenemos un control de lo que escribimos por lo que se documentara el proceso de el desarrollo de la actividad.

Descripción.

Para el desarrollo de la actividad tenemos que tener claros los conceptos de los comandos para crear tablas y columnas así como alterarlos y modificarlos así como visualizar las tablas y agregar datos en nuestra base de datos local así como poder modificarlos para ordenarlos de tal manera que nos lo muestre de forma ascendente o de forma descendente por lo que también debemos agregar productos, precios y clientes en nuestra base de datos para documentar el proceso de la creación de lavase de datos que previamente diseñamos de manera grafica. Nos adentraremos en el proceso de la creación de las tablas y columnas a través de los comandos utilizando el lenguaje SQL que nos servirá para poder manipular datos ya sea de manera local o remota así como poder entender más sobre el lenguaje SQL ya que es uno de los muchos lenguajes que se utilizan para la creación de las bases de datos en nuestro entorno.

Justificación.

El proceso para aprender el uso de el lenguaje SQL lo hicimos de manera de entender primero la estructura de las bases de datos así como crear una base de datos local, además de diferenciar las tablas y columnas así como asignándoles una llave primaria a cada tabla y alguna llave foránea para la creación de otras tablas, con esos conceptos que aprendimos sobre las bases de datos podemos crear una base de datos de manera visual así como lo hicimos en la actividad anterior, esto es necesario ya que aprender el lenguaje SQL será más difícil sin conocer el concepto de las bases de datos por lo que ahora veremos como utilizar el lenguaje SQL a través de los comandos para convertir la base de datos que previamente hicimos de distinta forma llegando a los mismos resultados además de comprender el uso de este lenguaje, aprenderemos a agregar datos y como modificarlos para que nos de el resultado que queramos así como documentar el proceso de esta actividad.

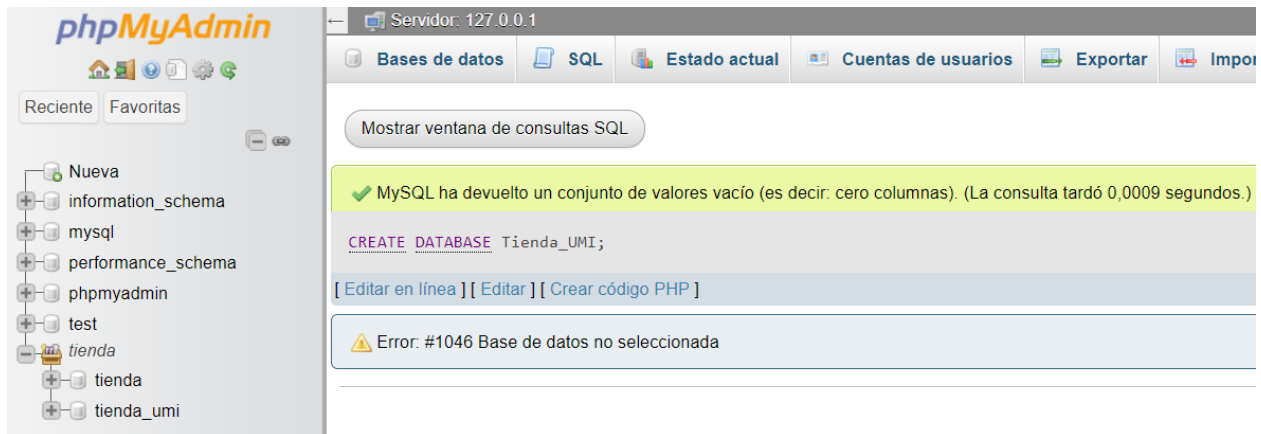
Desarrollo.

Creación base de datos.

Para la creación de la base de datos empezaremos por crear las tablas por lo que utilizaremos el comando (create database) para crear la base de datos donde estaremos trabajando por lo que le asignaremos el nombre de Tienda_UMI



La funcion nos crea una nueva base de datos, no para trabajar en una base de datos limpia por lo que nos crea una base nueva para trabajar



Creacion de tablas.

Una vez creada la base de datos crearemos nuestras tablas con el siguiente comando

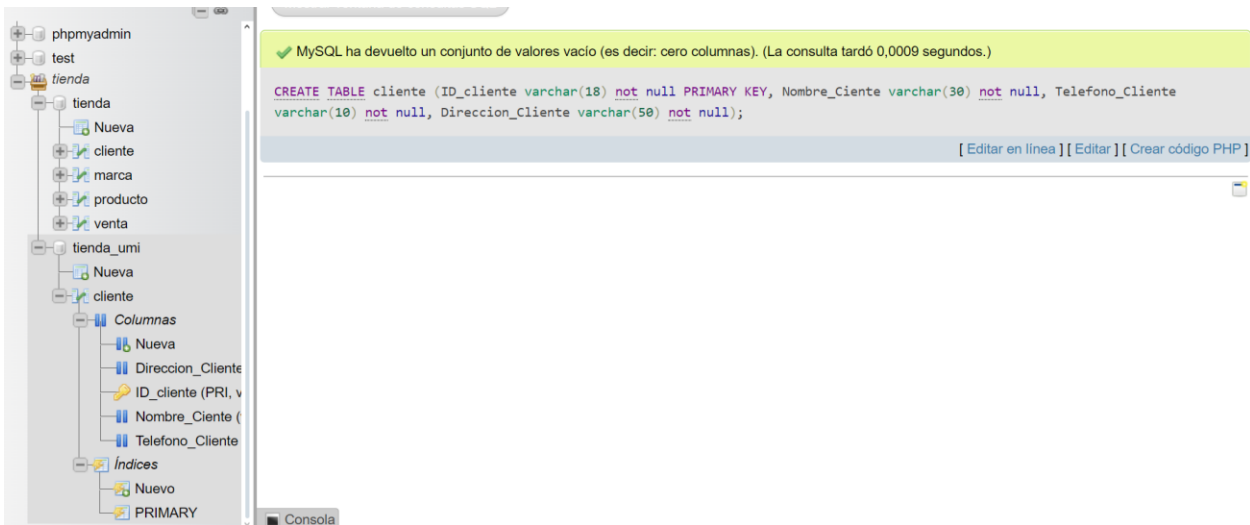
```

1 CREATE TABLE cliente (ID_cliente varchar(18) not null PRIMARY KEY,
2                           Nombre_Ciente varchar(30) not null,
3                           Telefono_Cliente varchar(10) not null,
4                           Direccion_Cliente varchar(50) not null);|

```

Este nos creara las tablas donde estara un numero de identificacion de el cliente, nombre, telefono y direccion del cliente asi como podremos asignarlos como tipo varchar ademas de hacerlo no nullo asignandole de cuantos caracteres puede ser escrita en la tabla, asi como asignandole la llave primaria al id del cliente que nos servira para hacer la tabla de venta.

Asi mismo para las tablas de marca



Así como se muestra en la izquierda podemos hacer muchas tablas con atributos en 4 líneas de código, así que ejecutaremos el siguiente código

```
1 CREATE TABLE Marca (ID_Marca varchar(10) not null PRIMARY KEY,
2                       Marca_Producto varchar(20) not null);
```

este nos creará la tabla de la marca con su llave primaria, por lo que crearemos la tabla del producto con una llave primaria y la llave foránea de la marca

```
1 CREATE TABLE Producto (Codigo_Producto Int(10) not null PRIMARY KEY,
2                          Nombre_Producto varchar(20) not null,
3                          Precio_Producto float not null);
```

una vez teniendo la tabla agregaremos la llave foránea con el siguiente comando;

```
1 ALTER table producto
2 add CONSTRAINT FK_marca
3 FOREIGN Key (ID_Marca) REFERENCES marca(ID_Marca);
```


Esto hara que la columna que tenemos en la tabla de marca como llave primaria, pase a ser foranea en una columna creada en la tabla de producto, y hacemos lo mismo para la tabla venta que llevara la llave foranea de cliente y producto usando los mismos comandos,

```
1 CREATE TABLE venta(Fecha_Venta date not null,  
2                     Numero_Venta int(11) PRIMARY key AUTO_INCREMENT,  
3                     ID_Cliente varchar(18) not null,  
4                     Codigo_Producto int(10) not null);|
```

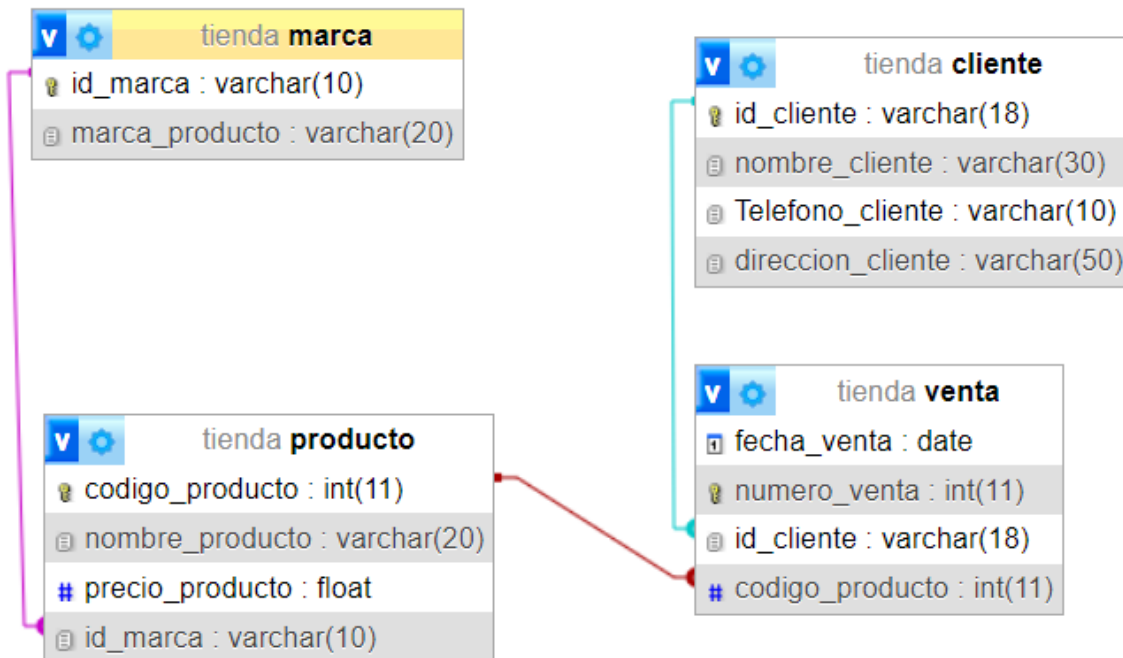
una vez creada la tabla utilizaremos los comandos siguientes para agregar nuestras llaves foraneas;

Llaves foraneas.

```
1 alter table venta  
2 add CONSTRAINT FK_Producto  
3 FOREIGN key (Codigo_Producto) references producto(Codigo_Producto);|
```

```
1 ALTER TABLE venta  
2 add CONSTRAINT FK_Cliente  
3 FOREIGN key (ID_Cliente) REFERENCES cliente(ID_Cliente);
```

Por lo que terminamos teniendo las mismas tablas de las actividades pasadas hechas con codigo



Insercion de datos.

Ahora empezaremos a agregar datos en nuestra base de datos por lo que haremos lo siguiente donde el comando insert into funciona para agregar datos a la tabla que elijamos así como los datos correspondientes que debemos agregar por lo que agregaremos diferentes clientes, productos y ventas utilizando la misma función, cambiando las tablas así como los datos a agregar

```

1 INSERT into producto (Codigo_Producto, Nombre_Producto, Precio_Producto)
2 VALUES(1, 'Laptop gamer', 12999),
3 (2, 'Telefono gamer', 15999),
4 (3, 'Consola ps6', 19999),
5 (4, 'novela madoka', 2499),
6 (5, 'Rtx5090ti', 49999);
  
```

Consultas.

Contar registros en una tabla.

Con el siguiente código podemos contar los registros existentes en una tabla;

Su consulta se ejecutó con éxito.

```
SELECT COUNT(*) FROM cliente;
```

☐ Perfilando [[Editar en línea](#)] [[Editar](#)] [[Explicar SQL](#)] [[Crear código PHP](#)] [[Actualizar](#)]

Opciones extra

COUNT(*)
5

Operaciones sobre los resultados de la consulta


Imprimir Copiar al portapapeles Exportar Mostrar gráfico Crear vista

El resultado que nos da es de 5 ya que ingrese 5 clientes distintos en la tabla de clientes, por lo cual vamos a definir si hay registros duplicados o no con el siguiente comando,

Duplicados de registros.

```
1 select Nombre_Cliente, ID_Cliente from cliente
2 GROUP by Nombre_Cliente, ID_Cliente
3 HAVING COUNT(*)>1|
```

Ya que seleccionamos el nombre y el id del cliente desde la tabla de cliente y los agrupamos por nombre y por id, despues contamos los grupos, si no hay duplicados el codigo nos devolvera un valor de cero, si hay algun duplicado, devolvera uno o mas registros.


 MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0004 segundos.)

```
select Nombre_Cliente, ID_Cliente from cliente GROUP by Nombre_Cliente, ID_Cliente HAVING COUNT(*)>1;
```

☐ Perfilando [[Editar en línea](#)] [[Editar](#)] [[Explicar SQL](#)] [[Crear código PHP](#)] [[Actualizar](#)]

Nombre_Cliente **ID_Cliente**

Operaciones sobre los resultados de la consulta



 [Crear vista](#)

Ordenar de manera alfabetica.

Seleccionamos desde la tabla de cliente la columna de Nombre_Cliente y lo ordenamos de manera ascendente con el comando de (asc), o podemos ordenar de manera descendente con el comando de (desc)

```
1 SELECT * FROM cliente
2 order by Nombre_Cliente asc
```

Nos devuelve la siguiente tabla donde estan ordenados de manera alfabetica desde la A a la Z en la tabla cliente

| | | ID_cliente | Nombre_Cliente | Telefono_Cliente | Direccion_Cliente |
|--------------------------|--|--------------|---------------------|------------------|-----------------------|
| <input type="checkbox"/> |  Editar  Copiar  Borrar | KANAMEMADOKA | Angel Mares | 6621234567 | avenida inventada 456 |
| <input type="checkbox"/> |  Editar  Copiar  Borrar | SAKURAKYOKO | Isai Bazan | 6671234567 | plaza ilusion 789 |
| <input type="checkbox"/> |  Editar  Copiar  Borrar | SAYAKAMIKI | Juan Carlos Peinado | 6651234567 | camino irreal 101 |
| <input type="checkbox"/> |  Editar  Copiar  Borrar | TOMOEMAMI | Martha Blas | 6312345678 | colonia falsa 123 |
| <input type="checkbox"/> |  Editar  Copiar  Borrar | HOMURAAKEMI | Victoria Sotelo | 6311234567 | calle falsa123 |

Ordenar de mayor a menor una tabla.

Con el siguiente código ordenaremos de mayor a menor la tabla de precio de los productos

```

1 SELECT * FROM producto
2 ORDER BY Precio_Producto DESC;

```

Dando como resultado la siguiente tabla donde esta el producto mas caro arriba y el mas barato abajo sin ordenarlo por el codigo con el numero mas alto

| | | | Nombre_Producto | Codigo_Producto | Precio_Producto | 1 | ID_Marca |
|--------------------------|--|--|-----------------|-----------------|-----------------|-------|----------|
| <input type="checkbox"/> | | | | Rtx5090ti | 1678901234 | 49999 | NULL |
| <input type="checkbox"/> | | | | Consola ps6 | 1134567813 | 19999 | NULL |
| <input type="checkbox"/> | | | | Telefono gamer | 1357924567 | 15999 | NULL |
| <input type="checkbox"/> | | | | Laptop gamer | 1234567890 | 12999 | NULL |
| <input type="checkbox"/> | | | | novela madoka | 1235689012 | 2499 | NULL |

Selección de campos.

Con el siguiente comando seleccionaremos los campos de ID_Cliente y de Codigo_Producto,

```

1 select ID_Cliente, Codigo_Producto
2 FROM venta;

```

Por lo que da como resultado esta tabla

| | | | | ID_Cliente | Codigo_Producto |
|--------------------------|--|--|--|-------------|-----------------|
| <input type="checkbox"/> | | | | HOMURAAKEMI | 1235689012 |
| <input type="checkbox"/> | | | | HOMURAAKEMI | 1357924567 |
| <input type="checkbox"/> | | | | SAYAKAMIKI | 1134567813 |

Aquí se pueden ver las ventas que tenemos, así como los clientes que nos compraron y los










codigos del producto que fueron vendidos.

Mostrar registros de año específico.

Con el siguiente código podemos filtrar los datos para que nos de el resultado que queramos, en este caso seleccionamos la fecha de la venta donde queremos que nos muestre las ventas del año 2024 por lo que escribimos el siguiente código;

```
1 select Fecha_Venta
2 from venta
3 where year (fecha_Venta)=2024
```

Aquí se muestra que seleccionamos la fecha de venta desde la tabla de venta donde definimos que el año que nos muestre sea igual al del 2024 por lo que los resultados que muestran son los siguientes;

| | | | | Fecha_Venta |
|--------------------------|--|--|--|-------------|
| <input type="checkbox"/> |  Editar |  Copiar |  Borrar | 2024-09-09 |
| <input type="checkbox"/> |  Editar |  Copiar |  Borrar | 2024-09-09 |
| <input type="checkbox"/> |  Editar |  Copiar |  Borrar | 2024-08-09 |

No tenemos clientes ni productos, solo las fechas de la venta que se realizaron en el año 2024.

Ver datos de una tabla.

Como vimos anteriormente podemos ver los datos de una tabla con la función select from, ya que esta nos permite seleccionar una columna de la tabla que queramos por lo que si queremos visualizar los datos solo debemos escribir el siguiente código;

```
1 SELECT * from cliente;
```

En este caso para vizualizar los clientes de la siguiente manera

✓ Mostrando filas 0 - 4 (total de 5, La consulta tardó 0,0004 segundos.)

SELECT * from cliente;

☐ Perfilando [[Editar en línea](#)] [[Editar](#)] [[Explicar SQL](#)] [[Crear código PHP](#)] [[Actualizar](#)]

☐ Mostrar todo | Número de filas: 25 | Filtrar filas: Ordenar según la clave: Ninguna

Opciones extra

| | | | | ID_cliente | Nombre_Cliente | Telefono_Cliente | Direccion_Cliente |
|--------------------------|--|--|--|--------------|---------------------|------------------|-----------------------|
| <input type="checkbox"/> | | | | HOMURAAKEMI | Victoria Sotelo | 6311234567 | calle falsa123 |
| <input type="checkbox"/> | | | | KANAMEMADOKA | Angel Mares | 6621234567 | avenida inventada 456 |
| <input type="checkbox"/> | | | | SAKURAKYOKO | Isai Bazan | 6671234567 | plaza ilusion 789 |
| <input type="checkbox"/> | | | | SAYAKAMIKI | Juan Carlos Peinado | 6651234567 | camino irreal 101 |
| <input type="checkbox"/> | | | | TOMOEMAMI | Martha Blas | 6312345678 | colonia falsa 123 |

Aquí nos muestra los clientes que tenemos registrados, no están de manera alfabética, están ordenados de manera en la que los fuimos agregando a la base de datos, Así como la tabla de producto;

✓ Mostrando filas 0 - 4 (total de 5, La consulta tardó 0,0025 segundos.)

`SELECT * from producto;`

☐ Perfilando [[Editar en línea](#)] [[Editar](#)] [[Explicar SQL](#)] [[Crear código PHP](#)] [[Actualizar](#)]

☐ Mostrar todo | Restaurar orden de las columnas | Número de filas: 25 ▾ Filtrar filas:

Opciones extra

| | | | | | Nombre_Producto | Codigo_Producto | Precio_Producto | ID_Marca |
|--------------------------|--|--|--|--|-----------------|-----------------|-----------------|----------|
| <input type="checkbox"/> |  Editar |  Copiar |  Borrar | | Consola ps6 | 1134567813 | 19999 | NULL |
| <input type="checkbox"/> |  Editar |  Copiar |  Borrar | | Laptop gamer | 1234567890 | 12999 | NULL |
| <input type="checkbox"/> |  Editar |  Copiar |  Borrar | | novela madoka | 1235689012 | 2499 | NULL |
| <input type="checkbox"/> |  Editar |  Copiar |  Borrar | | Telefono gamer | 1357924567 | 15999 | NULL |
| <input type="checkbox"/> |  Editar |  Copiar |  Borrar | | Rtx5090ti | 1678901234 | 49999 | NULL |

Conclusion.

En resumen, parece sencillo la creación de las bases de datos a través del lenguaje SQL donde tenemos que tener bien en claro que es lo que queremos hacer pues a partir de esta idea tenemos que trabajar en el código para crear las tablas así como las llaves primarias y llaves foráneas, es importante tener escrita bien la sintaxis, pues al no importar tanto si esta escrito en mayúsculas o minúsculas las funciones, es importante tener bien clara la forma en que se escribe la tabla o la columna que queremos seleccionar ya que esta mala sintaxis al escribir el código puede hacer que no nos muestre el resultado que queramos o incluso no funciona el código por lo que es importante tener esto claro, además de que es sencillo familiarizarse con algunos comandos de consulta como el select from, ya que es sencillo de utilizar y de aprender, así como el insert into, para agregar datos a una tabla, ya que de esta forma estaremos agregando datos a nuestra base de datos donde podemos hacerlo de manera automática con una aplicación de venta sin tener que introducir todos los datos de manera manual haciendo el trabajo de agregación mas rápido y fácil.

Referencias.

Hackemi. (s. f.-b). *GitHub - Hackemi/Actividades_base_de_datos*. GitHub.
[https://github.com/Hackemi/Actividades base de datos](https://github.com/Hackemi/Actividades_base_de_datos)

XAMPP Installers and Downloads for Apache Friends. (s. f.). <https://www.apachefriends.org/>