

CREACION DE UNA BASE DE DATOS “Empresa de alquiler de vehículos”

Para crear nuestra base de datos, lo primero es recopilar toda la información necesaria para cubrir las necesidades de nuestro cliente. En este caso se trata de una empresa de alquiler de vehículos, que nos facilitan los datos que desea organizar y dichos datos son:

Cientes y que cuya información quiere recoger en: id del cliente, nombre, apellidos, DNI/pasaporte, teléfono, sexo, fecha de nacimiento y país de nacimiento.

Vehículos que recoge: id del vehículo, número de matrícula, categoría, modelo y año ITV.

Reservas de vehículos y recoger: id de la reserva, trayecto, fecha del alquiler, hora de inicio del alquiler, hora del fin del alquiler, vehículo que se reserva y nombre del cliente.

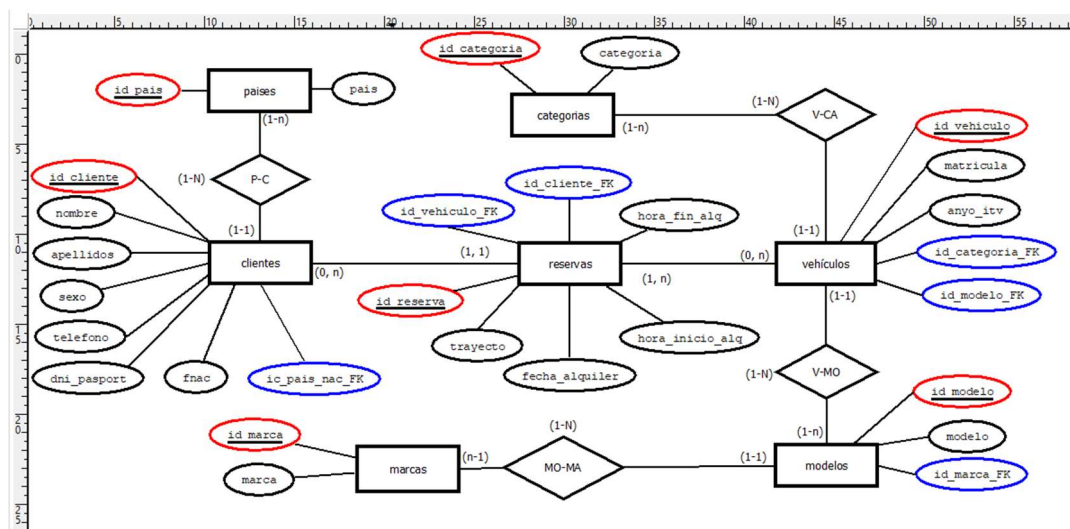
Es importante, saber que la empresa de alquiler de vehículos le interesa tener los datos de las categorías, los modelos y las marcas de los vehículos registrados.

Las categorías vienen definidas por: id de categorías, nombre de la categoría.

Los modelos vienen definidos por: id, nombre y marca de vehículo.

MODELO ENTIDAD RELACION

Primeramente, con la información facilitada creamos un diagrama entidad relación, cuyas entidades son: Países, clientes, categorías, marcas, modelos, vehículos y reservas, que las relacionamos tal y como se muestra en la siguiente imagen:



Las relaciones de las entidades las hemos realizado teniendo en cuenta la participación de una entidad en otra, esto se le llama cardinalidad y para obtener esta cardinalidad (ejem.(1-n)) se debe fijar una ocurrencia concreta de una entidad y averiguar cuántas ocurrencias de la otra entidad le corresponden como mínimo y como máximo, y después se realiza lo mismo en el otro sentido, ejemplo de cómo realizar la relación de entidad-relación de las entidades “países” y “clientes” es:

¿Un país puede tener 0 ó 1 cliente?

¿Un país puede tener 1 ó más cliente?

¿Un cliente puede tener 0 ó 1 país?

¿Un cliente puede tener 1 ó más países?

Respondiendo a dichas preguntas, deducimos que el modelo relacional es: (1-N), donde el 1 es el valor máximo obtenido y la N es más de 1.

GRAFO RELACIONAL.

La normalización es un proceso que consiste en aplicar una serie de reglas que convertir un modelo entidad-relación a un modelo relacional. De esta manera obtenemos los atributos de las entidades y evitando la repetición de atributos innecesariamente, esto lo representamos de la siguiente manera:

Países (**id_pais** INT AUT (PK), pais VARCHAR (30) NN UQ)

Cientes (**id_clientes** INT AUT (PK), nombre VARCHAR (30) NN, apellidos VARCHAR (60), dni_pasport VARCHAR (12) UQ NN, teléfono VARCHAR (12) NN, sexo ENUM (Mujer, Hombre, N/C) , fnac DATE, **id_pais_nac_FK** INT)

Categorías (**id_categoria** INT AUT (PK), categoria VARCHAR (20) NN UQ)

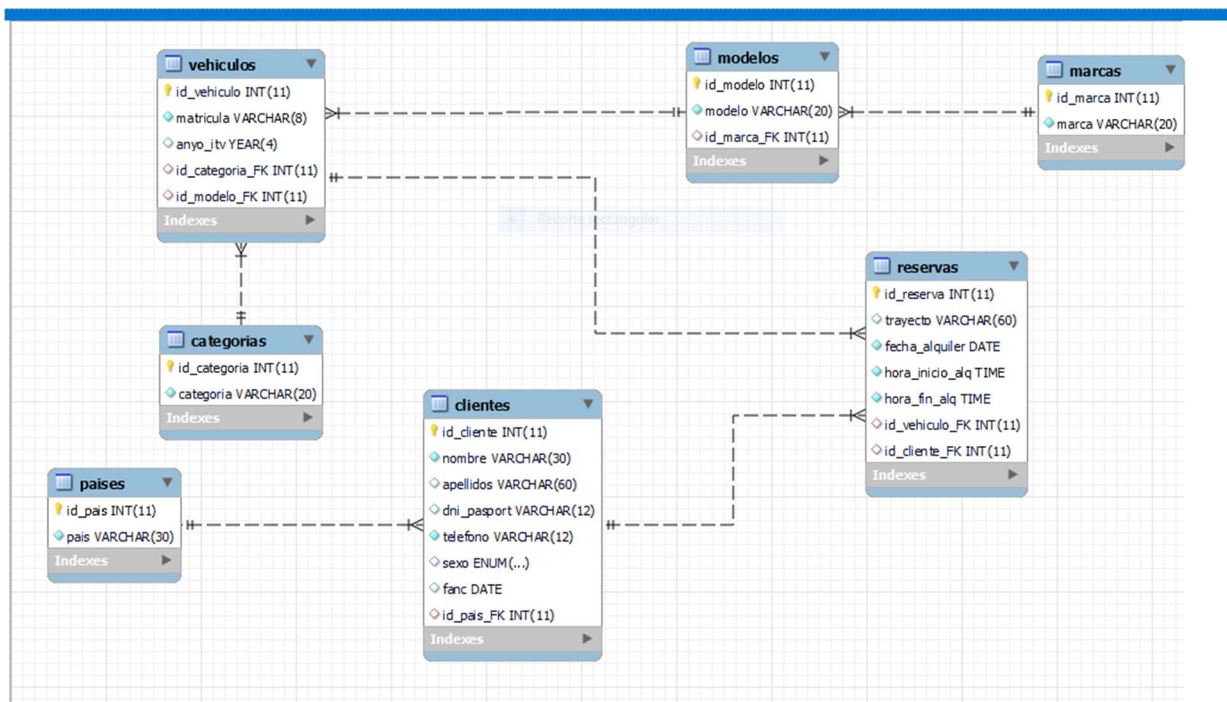
Marcas (**id_marca** INT AUT (PK), marca VARCHAR (20) NN UQ)

Modelos (**id_modelo** INT AUT (PK), modelo VARCHAR (20) NN UQ, **id_marca_FK** INT)

Vehículos (**id_vehiculo** INT AUT (PK), matricula VARCHAR (8) UQ NN, anyo_itv YEAR, **id_categoria_FK** INT, **id_modelo_FK** INT)

Reservas (**id_reserva** INT AUT (PK), trayecto VARCHAR (60), fecha_alquiler DATE NN, hora_inicio_alq TIME NN, hora_fin_alq TIME NN, **id_vehiculo_FK**, **id_cliente_FK**)

VISTA DE DISEÑO.

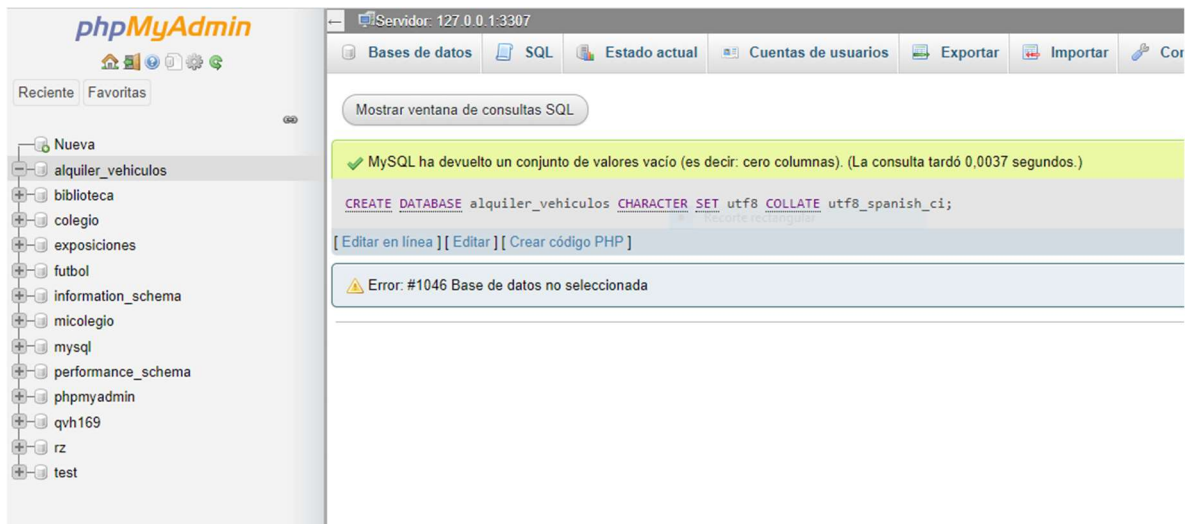


CREAMOS LA BASE DE DATOS.

Para crear la base de datos utilizaremos como gestor de base de datos “phpMyAdmin”, una vez dentro vamos, damos click a la pestaña “SQL” que se encuentra en la parte superior de la pantalla, tras seleccionar “SQL” nos aparece un cuadro en blanco para introducir el siguiente código:

```
CREATE DATABASE alquiler_vehiculos CHARACTER SET utf8 COLLATE utf8_spanish_ci;
```

Le damos a botón de continuar y nos aparece la siguiente pantalla, donde nos indica que la base de datos se ha creado sin ningún problema.



CREAMOS LAS TABLAS.

Una vez creada la base de datos procedemos a crear las tablas.

En primer lugar seleccionamos la base de datos “alquiler_vehiculo” y clickeamos la pestaña “SQL” e introducimos la siguiente secuencia de código:

```
CREATE TABLE paises (id_pais INT NOT NULL AUTO_INCREMENT PRIMARY KEY,  
                      pais VARCHAR (30) NOT NULL UNIQUE  
                      );
```

```
CREATE TABLE clientes (id_cliente INT NOT NULL AUTO_INCREMENT PRIMARY KEY,  
                        nombre VARCHAR(30) NOT NULL, apellidos VARCHAR(60),  
                        dni_pasport VARCHAR(12) UNIQUE,  
                        telefono VARCHAR(12) NOT NULL,  
                        sexo ENUM ("Mujer","Hombre","N/C"),  
                        fnac DATE,  
                        id_pais_FK INT,
```

```

        FOREIGN KEY (id_pais_FK) REFERENCES paises (id_pais)

    );

CREATE TABLE categorías (id_categoria INT NOT NULL AUTO_INCREMENT PRIMARY KEY,
    categoria VARCHAR(20) NOT NULL UNIQUE

);

CREATE TABLE marcas (id_marca INT NOT NULL AUTO_INCREMENT PRIMARY KEY,
    marca VARCHAR(20) NOT NULL UNIQUE

);

CREATE TABLE modelos (id_modelo INT NOT NULL AUTO_INCREMENT PRIMARY KEY,
    modelo VARCHAR(20) NOT NULL UNIQUE,
    id_marca_FK INT,
    FOREIGN KEY (id_marca_FK) REFERENCES marcas (id_marca)

);

CREATE TABLE vehículos (id_vehiculo INT NOT NULL AUTO_INCREMENT PRIMARY KEY,
    matricula VARCHAR(8) NOT NULL UNIQUE,
    anyo_itv YEAR,
    id_categoria_FK INT,
    FOREIGN KEY (id_categoria_FK) REFERENCES categorias (id_categoria),
    id_modelo_FK INT,
    FOREIGN KEY (id_modelo_FK) REFERENCES modelos (id_modelo)

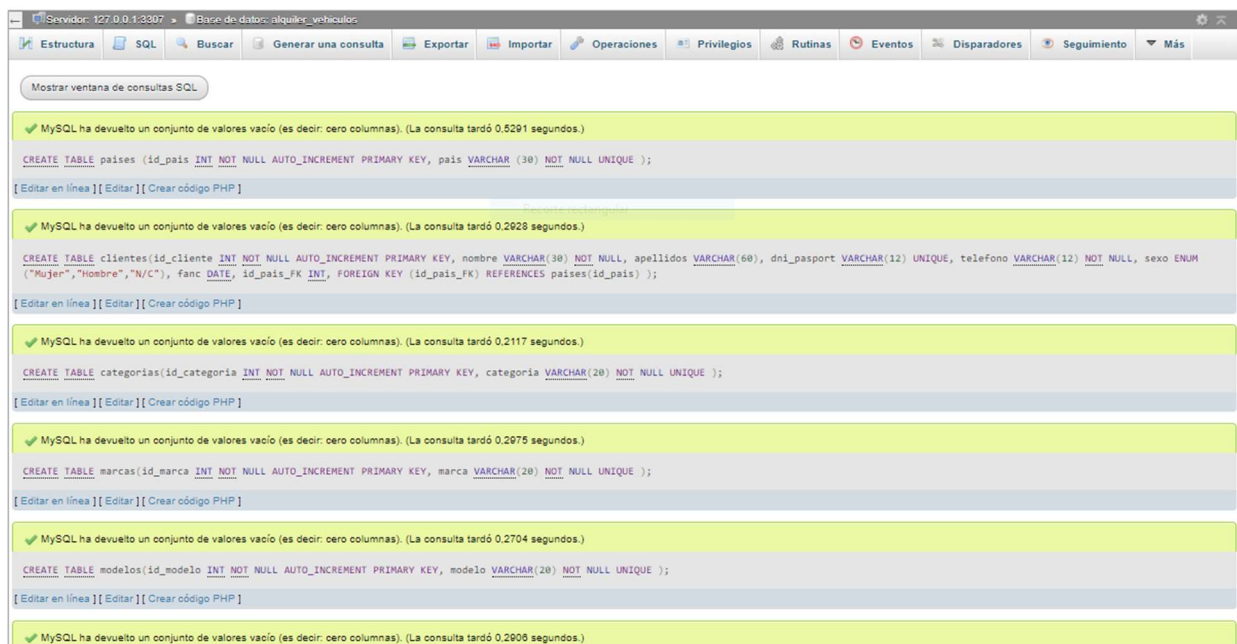
);

CREATE TABLE reservas (id_reserva INT NOT NULL AUTO_INCREMENT PRIMARY KEY,
    trayecto VARCHAR (60),
    fecha_alquiler DATE NOT NULL,
    hora_inicio_alq TIME NOT NULL,
    hora_fin_alq TIME NOT NULL,
    id_vehiculo_FK INT,
    FOREIGN KEY (id_vehiculo_FK) REFERENCES vehiculos (id_vehiculo),
    id_cliente_FK INT,
    FOREIGN KEY (id_cliente_FK) REFERENCES clientes (id_cliente)

);

```

Una vez introducido nuestro código, damos a continuar y no sale la siguiente pantalla donde nos dice que se ha creado todas las tablas exitosamente.



INSERCIÓN DE LOS DATOS EN LAS TABLAS

Mediante la consulta INSERT INTO insertamos los datos de cada tabla, con la siguiente sintaxis:

INSERT INTO nombreTabla (nombreAtributo1, nombreAtributo2,...) VALUES ("valor1", "valor2",...);

El código para la inserción de datos en la tabla países es :

INSERT INTO países (pais) VALUES ("España");

INSERT INTO países (pais) VALUES ("Portugal");

INSERT INTO países (pais) VALUES ("Francia");

INSERT INTO países (pais) VALUES ("Alemania");

INSERT INTO países (pais) VALUES ("Rumania");

INSERT INTO países (pais) VALUES ("Marruecos");

INSERT INTO países (pais) VALUES ("Italia");

Una vez introducido el código le damos a continuar y nos aparece en pantalla que está todo correcto.

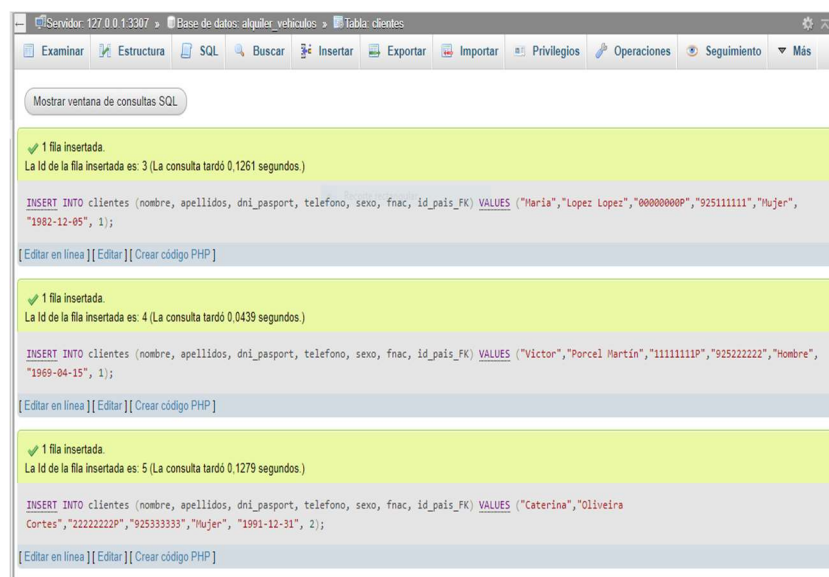


Procedemos igualmente con la tabla clientes a introducir los datos:

```
INSERT INTO clientes (nombre, apellidos, dni_pasport, telefono, sexo, fnac, id_pais_FK)
VALUES ("Maria","Lopez Lopez","00000000P","925111111","Mujer", "1982-12-05", 1);
INSERT INTO clientes (nombre, apellidos, dni_pasport, telefono, sexo, fnac, id_pais_FK)
VALUES ("Victor","Porcel Martín","11111111P","925222222","Hombre", "1969-04-15", 1);
INSERT INTO clientes (nombre, apellidos, dni_pasport, telefono, sexo, fnac, id_pais_FK)
VALUES ("Caterina","Oliveira Cortes","22222222P","925333333","Mujer", "1991-12-31", 2);
INSERT INTO clientes (nombre, apellidos, dni_pasport, telefono, sexo, fnac, id_pais_FK)
VALUES ("Danilo","Pereira Vidal","33333333P","925444444","Hombre", "1998-07-11", 2);
INSERT INTO clientes (nombre, apellidos, dni_pasport, telefono, sexo, fnac, id_pais_FK)
VALUES ("Camille","Lombrad Briand","44444444P","925555555","Mujer", "2002-08-25", 3);
INSERT INTO clientes (nombre, apellidos, dni_pasport, telefono, sexo, fnac, id_pais_FK)
VALUES ("Charles","Signoret Regnault","55555555P","925666666","Hombre", "1970-03-23", 3);
INSERT INTO clientes (nombre, apellidos, dni_pasport, telefono, sexo, fnac, id_pais_FK)
VALUES ("Giselle","Holbin","66666666P","925777777","Mujer", "1995-06-11", 4);
INSERT INTO clientes (nombre, apellidos, dni_pasport, telefono, sexo, fnac, id_pais_FK)
VALUES ("Morgan","Müller","77777777P","925888888","Hombre", "1988-11-15", 4);
INSERT INTO clientes (nombre, apellidos, dni_pasport, telefono, sexo, fnac, id_pais_FK)
VALUES ("Marius","Popa","88888888P","925999999","Hombre", "2001-05-30", 5);
INSERT INTO clientes (nombre, apellidos, dni_pasport, telefono, sexo, fnac, id_pais_FK)
```

```
VALUES ("Alexandru","Radu","999999999P","925000000","Hombre", "2000-01-28", 5);
INSERT INTO clientes (nombre, apellidos, dni_pasport, telefono, sexo, fnac, id_pais_FK)
VALUES ("Mohamed","Alami","12222222P","925121212","Hombre", "1987-11-30", 6);
INSERT INTO clientes (nombre, apellidos, dni_pasport, telefono, sexo, fnac, id_pais_FK)
VALUES ("Said","Alaovi","13333333P","925131313","Hombre", "1985-12-25", 6);
INSERT INTO clientes (nombre, apellidos, dni_pasport, telefono, sexo, fnac, id_pais_FK)
VALUES ("Bruno","Rossi Ferrari","14444444P","925141414","Hombre", "1997-10-25", 7);
INSERT INTO clientes (nombre, apellidos, dni_pasport, telefono, sexo, fnac, id_pais_FK)
VALUES ("Franco","Cabano Fabene","15555555P","925151515","Hombre", "1999-10-20", 7);
INSERT INTO clientes (nombre, apellidos, dni_pasport, telefono, sexo, fnac, id_pais_FK)
VALUES ("Manuel","Garcia Rodriguez","16666666P","925161616","Hombre", "1993-11-15", 1);
```

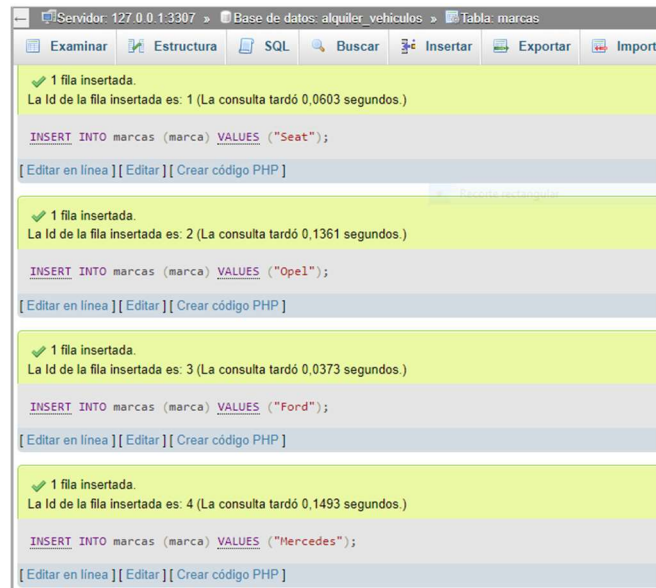
Una vez introducido todos los datos damos a continuar y nos sale que está todo correcto.



Este es el código de la inserción de los datos de la tabla marcas:

```
INSERT INTO marcas (marca) VALUES ("Seat");
INSERT INTO marcas (marca) VALUES ("Opel");
INSERT INTO marcas (marca) VALUES ("Ford");
INSERT INTO marcas (marca) VALUES ("Mercedes");
INSERT INTO marcas (marca) VALUES ("BMW");
INSERT INTO marcas (marca) VALUES ("Renault");
INSERT INTO marcas (marca) VALUES ("Peugeot");
INSERT INTO marcas (marca) VALUES ("KIA");
```


Confirmación tras dar a continuar:



Código de inserción de datos de la tabla modelos:

```
INSERT INTO modelos (modelo, id_marca_FK) VALUES ("Captur", 6);  
INSERT INTO modelos (modelo, id_marca_FK) VALUES ("Corsa", 2);  
INSERT INTO modelos (modelo, id_marca_FK) VALUES ("Fiesta", 3);  
INSERT INTO modelos (modelo, id_marca_FK) VALUES ("Rio", 8);.....
```

Confirmación tras dar a continuar:



Código de inserción de datos de la tabla categorias:


```

INSERT INTO categorias (categoria) VALUES ("Subcompacto");
INSERT INTO categorias (categoria) VALUES ("Compacto");
INSERT INTO categorias (categoria) VALUES ("Familiar");
INSERT INTO categorias (categoria) VALUES ("Berlina");
INSERT INTO categorias (categoria) VALUES ("Coupé");
INSERT INTO categorias (categoria) VALUES ("Monovolumen");
INSERT INTO categorias (categoria) VALUES ("Vehículos comerciales");

```

Confirmación tras dar a continuar:



Código de inserción de datos de la tabla vehiculos:

```

INSERT INTO vehiculos (matricula, anyo_itv, id_categoria_FK, id_modelo_FK)
VALUES ("1111 AAA", "2019", 1, 1);

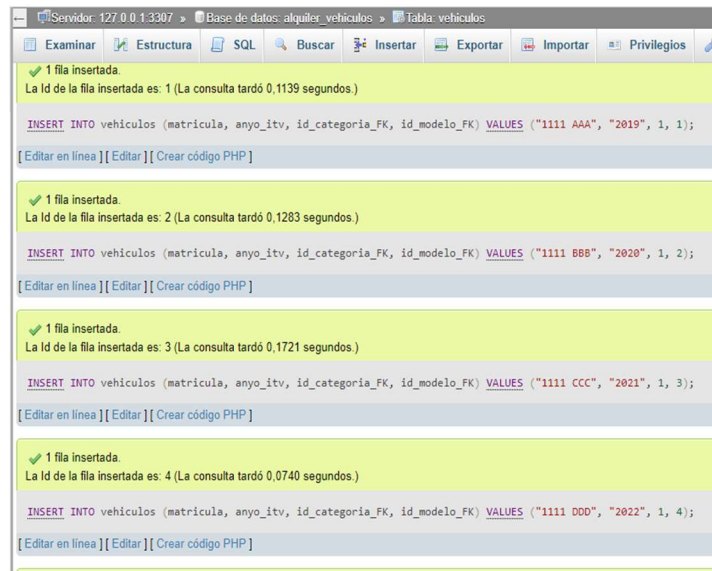
INSERT INTO vehiculos (matricula, anyo_itv, id_categoria_FK, id_modelo_FK)
VALUES ("1111 BBB", "2020", 1, 2);

INSERT INTO vehiculos (matricula, anyo_itv, id_categoria_FK, id_modelo_FK)
VALUES ("1111 CCC", "2021", 1, 3);

INSERT INTO vehiculos (matricula, anyo_itv, id_categoria_FK, id_modelo_FK)
VALUES ("1111 DDD", "2022", 1, 4); .....

```

Confirmación tras dar a continuar:



Código de inserción de datos de la tabla reservas:

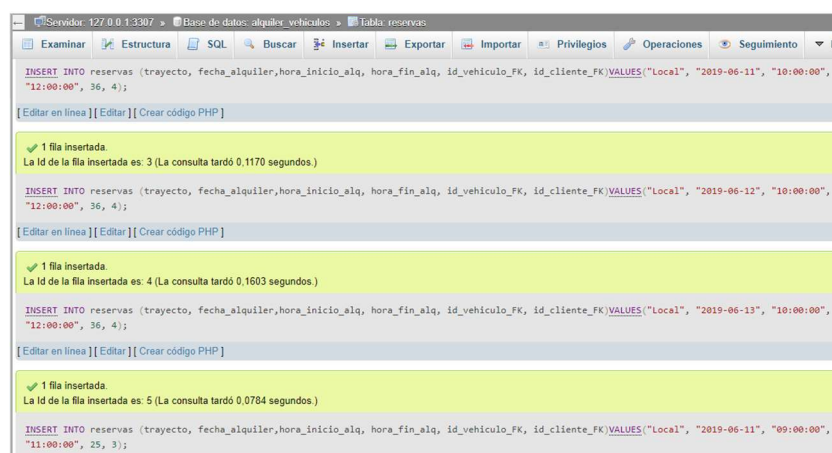
```
INSERT INTO reservas (trayecto, fecha_alquiler, hora_inicio_alq, hora_fin_alq, id_vehiculo_FK, id_cliente_FK) VALUES ("Local", "2019-06-11", "10:00:00", "12:00:00", 36, 4);
```

```
INSERT INTO reservas (trayecto, fecha_alquiler, hora_inicio_alq, hora_fin_alq, id_vehiculo_FK, id_cliente_FK) VALUES ("Local", "2019-06-12", "10:00:00", "12:00:00", 36, 4);
```

```
INSERT INTO reservas (trayecto, fecha_alquiler, hora_inicio_alq, hora_fin_alq, id_vehiculo_FK, id_cliente_FK) VALUES ("Local", "2019-06-13", "10:00:00", "12:00:00", 36, 4);
```

```
INSERT INTO reservas (trayecto, fecha_alquiler, hora_inicio_alq, hora_fin_alq, id_vehiculo_FK, id_cliente_FK) VALUES ("Local", "2019-06-11", "09:00:00", "11:00:00", 25, 3); .....
```

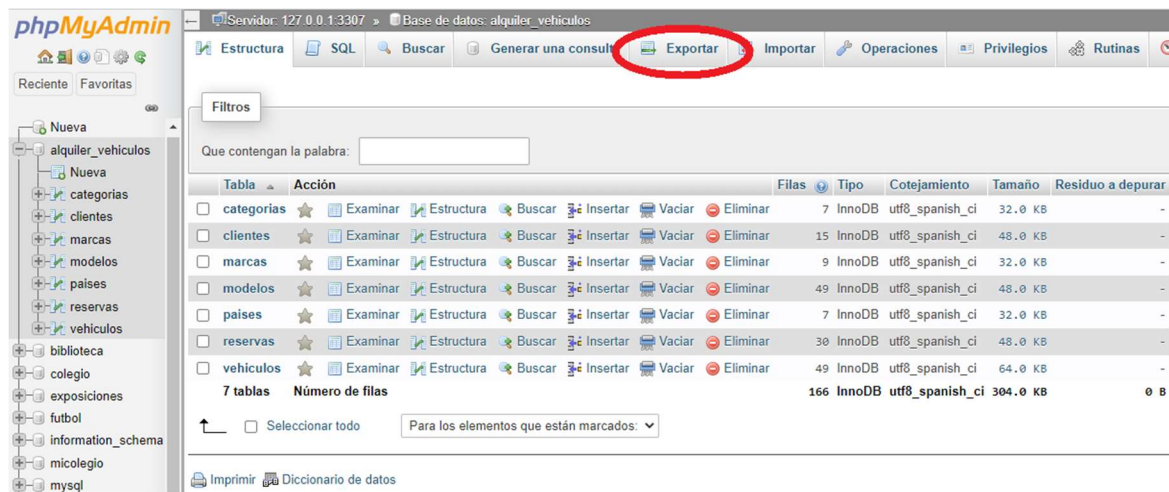
Confirmación tras dar a continuar:



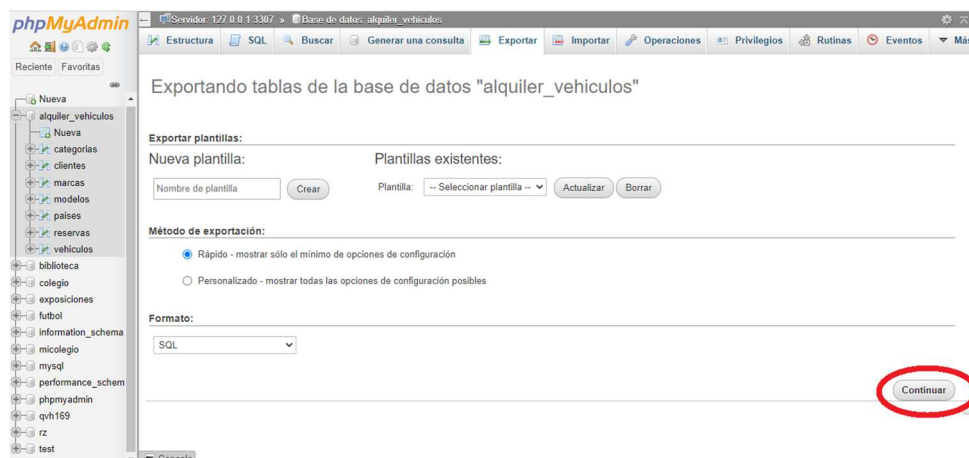
Ya tenemos creada la base de datos, las tablas y los datos insertados, realizamos una copia de seguridad de la base de datos:

1º Seleccionamos la base de datos que queremos copiar.

2º Damos a la pestaña Exportar que se encuentra en la parte superior.



3º Presionamos continuar y guardamos en el directorio que elijamos.



Y ya tenemos la copia de seguridad creada.

PROCEDAMOS A REALIZAR LAS CONSULTAS.

1º Consulta:

Muestra la cantidad de vehículos que tenemos de cada categoría, mostrando el nombre de la categoría y al lado la cantidad con el alias "Total_Vehículos".

Solución:

```
SELECT
c.categoria as Categoria,
COUNT(v.id_vehiculo) AS 'Total vehículos'
FROM vehiculos v
INNER JOIN categorias c ON (v.id_categoria_FK = c.id_categoria)
GROUP BY c.categoria
ORDER BY c.categoria ;
```

Mostrando filas 0 - 6 (total de 7, La consulta tardó 0,0081 segundos.)

```
SELECT c.categoria as Categoria, COUNT(v.id_vehiculo) AS 'Total vehículos' FROM vehiculos v INNER JOIN categorias c ON (v.id_categoria_FK = c.id_categoria)
GROUP BY c.categoria ORDER BY c.categoria ASC;
```

Perfilando [Editar en línea] [Editar] [Explicar SQL] [Crear código PHP] [Actualizar]

Mostrar todo | Número de filas: 25 | Filtrar filas: Buscar en esta tabla

+ Opciones

Categoría	Total vehículos
Berlina	7
Compacto	7
Coupé	7
Familiar	7
Monovolumen	7
Subcompacto	7
Vehículos comerciale	7

2ª Consulta:

Muestra toda la información de los clientes mayores a 40 años.

Solución:

```
SELECT * FROM clientes
WHERE (YEAR(CURRENT_DATE)-YEAR(fnac))>40;
```

Mostrando filas 0 - 1 (total de 2, La consulta tardó 0,0054 segundos.)

```
SELECT * FROM clientes WHERE (YEAR(CURRENT_DATE)-YEAR(fnac))>40;
```

Perfilando [Editar en línea] [Editar] [Explicar SQL] [Crear código PHP] [Actualizar]

Mostrar todo | Número de filas: 25 | Filtrar filas: Buscar en esta tabla | Sort by key: Ninguna

+ Opciones

	id_cliente	nombre	apellidos	dni_pasport	telefono	sexo	fnac	id_pais_FK
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar 4	Victor	Porcel	Martín	11111111P	925222222	H	1969-04-15	1
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar 8	Charles	Signoret	Regnault	55555555P	925666666	H	1970-03-23	3

Seleccionar todo | Para los elementos que están marcados: Editar Copiar Borrar Exportar

Mostrar todo | Número de filas: 25 | Filtrar filas: Buscar en esta tabla | Sort by key: Ninguna

3ª Consulta:

Muestra el nombre de los clientes que no tengan ninguna reserva realizada.

Solución:

```
SELECT
c.id_cliente AS 'ID del cliente',
CONCAT (c.nombre, ' ', c.apellidos) AS 'Nombre del cliente'
FROM clientes c
LEFT JOIN reservas r ON (c.id_cliente = r.id_cliente_FK)
WHERE r.id_cliente_FK IS NULL
```

Mostrando filas 0 - 1 (total de 2, La consulta tardó 0,0032 segundos.)

```
SELECT c.id_cliente AS 'ID del cliente', CONCAT (c.nombre, ' ', c.apellidos) AS 'Nombre del cliente' /*c.nombre || ' ' || c.apellidos AS 'Nombre de clientes'*/ FROM clientes c LEFT JOIN reservas r ON (c.id_cliente = r.id_cliente_FK) WHERE r.id_cliente_FK IS NULL;
```

Perfilando [Editar en línea] [Editar] [Explicar SQL] [Crear código PHP] [Actualizar]

☐ Mostrar todo | Número de filas: 25 | Filtrar filas: Buscar en esta tabla | Sort by key: Ninguna

+ Opciones

ID del cliente	Nombre del cliente
10	Morgan Müller
11	Marius Popa

☐ Mostrar todo | Número de filas: 25 | Filtrar filas: Buscar en esta tabla | Sort by key: Ninguna

4º Consulta:

Muestra todos los datos de los vehículos que se hayan reservado más de una vez.

Solución:

```
SELECT
v.id_vehiculo AS 'ID del vehículo',
c.categoria AS 'Categoría',
ma.marca AS 'Marca',
m.modelo AS 'Modelo',
v.matricula AS 'Matricula',
v.anyo_itv AS 'Año de la ITV',
COUNT(r.id_vehiculo_FK) as 'Nº de reservas'
FROM vehiculos v
INNER JOIN reservas r ON (v.id_vehiculo = r.id_vehiculo_FK)
INNER JOIN modelos m ON (v.id_modelo_FK = m.id_modelo)
INNER JOIN marcas ma ON (m.id_marca_FK = ma.id_marca)
INNER JOIN categorias c ON (v.id_categoria_FK = c.id_categoria)
GROUP BY r.id_vehiculo_FK
HAVING COUNT(r.id_vehiculo_FK)>1;
```

Servidor: 127.0.0.1:3307 » Base de datos: alquiler_vehiculos

Estructura SQL Buscar Generar una consulta Exportar Importar Operaciones Privilegios Rutinas Eventos

Ocultar ventana de consultas SQL

Mostrando filas 0 - 4 (total de 5, La consulta tardó 0,0215 segundos.)

```
SELECT v.id_vehiculo AS 'ID del vehículo', c.categoria AS 'Categoría', ma.marca AS 'Marca', m.modelo AS 'Modelo', v.matricula AS 'Matricula', v.anyo_itv AS 'Año de la ITV', COUNT(r.id_vehiculo_FK) as 'Nº de reservas' FROM vehiculos v INNER JOIN reservas r ON (v.id_vehiculo = r.id_vehiculo_FK) INNER JOIN modelos m ON (v.id_modelo_FK = m.id_modelo) INNER JOIN marcas ma ON (m.id_marca_FK = ma.id_marca) INNER JOIN categorias c ON (v.id_categoria_FK = c.id_categoria) GROUP BY r.id_vehiculo_FK HAVING COUNT(r.id_vehiculo_FK)>1;
```

Perfilando [Editar en línea] [Editar] [Explicar SQL] [Crear código PHP] [Actualizar]

☐ Mostrar todo | Número de filas: 25 | Filtrar filas: Buscar en esta tabla

+ Opciones

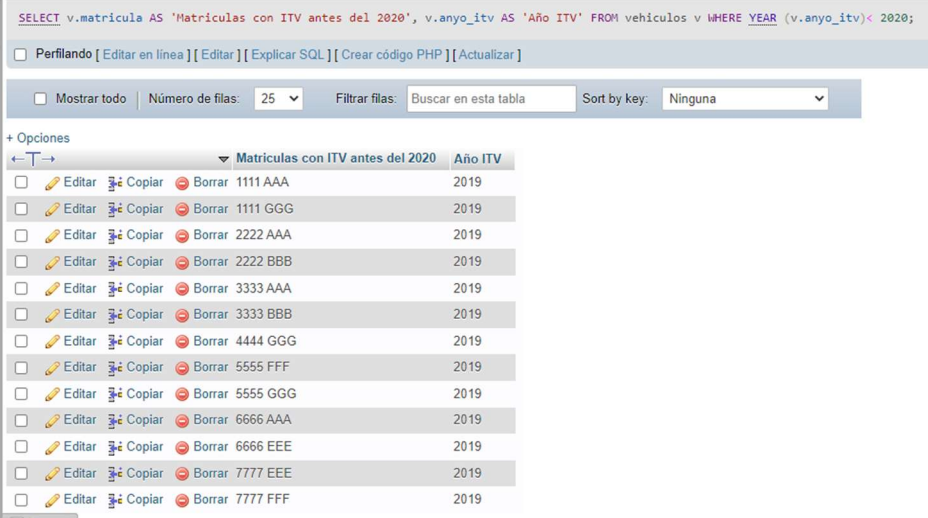
ID del vehículo	Categoría	Marca	Modelo	Matricula	Año de la ITV	Nº de reservas
7	Subcompacto	BMW	X2	1111 GGG	2019	4
25	Berlina	Opel	Insignia	4444 DDD	2021	3
36	Monovolumen	Renault	Scénic	6666 AAA	2019	4
41	Monovolumen	Opel	Combo Life	6666 FFF	2020	2
47	Vehiculos comerciale	Mercedes	Sprinter Furgón	7777 EEE	2019	2

5ª Consulta:

Muestra la matrícula de los coches que hayan pasado la itv antes del año 2020.

Solución:

```
SELECT  
  
v.matricula AS 'Matriculas con ITV antes del 2020',  
  
v.anyo_itv AS 'Año ITV'  
  
FROM vehiculos v  
  
WHERE YEAR (v.anyo_itv)< 2020;
```



SELECT v.matricula AS 'Matriculas con ITV antes del 2020', v.anyo_itv AS 'Año ITV' FROM vehiculos v WHERE YEAR (v.anyo_itv)< 2020;

Perfilando [Editar en línea] [Editar] [Explicar SQL] [Crear código PHP] [Actualizar]

Mostrar todo | Número de filas: 25 | Filtrar filas: Buscar en esta tabla | Sort by key: Ninguna

+ Opciones

				Matriculas con ITV antes del 2020	Año ITV
<input type="checkbox"/>	Editar	Copiar	Borrar	1111 AAA	2019
<input type="checkbox"/>	Editar	Copiar	Borrar	1111 GGG	2019
<input type="checkbox"/>	Editar	Copiar	Borrar	2222 AAA	2019
<input type="checkbox"/>	Editar	Copiar	Borrar	2222 BBB	2019
<input type="checkbox"/>	Editar	Copiar	Borrar	3333 AAA	2019
<input type="checkbox"/>	Editar	Copiar	Borrar	3333 BBB	2019
<input type="checkbox"/>	Editar	Copiar	Borrar	4444 GGG	2019
<input type="checkbox"/>	Editar	Copiar	Borrar	5555 FFF	2019
<input type="checkbox"/>	Editar	Copiar	Borrar	5555 GGG	2019
<input type="checkbox"/>	Editar	Copiar	Borrar	6666 AAA	2019
<input type="checkbox"/>	Editar	Copiar	Borrar	6666 EEE	2019
<input type="checkbox"/>	Editar	Copiar	Borrar	7777 EEE	2019
<input type="checkbox"/>	Editar	Copiar	Borrar	7777 FFF	2019

6ª Consulta:

Muestra la cantidad de clientes que hay de cada país ordena-dos alfabéticamente con el alias "Total Clientes".

Solución:

```
SELECT  
  
p.pais AS País,  
  
COUNT(c.id_cliente)AS 'Total clientes'  
  
FROM clientes c  
  
INNER JOIN paises p ON (c.id_pais_FK = p.id_pais)  
  
GROUP BY País  
  
ORDER BY País ASC;
```

✓ Mostrando filas 0 - 6 (total de 7, La consulta tardó 0,0065 segundos.) [País: ALEMANIA... - RUMANIA...]

```
SELECT p.país AS País, COUNT(c.id_cliente) AS 'Total clientes' FROM clientes c INNER JOIN países p ON (c.id_país_FK = p.id_país) GROUP BY País ORDER BY País ASC;
```

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

☐ Mostrar todo | Número de filas: 25 | Filtrar filas:

+ Opciones

País	Total clientes
Alemania	2
España	3
Francia	2
Italia	2
Marruecos	2
Portugal	2
Rumania	2

☐ Mostrar todo | Número de filas: 25 | Filtrar filas:

7ª Consulta:

Muestra el promedio de edades de los clientes, el cliente más joven y el cliente más mayor de cada país.

Solución:

```
SELECT
p.país AS País,
YEAR(CURRENT_DATE)-YEAR(MAX(c.fnac)) AS 'El más joven tiene',
YEAR(CURRENT_DATE)-YEAR(MIN(c.fnac)) AS 'El más mayor tiene',
TRUNCATE(AVG (YEAR(CURRENT_DATE)-YEAR(c.fnac)),2)AS 'Promedio de edad'
FROM clientes c
INNER JOIN países p ON (c.id_país_FK = p.id_país)
GROUP BY (País)
ORDER BY (País);
```

✓ Mostrando filas 0 - 6 (total de 7, La consulta tardó 0,0110 segundos.)

```
SELECT p.país AS País, YEAR(CURRENT_DATE)-YEAR(MAX(c.fnac)) AS 'El más joven tiene', YEAR(CURRENT_DATE)-YEAR(MIN(c.fnac)) AS 'El más mayor tiene', TRUNCATE(AVG (YEAR(CURRENT_DATE)-YEAR(c.fnac)),2)AS 'Promedio de edad' FROM clientes c INNER JOIN países p ON (c.id_país_FK = p.id_país) GROUP BY (País) ORDER BY (País);
```

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

☐ Mostrar todo | Número de filas: 25 | Filtrar filas:

+ Opciones

País	El más joven tiene	El más mayor tiene	Promedio de edad
Alemania	27	34	30.50
España	29	53	40.66
Francia	20	52	36.00
Italia	23	25	24.00
Marruecos	35	37	36.00
Portugal	24	31	27.50
Rumania	21	22	21.50

8º Consulta:

Muestra el nombre de los clientes y la cantidad de vehículos que ha reservado cada uno, ordenado alfabéticamente.

Solución:

```
SELECT  
  
CONCAT(c.nombre, ' ',c.apellidos) AS 'Nombre del cliente',  
  
COUNT(r.id_reserva) AS 'Nº de reservas'  
  
FROM clientes c  
  
INNER JOIN reservas r ON (c.id_cliente = r.id_cliente_FK)  
  
GROUP BY c.nombre  
  
ORDER BY c.nombre ASC;
```

Mostrando filas 0 - 12 (total de 13). La consulta tardó 0,0082 segundos.)

```
SELECT CONCAT(c.nombre, ' ',c.apellidos) AS 'Nombre del cliente', COUNT(r.id_reserva) AS 'Nº de reservas' FROM clientes c INNER JOIN reservas r ON (c.id_cliente = r.id_cliente_FK) GROUP BY c.nombre ORDER BY c.nombre ASC;
```

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

☐ Mostrar todo | Número de filas: 25 | Filtrar filas:

+ Opciones

Nombre del cliente	Nº de reservas
Alexandru Radu	1
Bruno Rossi Ferrari	2
Camille Lombard Briand	2
Caterina Oliveira Cortes	4
Charles Signoret Renault	1
Daniilo Pereira Vidal	1
Franco Cabano Fabene	1
Giselle Holbin	1
Manuel Garcia Rodriguez	1
Maria Lopez Lopez	5
Mohamed Alami	3
Said Alaoui	1
Victor Porcel Martin	7

9º Consulta:

Muestra el nombre de las marcas y la cantidad de vehículos que pertenecen a cada una.

Solución:

```
SELECT  
  
m.marca AS Marca,  
  
COUNT(v.id_vehiculo) AS Cantidad  
  
FROM vehiculos v  
  
INNER JOIN modelos mo ON (v.id_modelo_FK = mo.id_modelo)  
  
INNER JOIN marcas m ON (mo.id_marca_FK = m.id_marca)  
  
GROUP BY m.marca  
  
ORDER BY m.marca ASC;
```

Mostrando filas 0 - 8 (total de 9, La consulta tardó 0,0267 segundos.)

```
SELECT m.marca AS Marca, COUNT(v.id_vehiculo) AS Cantidad FROM vehiculos v INNER JOIN modelos mo ON (v.id_modelo_FK = mo.id_modelo) INNER JOIN marcas m ON (mo.id_marca_FK = m.id_marca) GROUP BY m.marca ORDER BY m.marca ASC;
```

Perfilando [Editar en línea] [Editar] [Explicar SQL] [Crear código PHP] [Actualizar]

Mostrar todo | Número de filas: 25 | Filtrar filas: Buscar en esta tabla

+ Opciones

Marca	Cantidad
BMW	6
Citroën	2
Ford	7
KIA	4
Mercedes	4
Opel	10
Peugeot	3
Renault	8
Seat	5

10º Consulta:

Muestra todos los datos de los clientes españoles que hayan reservado un vehículo más de una vez.

Solución:

```
SELECT
c.id_cliente AS 'ID Cliente',
CONCAT(c.nombre, ' ', c.apellidos) AS 'Nombre de cliente',
c.dni_pasport AS DNI,
c.telefono AS Teléfono,
c.sexo AS Sexo,
c.fnac AS 'Fecha de nacimiento'
FROM clientes c
WHERE c.id_pais_FK IN (SELECT
p.id_pais
FROM paises p
WHERE p.pais = 'España') AND
c.id_cliente IN (SELECT
r.id_cliente_FK
FROM reservas r
GROUP BY r.id_cliente_FK
HAVING COUNT(r.id_reserva)>=2);
```

Mostrando filas 0 - 1 (total de 2, La consulta tardó 0.0212 segundos.)

SELECT c.id_cliente AS 'ID Cliente', CONCAT(c.nombre, ' ', c.apellidos) AS 'Nombre de cliente', c.dni_pasport AS DNI, c.telefono AS Teléfono, c.sexo AS Sexo, c.fnac AS 'Fecha de nacimiento' FROM clientes c WHERE c.id_pais_FK IN (SELECT p.id_pais FROM paises p WHERE p.pais = 'España') AND c.id_cliente IN (SELECT r.id_cliente_FK FROM reservas r GROUP BY r.id_cliente_FK HAVING COUNT(r.id_reserva)>2);

Perfilando

Editar en línea

Editar

Explicar SQL

Crear código PHP

Actualizar

Mostrar todo

Número de filas: 25

Filtrar filas:

Buscar en esta tabla

Sort by key: Ninguna

+ Opciones

←

T

→

ID Cliente

Nombre de cliente

DNI

Teléfono

Sexo

Fecha de nacimiento

Editar

Copiar

Borrar

3

Editar

Copiar

Borrar

4

Maria Lopez Lopez

00000000P

925111111

Mujer

1982-12-05

Victor Porcel Martin

11111111P

925222222

Hombre

1969-04-15

11º Consulta:

Muestra todos los datos de los vehículos de la marca “Renault” que hayan sido reservados menos de tres veces.

Solución:

```
SELECT
v.id_vehiculo AS 'ID Vehículo',
c.categoria AS 'Categoría',
ma.marca AS 'Marca',
m.modelo AS 'Modelo',
v.matricula AS 'Matrícula',
v.anyo_itv AS 'Año ITV'
FROM vehiculos v
INNER JOIN modelos m ON (v.id_modelo_FK = m.id_modelo)
INNER JOIN marcas ma ON (m.id_marca_FK = ma.id_marca)
INNER JOIN categorias c ON (v.id_categoria_FK = c.id_categoria)
WHERE ma.id_marca IN (SELECT
                        ma.id_marca
                        FROM marcas ma
                        WHERE ma.marca = 'Renault') AND
v.id_vehiculo IN (SELECT
                  r.id_vehiculo_FK
                  FROM reservas r
                  GROUP BY r.id_vehiculo_FK
                  HAVING COUNT(r.id_reserva)<3);
```

✓ Mostrando filas 0 - 2 (total de 3. La consulta tardó 0,0276 segundos.)

```
SELECT v.id_vehiculo AS 'ID Vehículo', c.categoria AS 'Categoría', ma.marca AS 'Marca', m.modelo AS 'Modelo', v.matricula AS 'Matrícula', v.anyo_itv AS 'Año ITV'
FROM vehiculos v INNER JOIN modelos m ON (v.id_modelo_FK = m.id_modelo) INNER JOIN marcas ma ON (m.id_marca_FK = ma.id_marca) INNER JOIN categorias c ON
(v.id_categoria_FK = c.id_categoria) WHERE ma.id_marca IN (SELECT ma.id_marca FROM marcas ma WHERE ma.marca = 'Renault') AND v.id_vehiculo IN (SELECT
r.id_vehiculo_FK FROM reservas r GROUP BY r.id_vehiculo_FK HAVING COUNT(r.id_reserva)<3);
```

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

☐ Mostrar todo | Número de filas: 25 | Filtrar filas:

+ Opciones

ID Vehículo	Categoría	Marca	Modelo	Matrícula	Año ITV
33	Coupé	Renault	Arkna Coupé	5555 EEE	2022
44	Vehículos comerciale	Renault	Kangoo Furgón	7777 BBB	2021
45	Vehículos comerciale	Renault	Trafic Furgón	7777 CCC	2021

☐ Mostrar todo | Número de filas: 25 | Filtrar filas:

12º Consulta:

Muestra la matrícula y año de itv de los vehículos que pertenezcan a la categoría "Berlina" y que no se hayan reservado nunca.

Solución:

```
SELECT
v.matricula AS Matrículas,
v.anyo_itv AS 'Año ITV'
FROM vehiculos v
WHERE v.id_categoria_FK IN (SELECT
c.id_categoria
FROM categorias c
WHERE c.categoria = 'Berlina') AND
v.id_vehiculo IN (SELECT
v.id_vehiculo
FROM reservas r
RIGHT JOIN vehiculos v ON (r.id_vehiculo_FK = v.id_vehiculo)
WHERE r.id_vehiculo_FK IS NULL);
```

✓ Mostrando filas 0 - 4 (total de 5, La consulta tardó 0,0141 segundos.)

```
SELECT v.matricula AS Matrículas, v.anyo_itv AS 'Año ITV' FROM vehiculos v WHERE v.id_categoria_FK IN (SELECT c.id_categoria FROM categorias c WHERE c.categoria = 'Berlina') AND v.id_vehiculo IN (SELECT v.id_vehiculo FROM reservas r RIGHT JOIN vehiculos v ON (r.id_vehiculo_FK = v.id_vehiculo) WHERE r.id_vehiculo_FK IS NULL);
```

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

☐ Mostrar todo | Número de filas: 25 | Filtrar filas: Buscar en esta tabla | Sort by key: Ninguna

+ Opciones

			Matrículas	Año ITV
<input type="checkbox"/>				4444 AAA 2022
<input type="checkbox"/>				4444 CCC 2021
<input type="checkbox"/>				4444 EEE 2020
<input type="checkbox"/>				4444 FFF 2020
<input type="checkbox"/>				4444 GGG 2019

13º Consulta:

Muestra el nombre de los clientes españoles cuya edad sea superior a la media.

Solución:

```
FROM clientes c
WHERE (c.id_pais_FK IN (
```

```

SELECT

    p.id_pais

FROM paises p

WHERE p.pais = 'España')

) AND

(YEAR(CURRENT_DATE)-YEAR(c.fnac)) > (

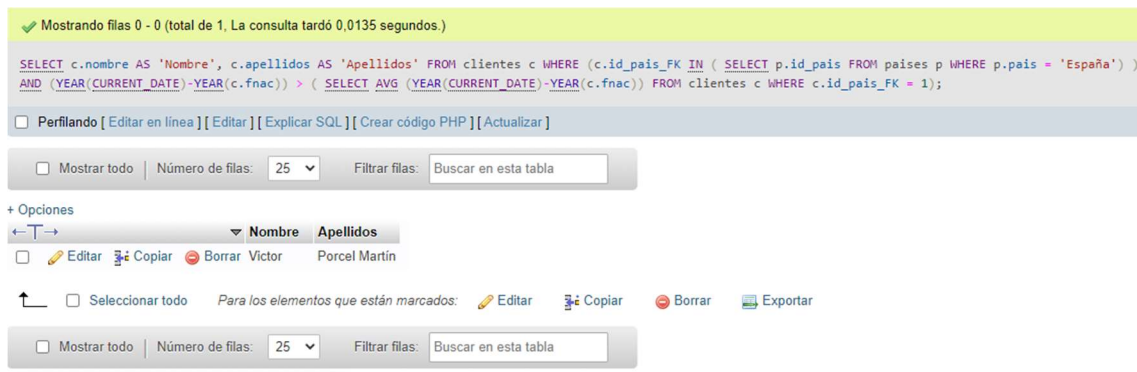
SELECT

    AVG (YEAR(CURRENT_DATE)-YEAR(c.fnac))

FROM clientes c

WHERE c.id_pais_FK = 1);

```



14º Consulta:

Modifica el campo de sexo para que aparezca "H" de hombre dónde actualmente aparecía la palabra que habías designado para los hombres.

Solución:

```

/*Primero creamos los valores del enum insertando 'H'*/

ALTER TABLE clientes MODIFY COLUMN sexo enum('Mujer', 'Hombre','H', 'N/C');

/*Cambiamos los valores 'Hombre' de la columna sexo por 'H'*/

UPDATE clientes SET sexo = 'H' WHERE sexo = 'Hombre';

/*Borramos el valor 'Hombre' de la lista de enum */

ALTER TABLE clientes MODIFY COLUMN sexo enum('Mujer','H', 'N/C');

```

```

✓ MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 1,0207 segundos.)

/*Primero creamos los valores del enum insertando 'H'*/ ALTER TABLE clientes MODIFY COLUMN sexo enum('Mujer', 'Hombre', 'H', 'N/C');
[ Editar en línea ] [ Editar ] [ Crear código PHP ]

✓ 11 filas afectadas. (La consulta tardó 0,2122 segundos.)

/*Cambiamos los valores 'Hombre' de la columna sexo por 'H'*/ UPDATE clientes SET sexo = 'H' WHERE sexo = 'Hombre';
[ Editar en línea ] [ Editar ] [ Crear código PHP ]

✓ MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,8254 segundos.)

/*Borramos el valor 'Hombre' de la lista de enum */ ALTER TABLE clientes MODIFY COLUMN sexo enum('Mujer', 'H', 'N/C');
[ Editar en línea ] [ Editar ] [ Crear código PHP ]

```

+ Opciones

				id_cliente	nombre	apellidos	dni_pasport	telefono	sexo	fnac	id_pais_FK
<input type="checkbox"/>	Editar	Copiar	Borrar	3	Maria	Lopez Lopez	00000000P	925111111	Mujer	1982-12-05	1
<input type="checkbox"/>	Editar	Copiar	Borrar	4	Victor	Porcel Martín	11111111P	925222222	H	1969-04-15	1
<input type="checkbox"/>	Editar	Copiar	Borrar	5	Caterina	Oliveira Cortes	22222222P	925333333	Mujer	1991-12-31	2
<input type="checkbox"/>	Editar	Copiar	Borrar	6	Daniilo	Pereira Vidal	33333333P	925444444	H	1998-07-11	2
<input type="checkbox"/>	Editar	Copiar	Borrar	7	Camille	Lombrad Briand	44444444P	925555555	Mujer	2002-08-25	3
<input type="checkbox"/>	Editar	Copiar	Borrar	8	Charles	Signoret Regnault	55555555P	925666666	H	1970-03-23	3
<input type="checkbox"/>	Editar	Copiar	Borrar	9	Giselle	Holbin	66666666P	925777777	Mujer	1995-06-11	4
<input type="checkbox"/>	Editar	Copiar	Borrar	10	Morgan	Müller	77777777P	925888888	H	1988-11-15	4

15º Consulta:

Modifica el país para que aparezca "Spain" dónde actualmente aparece "España"

Solución:

UPDATE paises SET paises.pais = 'Spain' WHERE paises.pais = 'España';

```

✓ 1 fila afectada. (La consulta tardó 0,0619 segundos.)

UPDATE paises SET paises.pais = 'Spain' WHERE paises.pais = 'España';
[ Editar en línea ] [ Editar ] [ Crear código PHP ]

```

+ Opciones

				id_pais	pais
<input type="checkbox"/>	Editar	Copiar	Borrar	1	Spain
<input type="checkbox"/>	Editar	Copiar	Borrar	2	Portugal
<input type="checkbox"/>	Editar	Copiar	Borrar	3	Francia
<input type="checkbox"/>	Editar	Copiar	Borrar	4	Alemania
<input type="checkbox"/>	Editar	Copiar	Borrar	5	Rumania
<input type="checkbox"/>	Editar	Copiar	Borrar	6	Marruecos
<input type="checkbox"/>	Editar	Copiar	Borrar	7	Italia

16º Consulta:

Asigna la palabra "corto" a los trayectos para los que el tiempo de reserva haya sido inferior a 2 horas, asigna la palabra "medio" a los trayectos para los que el tiempo de reserva del vehículo haya sido de más de 3 horas e inferior a 5 horas y asigna la palabra "largo" para todos los trayectos en los que el tiempo de reserva del vehículo haya sido superior a las 5 horas.

Solución:

```
UPDATE reservas SET reservas.trayecto = 'Corto' WHERE (hour(reservas.hora_fin_alq)-
hour(reservas.hora_inicio_alq)<=2);
```

```
UPDATE reservas SET reservas.trayecto = 'Medio' WHERE (hour(reservas.hora_fin_alq)-
hour(reservas.hora_inicio_alq)>2 AND hour(reservas.hora_fin_alq)-
hour(reservas.hora_inicio_alq)<=5);
```

```
UPDATE reservas SET reservas.trayecto = 'Largo' WHERE (hour(reservas.hora_fin_alq)-
hour(reservas.hora_inicio_alq)>5);
```

11 filas afectadas. (La consulta tardó 0,0438 segundos.)

```
UPDATE reservas SET reservas.trayecto = 'Corto' WHERE (hour(reservas.hora_fin_alq)-hour(reservas.hora_inicio_alq)<=2);
```

[Editar en línea] [Editar] [Crear código PHP]

0 filas afectadas. (La consulta tardó 0,0064 segundos.)

```
UPDATE reservas SET reservas.trayecto = 'Medio' WHERE (hour(reservas.hora_fin_alq)-hour(reservas.hora_inicio_alq)>2 AND hour(reservas.hora_fin_alq)-hour(reservas.hora_inicio_alq)<=5);
```

[Editar en línea] [Editar] [Crear código PHP]

0 filas afectadas. (La consulta tardó 0,0033 segundos.)

```
UPDATE reservas SET reservas.trayecto = 'Largo' WHERE (hour(reservas.hora_fin_alq)-hour(reservas.hora_inicio_alq)>5);
```

[Editar en línea] [Editar] [Crear código PHP]

+ Opciones

				id_reserva	trayecto	fecha_alquiler	hora_inicio_alq	hora_fin_alq
<input type="checkbox"/>	Editar		Copiar		Borrar	1	Corto	2019-06-11 10:00:00 12:00:00
<input type="checkbox"/>	Editar		Copiar		Borrar	2	Corto	2019-06-11 10:00:00 12:00:00
<input type="checkbox"/>	Editar		Copiar		Borrar	3	Corto	2019-06-12 10:00:00 12:00:00
<input type="checkbox"/>	Editar		Copiar		Borrar	4	Corto	2019-06-13 10:00:00 12:00:00
<input type="checkbox"/>	Editar		Copiar		Borrar	5	Corto	2019-06-11 09:00:00 11:00:00
<input type="checkbox"/>	Editar		Copiar		Borrar	6	Corto	2019-06-11 09:00:00 11:00:00
<input type="checkbox"/>	Editar		Copiar		Borrar	7	Corto	2020-05-25 09:00:00 11:00:00
<input type="checkbox"/>	Editar		Copiar		Borrar	8	Medio	2020-06-15 09:00:00 14:00:00
<input type="checkbox"/>	Editar		Copiar		Borrar	9	Corto	2020-08-24 09:00:00 11:00:00
<input type="checkbox"/>	Editar		Copiar		Borrar	10	Medio	2020-09-25 10:00:00 13:00:00
<input type="checkbox"/>	Editar		Copiar		Borrar	11	Medio	2020-10-02 09:00:00 14:00:00
<input type="checkbox"/>	Editar		Copiar		Borrar	12	Medio	2020-11-26 09:00:00 14:00:00
<input type="checkbox"/>	Editar		Copiar		Borrar	13	Medio	2021-01-05 08:00:00 12:00:00
<input type="checkbox"/>	Editar		Copiar		Borrar	14	Medio	2021-02-14 09:00:00 12:00:00
<input type="checkbox"/>	Editar		Copiar		Borrar	15	Largo	2021-03-15 09:00:00 22:00:00
<input type="checkbox"/>	Editar		Copiar		Borrar	16	Largo	2021-04-15 09:00:00 22:00:00

17º Consulta:

Crea una vista que permita recuperar toda la información de las reservas realizadas por clientes no españoles.

Solución:

/*Para crear una vista utilizamos la sintaxis siguiente:*/

```
CREATE OR REPLACE VIEW vista_reservas AS
SELECT
r.id_reserva AS ID,
r.trayecto AS Recorrido,
r.fecha_alquiler AS 'Fecha alquiler',
```



```

CONCAT(r.hora_inicio_alq, ' - ', r.hora_fin_alq) AS 'Hora de salida y entrada',

ca.categoria AS 'Tipo de vehículo',

ma.marca AS Marca,

m.modelo AS Modelo,

CONCAT(c.nombre, ' ', c.apellidos) AS 'Nombre del cliente',

p.pais AS País

FROM reservas r

INNER JOIN vehiculos v ON (r.id_vehiculo_FK = v.id_vehiculo)

INNER JOIN categorias ca ON (v.id_categoria_FK = ca.id_categoria)

INNER JOIN modelos m ON (v.id_modelo_FK = m.id_modelo)

INNER JOIN marcas ma ON (m.id_marca_FK = ma.id_marca)

INNER JOIN clientes c ON (r.id_cliente_FK = c.id_cliente)

INNER JOIN paises p ON (c.id_pais_FK = p.id_pais)

WHERE p.pais != 'Spain';

```



ID	Recorrido	Fecha alquiler	Hora de salida y entrada	Tipo de vehiculo	Marca	Modelo	Nombre del cliente	País
9	Corto	2020-08-24	09:00:00 - 11:00:00	Subcompacto	BMW	X2	Charles Signoret Renault	Francia
10	Medio	2020-09-25	10:00:00 - 13:00:00	Subcompacto	BMW	X2	Camille Lombrad Briand	Francia
11	Medio	2020-10-02	09:00:00 - 14:00:00	Subcompacto	BMW	X2	Giselle Holbin	Alemania
12	Medio	2020-11-26	09:00:00 - 14:00:00	Subcompacto	Seat	Ateca	Caterina Oliveira Cortes	Portugal
13	Medio	2021-01-05	08:00:00 - 12:00:00	Monovolumen	Opel	Vivaro Tourer	Caterina Oliveira Cortes	Portugal
14	Medio	2021-02-14	09:00:00 - 12:00:00	Monovolumen	Opel	Combo Life	Caterina Oliveira Cortes	Portugal
17	Largo	2021-05-15	09:00:00 - 22:00:00	Vehiculos comerciale	Renault	Trafic Furgón	Alexandru Radu	Rumania
18	Largo	2021-06-15	09:00:00 - 22:00:00	Vehiculos comerciale	Renault	Kangoo Furgón	Mohamed Alami	Marruecos
19	Largo	2021-07-15	08:00:00 - 23:00:00	Monovolumen	Seat	Alhambra	Mohamed Alami	Marruecos
20	Largo	2021-08-15	08:00:00 - 23:00:00	Vehiculos comerciale	Ford	Transit Furgón	Mohamed Alami	Marruecos
21	Largo	2021-09-15	08:00:00 - 23:00:00	Vehiculos comerciale	Opel	Movano Furgón	Said Alaovi	Marruecos
22	Corto	2019-01-05	09:00:00 - 11:00:00	Familiar	Mercedes	Clase C	Bruno Rossi Ferrari	Italia
23	Medio	2019-01-17	09:00:00 - 13:00:00	Berlina	Mercedes	Clase S	Danilo Pereira Vidal	Portugal
25	Corto	2019-02-04	09:00:00 - 11:00:00	Subcompacto	BMW	X2	Caterina Oliveira Cortes	Portugal
28	Corto	2019-03-04	09:00:00 - 11:00:00	Coupé	BMW	Serie 6	Franco Cabano Fabene	Italia
29	Medio	2019-03-06	09:00:00 - 13:00:00	Coupé	Renault	Arkna Coupé	Bruno Rossi Ferrari	Italia

18º Consulta:

Realiza una copia de seguridad de la base de datos.

Procedimiento identoco al seguido en la copia de seguridad tras la creación de la base, tabla e inserción de los datos.

19º Consulta:

Elimina la tabla de categorías.

Solución:

```

/*En primer lugar debemos averiguar como
es nombreda la clave foranea de la tabla

```

vehiculos que está relacionada con la tabla

categoría, con la siguiente sentencia.*/

SHOW CREATE TABLE vehiculos;

```
Su consulta se ejecutó con éxito.

SHOW CREATE TABLE vehiculos;

Perfilando [ Editar en línea ] [ Editar ] [ Crear código PHP ] [ Actualizar ]

+ Opciones
Table Create Table
CREATE TABLE `vehiculos` (
  `id_vehiculo` int(11) NOT NULL AUTO_INCREMENT,
  `matricula` varchar(8) COLLATE utf8_spanish_ci NOT NULL,
  `anyo_itv` year(4) DEFAULT NULL,
  `id_categoria_fk` int(11) DEFAULT NULL,
  `id_modelo_fk` int(11) DEFAULT NULL,
  PRIMARY KEY (`id_vehiculo`),
  UNIQUE KEY `matricula` (`matricula`),
  KEY `id_categoria_fk` (`id_categoria_fk`),
  KEY `id_modelo_fk` (`id_modelo_fk`),
  CONSTRAINT `vehiculos_ibfk_1` FOREIGN KEY (`id_categoria_fk`) REFERENCES `categorias` (`id_categoria`),
  CONSTRAINT `vehiculos_ibfk_2` FOREIGN KEY (`id_modelo_fk`) REFERENCES `modelos` (`id_modelo`)
) ENGINE=InnoDB AUTO_INCREMENT=50 DEFAULT CHARSET=utf8 COLLATE=utf8_spanish_ci
```

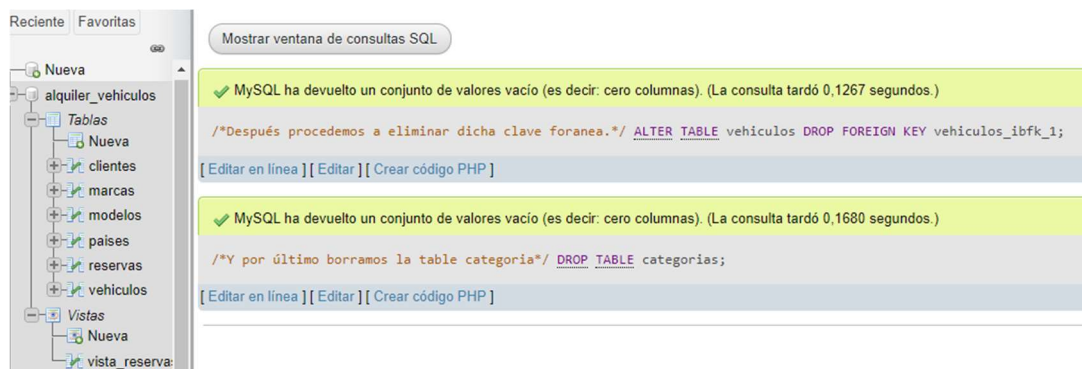
/*Después procedemos a eliminar dicha clave foranea.*/

ALTER TABLE vehiculos

DROP FOREIGN KEY vehiculos_ibfk_1;

/*Y por último borramos la table categoria*/

DROP TABLE categorias;



20º Consulta:

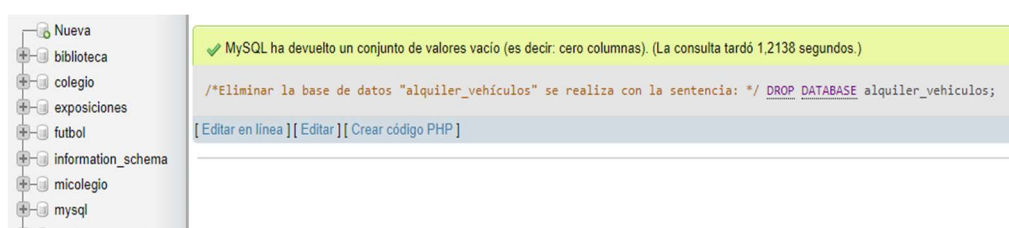
Elimina la base de datos.

Solución:

/*Eliminar la base de datos "alquiler_vehículos"

se realiza con la sentencia: */

DROP DATABASE alquiler_vehiculos;



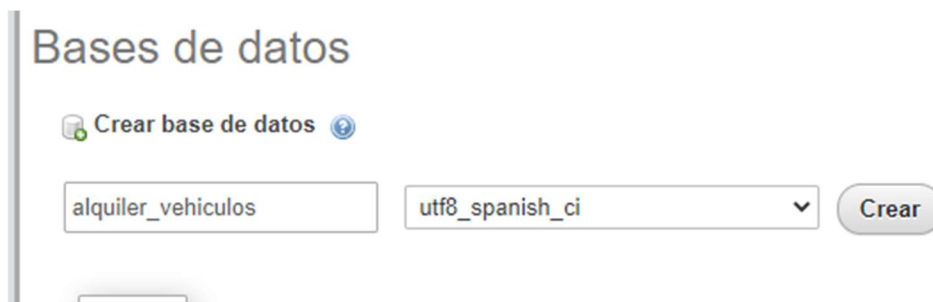
21ª Consulta:

Importa la copia de seguridad de la base de datos.

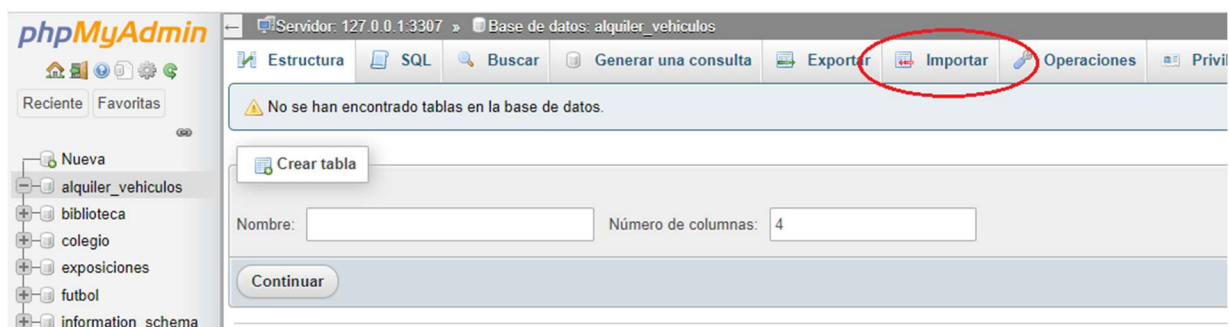
Solución:

Tras haber eliminado la base de datos y querer importar la copia de seguridad que tenemos creada, lo primero que tenemos que hacer es crear una base de datos nueva y vacía para después importar los datos.

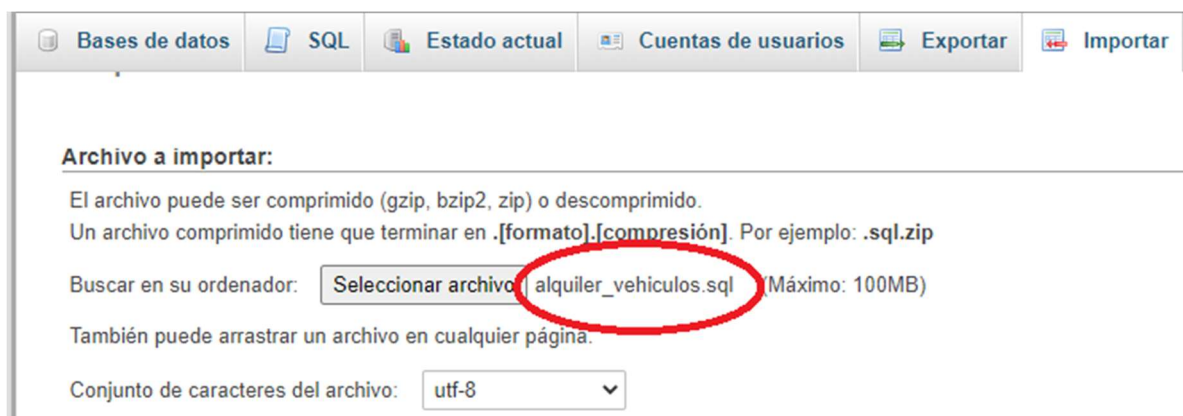
La pantalla que nos sale tras dar a nueva base de datos es:



Una vez ya creada la base de datos, la seleccionamos y le damos a la etiqueta "Importar", como se muestra en la imagen:



Nos aparece una nueva pantalla donde le damos a donde nos dice "seleccionar archivo" y cargamos nuestro archivo que contiene la base de datos.



Le damos a continuar y ya tenemos importada y cargada la base de datos.

phpMyAdmin

Servidor: 127.0.0.1:3307 » Base de datos: alquiler_vehiculos

Estructura SQL Buscar Generar una consulta Exportar Importar Operaciones Privilegios Rutinas Más

Reciente Favoritas

Nueva

alquiler_vehiculos

Tablas

- Nueva
- categorias
- clientes
- marcas
- modelos
- paises
- reservas
- vehiculos
- Vistas

Importación ejecutada exitosamente, 46 consultas ejecutadas. (alquiler_vehiculos.sql)

MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0012 segundos.)

```
-- phpMyAdmin SQL Dump -- version 5.1.1 -- https://www.phpmyadmin.net/ -- -- Servidor: 127.0.0.1:3307 -- Tiempo de generación: 04-04-2022 a las 17:52:17 --
Versión del servidor: 10.4.22-MariaDB -- Versión de PHP: 8.0.15 SET SQL_MODE = "NO_AUTO_VALUE_ON_ZERO";
```

[Editar en línea] [Editar] [Crear código PHP]

MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0002 segundos.)

```
START TRANSACTION;
```

[Editar en línea] [Editar] [Crear código PHP]