

Escenario

Una empresa proveedora de Internet necesita una base de datos para almacenar cada uno de sus clientes junto con el plan/pack que tiene contratado.

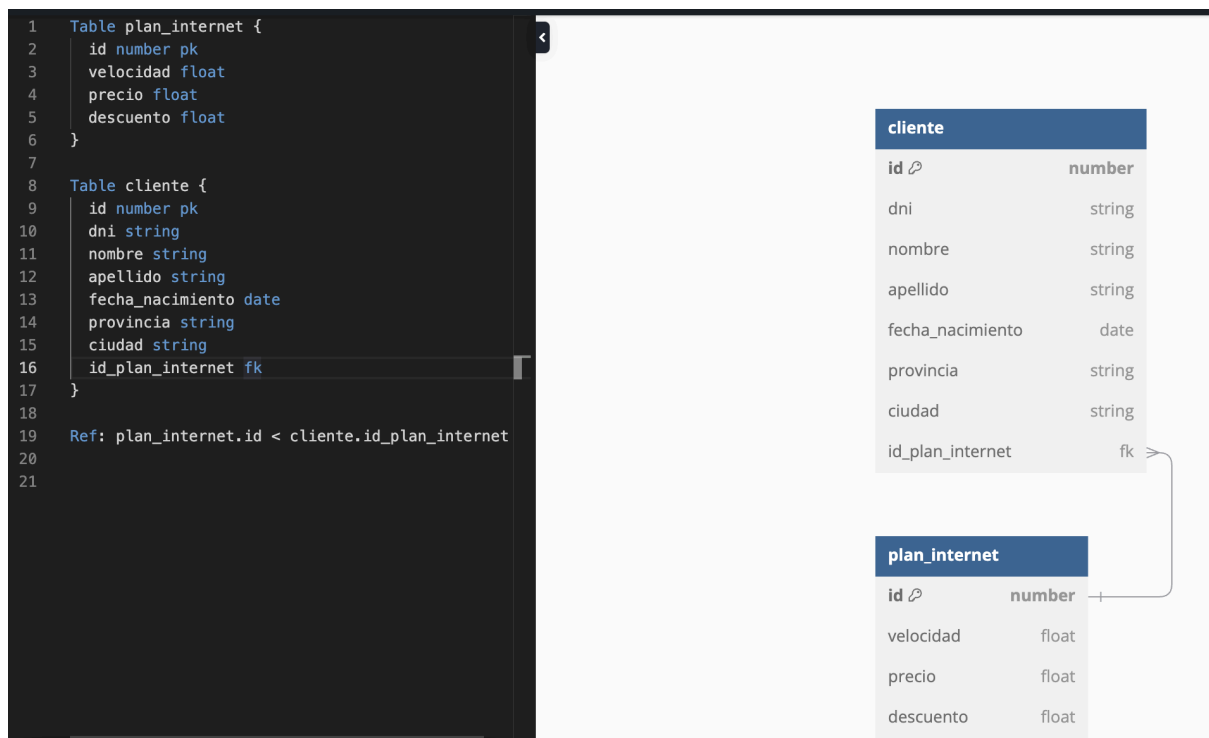
Mediante un análisis previo se conoce que se tiene que almacenar la siguiente información:

De los clientes se debe registrar: dni, nombre, apellido, fecha de nacimiento, provincia, ciudad.

En cuanto a los planes de internet: identificación del plan, velocidad ofrecida en megas, precio, descuento.

Ejercicio 1

Luego del planteo de los requerimientos de la empresa, se solicita modelar los mismos mediante un DER (Diagrama Entidad-Relación).





Ejercicio 2

Una vez modelada y planteada la base de datos, responder a las siguientes preguntas:

a. ¿Cuál es la primary key para la tabla de clientes? Justificar respuesta

La PK de la tabla clientes es el atributo “id” el DNI se puede repetir y puede presentar vulnerabilidades ante un id autoincremental.

b. ¿Cuál es la primary key para la tabla de planes de internet? Justificar respuesta.

La PK de la tabla planes_internet es el atributo “id” ya que no hay ningún otro campo con el cual crear una PK.

c. ¿Cómo serían las relaciones entre tablas? ¿En qué tabla debería haber foreign key? ¿A qué campo de qué tabla hace referencia dicha foreign key? Justificar respuesta.

Dado que consideramos que un cliente puede tener un solo plan de internet pero que a un plan lo puede tener muchos clientes se genera una relación ‘1 a n’ entre los planes de internet y los clientes.

De esta forma la FK queda del lado del cliente, en este caso el atributo ‘id_plan_internet’ hace referencia al ‘id’ de la tabla ‘planes_internet’.



Ejercicio 3

Una vez realizado el planteo del diagrama y de haber respondido estas preguntas, utilizar **PHPMyAdmin o MySQL Workbench** para ejecutar lo siguiente:

Se solicita crear una nueva base de datos llamada “**empresa_internet**”.

Incorporar 10 registros en la tabla de clientes y 5 en la tabla de planes de internet. Realizar las asociaciones/relaciones correspondientes entre estos registros.

```
CREATE DATABASE empresa_internet;

CREATE TABLE plan_internet(
    id int auto_increment PRIMARY KEY,
    velocidad float,
    precio float,
    descuento float
);

CREATE TABLE cliente(
    id int auto_increment PRIMARY KEY,
```

```

dni varchar(20),
nombre varchar(200),
apellido varchar(200),
fecha_nacimiento date,
provincia varchar(200),
ciudad varchar(200),
id_plan_internet int,
FOREIGN KEY (id_plan_internet) REFERENCES plan_internet(id)
);

INSERT INTO plan_internet (velocidad, precio, descuento) VALUES
(50, 29.99, 0),
(100, 39.99, 5),
(200, 49.99, 10),
(500, 69.99, 15),
(1000, 89.99, 20);

INSERT INTO cliente (dni, nombre, apellido, fecha_nacimiento, provincia, ciudad,
id_plan_internet) VALUES
('12345678A', 'Juan', 'Pérez', '1985-04-23', 'Madrid', 'Madrid', 1),
('23456789B', 'Maria', 'Gómez', '1978-07-19', 'Barcelona', 'Barcelona', 2),
('34567890C', 'Luis', 'Martínez', '1990-10-11', 'Valencia', 'Valencia', 3),
('45678901D', 'Ana', 'López', '1980-12-01', 'Sevilla', 'Sevilla', 1),
('56789012E', 'Pedro', 'Fernández', '1995-05-20', 'Valencia', 'Valencia', 2),
('67890123F', 'Laura', 'Sánchez', '1983-09-15', 'Sevilla', 'Sevilla', 4),
('78901234G', 'Eduardo', 'Ramírez', '1975-11-03', 'Sevilla', 'Sevilla', 5),
('89012345H', 'Sandra', 'Vidal', '1992-03-08', 'Valencia', 'Valencia', 3),
('90123456I', 'José', 'Gil', '1988-02-25', 'Barcelona', 'Barcelona', 4),
('98765432J', 'Beatriz', 'Pérez', '1986-06-12', 'Barcelona', 'Barcelona', 5);

```



Ejercicio 4

Plantear 10 consultas SQL que se podrían realizar a la base de datos. Expresar las sentencias.

Todos los clientes y su plan de internet

```

SELECT c.nombre, c.apellido, pi.velocidad, pi.precio
FROM cliente c
JOIN plan_internet pi ON c.id_plan_internet = pi.id;

```

Contar el número total de clientes por ciudad

```
SELECT ciudad, COUNT(*) AS total_clientes  
FROM cliente  
GROUP BY ciudad;
```

Seleccionar clientes mayores de 30 años

```
SELECT nombre, apellido, fecha_nacimiento  
FROM cliente  
WHERE DATE_PART('year', AGE(fecha_nacimiento)) > 30;
```

Obtener los planes de internet que tienen precio menor a 50.00

```
SELECT *  
FROM plan_internet  
WHERE precio < 50.00;
```

Seleccionar el cliente con el plan de internet más caro

```
SELECT c.nombre, c.apellido, pi.velocidad, pi.precio  
FROM cliente c  
JOIN plan_internet pi ON c.id_plan_internet = pi.id  
WHERE pi.precio = (SELECT MAX(precio) FROM plan_internet);
```

Listar todos los clientes de la provincia de 'Madrid'

```
SELECT *  
FROM cliente  
WHERE provincia = 'Madrid';
```

obtener el promedio de precios por planes de internet

```
SELECT AVG(precio) AS promedio_precio  
FROM plan_internet;
```

Contar cuantos planes de internet hay por velocidad de internet

```
SELECT velocidad, COUNT(*) AS total_planes  
FROM plan_internet
```

GROUP BY velocidad;

Seleccionar clientes que han recibido un descuento del plan

```
SELECT c.nombre, c.apellido, pi.descuento
FROM cliente c
JOIN plan_internet pi ON c.id_plan_internet = pi.id
WHERE pi.descuento > 0;
```

Listar el cliente más joven junto su plan de internet

```
SELECT c.nombre, c.apellido, c.fecha_nacimiento, pi.velocidad, pi.precio
FROM cliente c
JOIN plan_internet pi ON c.id_plan_internet = pi.id
ORDER BY c.fecha_nacimiento DESC
LIMIT 1;
```

```
-- Obtener los nombres y apellidos de los clientes que tienen planes con
velocidades superiores a 100 Mbps
SELECT c.nombre, c.apellido
FROM cliente c
JOIN plan_internet p ON c.id_plan_internet = p.id
WHERE p.velocidad > 100;

-- Encontrar el plan más común entre los clientes y mostrar su velocidad, precio y
descuento
SELECT p.velocidad, p.precio, p.descuento
FROM plan_internet p
JOIN cliente c ON p.id = c.id_plan_internet
GROUP BY p.id
ORDER BY COUNT(c.id) DESC
LIMIT 1;

-- Listar todos los nombres de clientes y el total de descuento que reciben con el
plan que tienen
SELECT c.nombre, c.apellido, p.descuento
FROM cliente c
JOIN plan_internet p ON c.id_plan_internet = p.id;
```

```
-- Obtener la ciudad con el menor número de clientes, junto con el conteo de esos
clientes
SELECT ciudad, COUNT(*) AS numero_clientes
FROM cliente
GROUP BY ciudad
ORDER BY numero_clientes ASC
LIMIT 1;
```