1 Cree una tabla autos y otra tabla marcas y relaciónelos.

En la figura # 1 se muestra la tabla marcas, la cual tiene dos columnas. La primera es un idMarca de identificación y la segunda es el nombre de la marca que es única en este campo, lo cual quiere decir que no se va a repetir el nombre de marca.

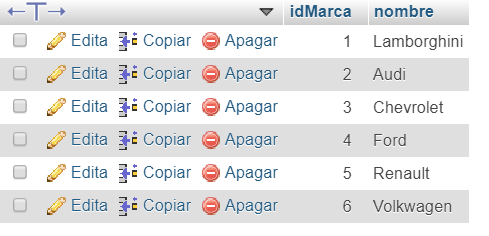


Figura # 1

En la figura # 2 se muestra la tabla autos que tiene cuatro columnas, un id único para cada carro, un nombre que referencia que carro , una columna precio que muestra el valor de cada carro y una columna idMarca que es la utilizada para saber la marca del carro.



Figura# 2

La relación se muestra en la figura # 3.

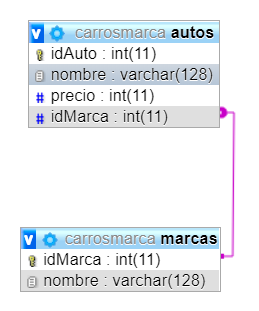


Figura # 3

1.1 Consultar todos los autos que contengan la cadena GT.

SELECT nombre FROM autos WHERE nombre LIKE "%gt%";

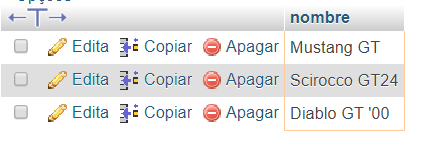


Figura # 4

1.2 Consultar todos los autos cuya marca finalice con la letra t.

SELECT au.nombre,au.precio,ma.nombre FROM autos AS au INNER JOIN marcas AS ma ON au.idMarca = ma.idMarca WHERE ma.nombre LIKE "%t";



Figura # 5

1.3 consultar la referencia del auto, precio y nombre de la marca de los autos cuyo precio esté en un rango delimitado entre 100 y 550;

SELECT au.nombre,au.precio,ma.nombre FROM autos AS au INNER JOIN marcas AS ma ON au.idMarca = ma.idMarca WHERE au.precio BETWEEN 100 AND 550 ORDER BY au.nombre ASC;

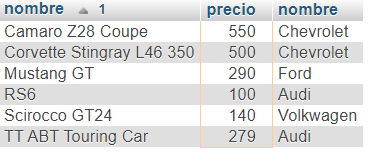


Figura # 6

1.4 Consultar referencia del auto y marca de los autos cuyas marcas sean iguales a CHEVROLET y FORD, organizarlos alfabéticamente de acuerdo a la referencia del auto.

SELECT au.nombre,au.precio,ma.nombre FROM autos AS au INNER JOIN marcas AS ma ON au.idMarca = ma.idMarca WHERE ma.nombre LIKE "%chevrolet%" or ma.nombre LIKE "%ford%";

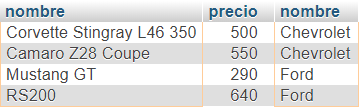


Figura # 7

2 base de datos Facebook.

El siguiente código muestra los datos básicos de todos los usuarios de nuestra base de datos.

SELECT ing.idIngresos,ing.nombre AS Nickname,ing.password,dat.nombre,dat.apellido,dat.nacimiento,dat.estado,dat.correo FROM usuarios AS usu INNER JOIN datos AS dat on usu.datos = dat.idDatos INNER JOIN ingresos AS ing ON usu.idUsuarios=ing.idIngresos;

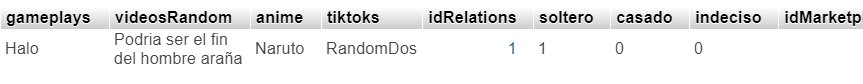


Datos generales de cada uno de nuestros usuarios.

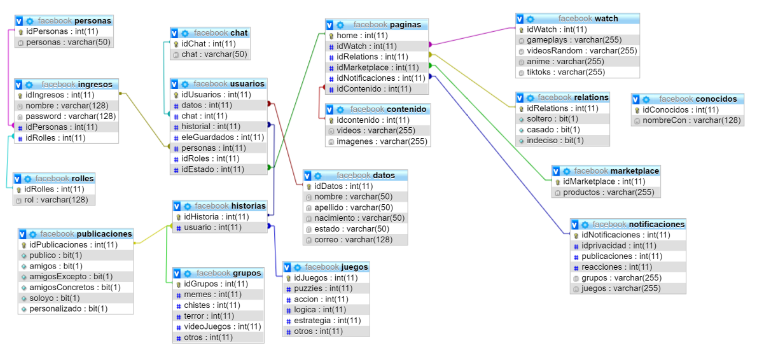
Para poder ingresar a los datos de actividades de cada usuario se puede usar el siguiente código.

SELECT \* FROM paginas AS pag INNER JOIN watch AS wat ON pag.idWatch = wat.idWatch INNER JOIN relations AS rel ON pag.idRelations = rel.idRelations INNER JOIN marketplace AS mark ON pag.idMarketplace = mark.idMarketplace INNER JOIN notificaciones AS noti ON pag.idNotificaciones = noti.idNotificaciones INNER JOIN contenido AS cont ON pag.idContenido = cont.idcontenido WHERE pag.home = 1;

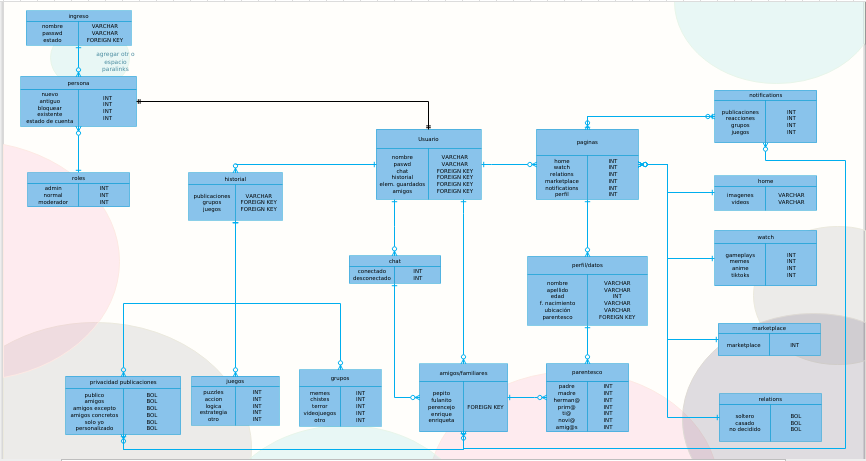




Esta es la información perteneciente al primer usuario.



la explicación de la base de datos se muestra en el siguiente diagrama, en donde se evidencia la organización que se tomo como referencia para realizar el segundo punto del taller.

Diagrama

Este consiste en que primero el usuario ingresa a la plataforma, luego se valida si es nuevo, antiguo, si el usuario esta bloqueado o ya existe, con un estado de cuenta, luego este pasa a roles en donde se guarda si es moderador , administrador o usuario normal, una vez ingresado y verificado el usuario tiene sus datos registrados, su nombre, contraseña, el chat que tiene, su historial en donde se ven las publicaciones que ha tenido, etc.

3) Importe el archivo adjunto llamado copas.sql al entorno MYSQL Workbench o phpmyadmin y realizar las siguientes consultas.

3.1 Consultar todos los equipos que pertenecen al grupo B.

SELECT sel.nombre, sel.grupo FROM seleccion AS sel INNER JOIN grupo AS gru on sel.grupo = gru.id\_grupo where gru.grupo = "B";



Figura # 8

3.2 Consultar todos los partidos del grupo A.

SELECT sel.nombre,sel2.nombre FROM partido AS par INNER JOIN selección AS sel ON par.id\_seleccion = sel.id\_seleccion INNER JOIN selección AS sel2 ON sel2.id\_seleccion = par.id\_seleccion\_dos INNER JOIN grupo AS gru ON sel.grupo = gru.id\_grupo WHERE gru.grupo = “A” ORDER BY par.id\_partido ASC;



Figura # 9

3.3 Consulte todos los partidos de los equipos que contengan la letra u en su nombre.

SELECT par.id\_partido,sel.nombre, sel2.nombre FROM partido AS par INNER JOIN seleccion AS sel ON par.id\_seleccion = sel.id\_seleccion INNER JOIN seleccion AS sel2 ON sel2.id\_seleccion = par.id\_seleccion\_dos INNER JOIN grupo AS gru ON sel.grupo = gru.id\_grupo WHERE sel.nombre LIKE "%u%" or sel2.nombre LIKE "%u%";



Figura # 10

3.4 Consultar los equipos que ganaron de visitantes y el marcador del partido indicando la consulta de cualquiera de las dos formas que usted desee.

SELECT par.id\_partido as partido, sel2.nombre as visitante,res.goles\_equipo\_uno as Marvisitantes,res.goles\_equipo\_dos as MarLocales,sel.nombre as local FROM resultado AS res INNER JOIN partido AS par ON res.id\_partido = par.id\_partido INNER JOIN seleccion AS sel ON sel.id\_seleccion = par.id\_seleccion INNER JOIN seleccion AS sel2 ON sel2.id\_seleccion = par.id\_seleccion\_dos INNER JOIN grupo AS gro ON gro.id\_grupo = sel.grupo where res.goles\_equipo\_uno > res.goles\_equipo\_dos ORDER BY par.id\_partido asc;



Figura # 11

3.5 Consulta los partidos y resultados de la selección Colombia.

SELECT par.id\_partido as partido, sel2.nombre as visitante,res.goles\_equipo\_uno as Marvisitantes,res.goles\_equipo\_dos as MarLocales,sel.nombre as local FROM resultado AS res INNER JOIN partido AS par ON res.id\_partido = par.id\_partido INNER JOIN seleccion AS sel ON sel.id\_seleccion = par.id\_seleccion INNER JOIN seleccion AS sel2 ON sel2.id\_seleccion = par.id\_seleccion\_dos INNER JOIN grupo AS gro ON gro.id\_grupo = sel.grupo where sel.id\_seleccion = "1" or sel2.id\_seleccion = "1" ORDER BY par.id\_partido asc;



Figura 12