

Proyecto bases de datos

Base de datos de la formula 1



30 de mayo de 2022

Álvaro fraidias monteagudo

780336

Contenido

[1. Esquema conceptual 2](#_Toc104807093)

[2. Implementación 2](#_Toc104807094)

[3. Creación tablas 3](#_Toc104807095)

[4. Importación datos 7](#_Toc104807096)

[5. Consultas SQL 11](#_Toc104807097)



# 1. Esquema conceptual

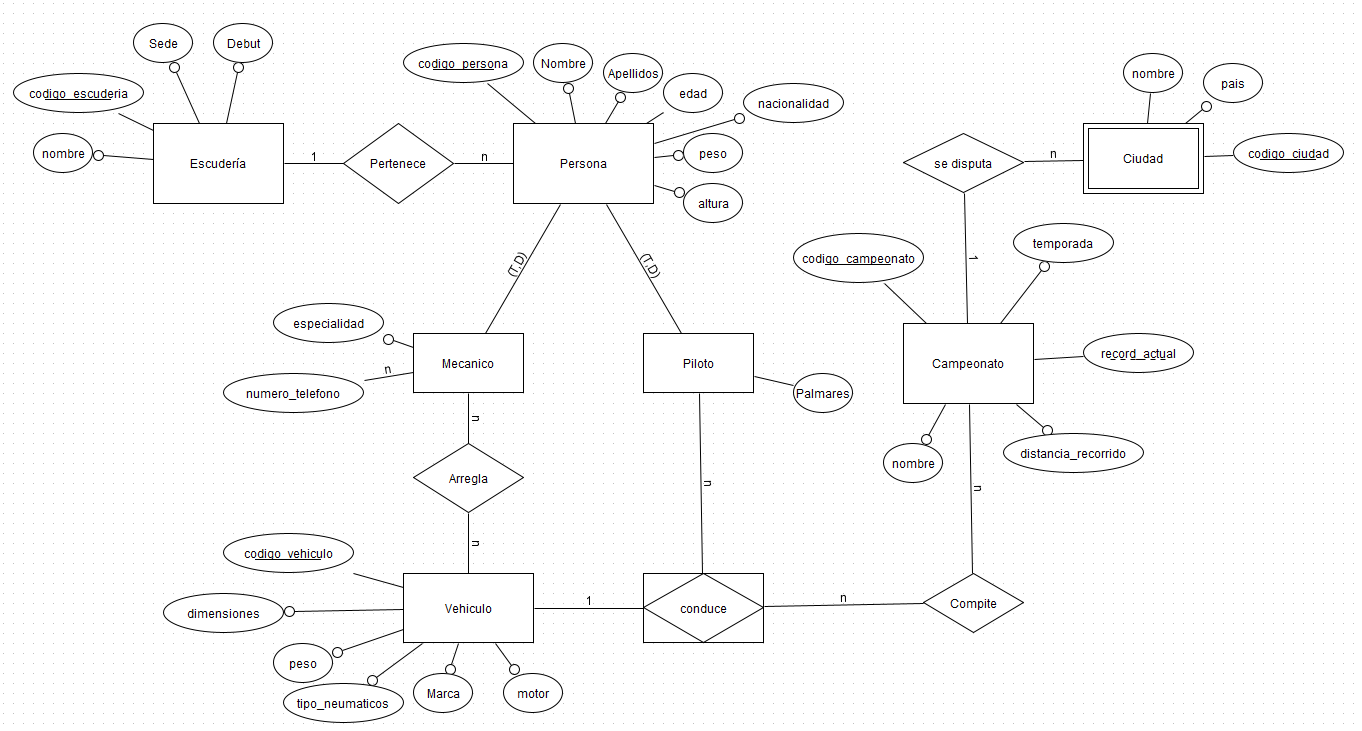


Ilustración : Esquema conceptual BD F1

Este esquema ya se explicó con mayor detenimiento en las prácticas anteriores. Es por ello que nos centraremos en su implementación física.

# 2. Implementación

A la hora de implementar esta base de datos he utilizado el SGBD PostgreSQL para aprovechar la máquina virtual proporcionada por la asignatura de Bases de Datos 2 así como el conocimiento que aprendemos de este SGBD en dicha asignatura.



Ilustración : Logo PostgreSQL

PostgreSQL, también llamado Postgres, es un sistema de gestión de bases de datos relacional orientado a objetos y de código abierto, publicado bajo la licencia PostgreSQL, similar a la BSD o la MIT.

# 3. Creación tablas

**--ELIMINACIÓN TABLAS--**

DROP TABLE IF EXISTS persona;

DROP TABLE IF EXISTS escuderia;

DROP TABLE IF EXISTS mecanico;

DROP TABLE IF EXISTS vehiculo;

DROP TABLE IF EXISTS arregla;

DROP TABLE IF EXISTS piloto;

DROP TABLE IF EXISTS campeonato;

DROP TABLE IF EXISTS compite;

**-- CREACION TABLAS --**

**-- TABLA ESCUDERIA--**

CREATE TABLE escuderia(

codigo\_escuderia INT NOT NULL,

nombre VARCHAR(200) NOT NULL,

sede VARCHAR(200)NOT NULL,

debut INT NOT NULL,

PRIMARY KEY (codigo\_escuderia)

);

**-- TABLA PERSONA --**

CREATE TABLE persona(

codigo\_persona INT,

nombre VARCHAR(200) NOT NULL,

apellidos VARCHAR(200)NOT NULL,

edad INT NOT NULL,

peso INT NOT NULL,

altura REAL NOT NULL,

nacionalidad VARCHAR(200)NOT NULL,

codigo\_escuderia INT REFERENCES escuderia,

PRIMARY KEY (codigo\_persona)

);

**-- TABLA VEHICULO--**

CREATE TABLE vehiculo(

codigo\_vehiculo INT NOT NULL,

dimensiones VARCHAR(200) NOT NULL,

peso INT NOT NULL,

tipo\_neumaticos VARCHAR(20) NOT NULL,

marca VARCHAR(200) NOT NULL,

motor VARCHAR(200) NOT NULL,

PRIMARY KEY (codigo\_vehiculo)

);

**--** **TABLA PILOTO--**

CREATE TABLE piloto(

codigo\_persona INT NOT NULL REFERENCES persona,

palmares INT NOT NULL,

codigo\_vehiculo INT NOT NULL REFERENCES vehiculo,

PRIMARY KEY (codigo\_persona)

);

**-- TABLA MECANICO--**

CREATE TABLE mecanico(

codigo\_persona INT NOT NULL REFERENCES persona,

especialidad VARCHAR(200) NOT NULL,

numero\_telefono INT,

PRIMARY KEY(codigo\_persona)

);

**-- TABLA ARREGLA--**

CREATE TABLE arregla(

codigo\_persona INT NOT NULL REFERENCES mecanico,

codigo\_vehiculo INT NOT NULL REFERENCES vehiculo,

PRIMARY KEY (codigo\_persona, codigo\_vehiculo)

);

**-- TABLA CAMPEONATO--**

CREATE TABLE campeonato(

codigo\_campeonato INT NOT NULL,

nombre VARCHAR(200) NOT NULL,

tipo VARCHAR(200) NOT NULL,

temporada VARCHAR(50) NOT NULL,

record\_actual REAL UNIQUE,

distancia\_recorrido REAL NOT NULL,

PRIMARY KEY (codigo\_campeonato)

);

**-- TABLA CIUDAD--**

CREATE TABLE ciudad(

codigo\_ciudad INT NOT NULL,

pais VARCHAR(200) NOT NULL,

nombre VARCHAR(200) NOT NULL,

codigo\_campeonato INT NOT NULL REFERENCES campeonato,

PRIMARY KEY (codigo\_ciudad, codigo\_campeonato)

);

**-- TABLA COMPITE--**

CREATE TABLE compite(

codigo\_campeonato INT NOT NULL REFERENCES campeonato,

codigo\_persona INT NOT NULL REFERENCES persona,

PRIMARY KEY (codigo\_campeonato, codigo\_persona)

);

# 4. Importación datos

**--INSERTAR VALORES TABLA ESCUDERIA**

INSERT INTO escuderia(codigo\_escuderia, nombre, sede, debut) VALUES (1, 'Ferrari', 'Maranello, Italia', 1950);

INSERT INTO escuderia(codigo\_escuderia, nombre, sede, debut) VALUES (2, 'Mclaren', 'Woking, Inglaterra', 1966);

INSERT INTO escuderia(codigo\_escuderia, nombre, sede, debut) VALUES (3, 'Mercedes', 'Stuttgart, Alemania', 1954);

INSERT INTO escuderia(codigo\_escuderia, nombre, sede, debut) VALUES (4, 'Aston Martin', 'Gaydon, Reino Unido', 1959);

INSERT INTO escuderia(codigo\_escuderia, nombre, sede, debut) VALUES (5, 'BMW', 'Múnich, Alemania', 1945);

INSERT INTO escuderia(codigo\_escuderia, nombre, sede, debut) VALUES (6, 'Alpine F1 Team', 'Enstone, Reino Unido', 1978);

**--INSERTAR VALORES TABLA VEHICULO**

INSERT INTO vehiculo(codigo\_vehiculo, dimensiones, peso, tipo\_neumaticos, marca,motor ) VALUES (1,'3 metros de largo y 2,8 metros de ancho', 899,'duros','Ferrari','V6 Turbo');

INSERT INTO vehiculo(codigo\_vehiculo, dimensiones, peso, tipo\_neumaticos, marca,motor ) VALUES (2,'3 metros de largo y 2,8 metros de ancho', 897,'mixtos', 'Mclaren','Mercedes-AMG M12 E Performance V6 Turbo');

INSERT INTO vehiculo(codigo\_vehiculo, dimensiones, peso, tipo\_neumaticos, marca,motor ) VALUES (3,'3 metros de largo y 2,8 metros de ancho', 898,'blandos', 'Mercedes','M09 EQ Power+ 1.6 L V6 Turbo');

INSERT INTO vehiculo(codigo\_vehiculo, dimensiones, peso, tipo\_neumaticos, marca,motor ) VALUES (4,'3 metros de largo y 2,8 metros de ancho', 899,'blandos', 'Mercedes','V6 Turbo');

**--INSERTAR VALORES TABLA PERSONA**

INSERT INTO persona(codigo\_persona, nombre, apellidos, edad, peso, altura, nacionalidad, codigo\_escuderia) VALUES (1, 'Carlos', 'Sainz', 27, 66, 1.77, 'España', 1);

INSERT INTO persona(codigo\_persona, nombre, apellidos, edad, peso, altura, nacionalidad, codigo\_escuderia) VALUES (2, 'Daniel', 'Ricciardo', 32, 66, 1.8, 'Australia', 2);

INSERT INTO persona(codigo\_persona, nombre, apellidos, edad, peso, altura, nacionalidad, codigo\_escuderia) VALUES (3, 'Lewis', 'Hamilton', 37, 73, 1.75, 'Reino Unido', 3);

INSERT INTO persona(codigo\_persona, nombre, apellidos, edad, peso, altura, nacionalidad, codigo\_escuderia) VALUES (4, 'Fernando', 'Alonso', 35, 74, 1.83, 'España', 6);

INSERT INTO persona(codigo\_persona, nombre, apellidos, edad, peso, altura, nacionalidad, codigo\_escuderia) VALUES (10, 'Andrea', 'De Adamich', 27, 73, 1.73, 'Italia', 1);

INSERT INTO persona(codigo\_persona, nombre, apellidos, edad, peso, altura, nacionalidad, codigo\_escuderia) VALUES (5, 'Mauricio', 'Colmenero', 35, 80, 1.68, 'España', 2);

INSERT INTO persona(codigo\_persona, nombre, apellidos, edad, peso, altura, nacionalidad, codigo\_escuderia) VALUES (6, 'Carlos', 'Catalan', 43, 72, 1.85, 'Alemania', 3);

INSERT INTO persona(codigo\_persona, nombre, apellidos, edad, peso, altura, nacionalidad, codigo\_escuderia) VALUES (7, 'Miguel', 'Martinez', 34, 70, 1.75, 'España', 3);

INSERT INTO persona(codigo\_persona, nombre, apellidos, edad, peso, altura, nacionalidad, codigo\_escuderia) VALUES (8, 'Clark', 'Steve', 40, 70, 1.7, 'Reino Unido', 1);

**--INSERTAR VALORES TABLA PILOTO**

INSERT INTO piloto(codigo\_persona, palmares, codigo\_vehiculo) VALUES (1, 0, 1);

INSERT INTO piloto(codigo\_persona, palmares, codigo\_vehiculo) VALUES (2, 8, 2);

INSERT INTO piloto(codigo\_persona, palmares, codigo\_vehiculo) VALUES (3, 103, 3);

INSERT INTO piloto(codigo\_persona, palmares, codigo\_vehiculo) VALUES (4, 23, 4);

INSERT INTO piloto(codigo\_persona, palmares, codigo\_vehiculo) VALUES (10, 0, 1);

**--INSERTAR VALORES TABLA MECANICO**

INSERT INTO mecanico(codigo\_persona, especialidad, numero\_telefono) VALUES (5, 'ruedas', 987654321);

INSERT INTO mecanico(codigo\_persona, especialidad, numero\_telefono) VALUES (6, 'amortiguacion', 122333444);

INSERT INTO mecanico(codigo\_persona, especialidad, numero\_telefono) VALUES (7, 'amortiguacion', 16798444);

INSERT INTO mecanico(codigo\_persona, especialidad, numero\_telefono) VALUES (8, 'motor', 123456789);

**--INSERTAR VALORES TABLA ARREGLA**

INSERT INTO arregla(codigo\_persona, codigo\_vehiculo) VALUES (5,2);

INSERT INTO arregla(codigo\_persona, codigo\_vehiculo) VALUES (6,3);

INSERT INTO arregla(codigo\_persona, codigo\_vehiculo) VALUES (8,1);

**--INSERTAR VALORES TABLA CAMPEONATO**

INSERT INTO campeonato(codigo\_campeonato, nombre, tipo, temporada, record\_actual, distancia\_recorrido) VALUES (1, 'Gilles Villeneuve', 'Híbrido', '2010-2019', 1.10240, 4.361);

INSERT INTO campeonato(codigo\_campeonato, nombre, tipo, temporada, record\_actual, distancia\_recorrido) VALUES (2, 'Hockenheimring', 'Autódromo', '2014, 2016, 2018-2019', 1.11212, 4.574);

INSERT INTO campeonato(codigo\_campeonato, nombre, tipo, temporada, record\_actual, distancia\_recorrido) VALUES (3, 'Hungaroring', 'Híbrido', '1986-Hoy', 1.13447, 4.381);

INSERT INTO campeonato(codigo\_campeonato, nombre, tipo, temporada, record\_actual, distancia\_recorrido) VALUES (4, 'Autodromo Nazionale di Monza', 'Híbrido', '1980-Hoy', 1.18887, 5.793);

INSERT INTO campeonato(codigo\_campeonato, nombre, tipo, temporada, record\_actual, distancia\_recorrido) VALUES (5, 'Indianápolis', 'Híbrido', '1977-Hoy', 1.28887, 5.893);

**--INSERTAR VALORES TABLA CIUDAD**

INSERT INTO ciudad(codigo\_ciudad, pais, nombre, codigo\_campeonato) VALUES (1, 'Canadá', 'Montreal',1);

INSERT INTO ciudad(codigo\_ciudad, pais, nombre, codigo\_campeonato) VALUES (2, 'Alemania', 'Hockenheim, Baden-Württemberg',2);

INSERT INTO ciudad(codigo\_ciudad, pais, nombre, codigo\_campeonato) VALUES (3, 'Hungria', 'Mogyoród',3);

INSERT INTO ciudad(codigo\_ciudad, pais, nombre, codigo\_campeonato) VALUES (4, 'Italia', 'Viale di Vedano',4);

**--INSERTAR VALORES TABLA COMPITE**

INSERT INTO compite(codigo\_campeonato, codigo\_persona) VALUES (1,1);

INSERT INTO compite(codigo\_campeonato, codigo\_persona) VALUES (1,2);

INSERT INTO compite(codigo\_campeonato, codigo\_persona) VALUES (1,3);

INSERT INTO compite(codigo\_campeonato, codigo\_persona) VALUES (1,4);

INSERT INTO compite(codigo\_campeonato, codigo\_persona) VALUES (1,5);

INSERT INTO compite(codigo\_campeonato, codigo\_persona) VALUES (1,6);

INSERT INTO compite(codigo\_campeonato, codigo\_persona) VALUES (2,1);

INSERT INTO compite(codigo\_campeonato, codigo\_persona) VALUES (2,2);

INSERT INTO compite(codigo\_campeonato, codigo\_persona) VALUES (2,3);

INSERT INTO compite(codigo\_campeonato, codigo\_persona) VALUES (2,4);

INSERT INTO compite(codigo\_campeonato, codigo\_persona) VALUES (2,5);

INSERT INTO compite(codigo\_campeonato, codigo\_persona) VALUES (2,6);

INSERT INTO compite(codigo\_campeonato, codigo\_persona) VALUES (3,1);

INSERT INTO compite(codigo\_campeonato, codigo\_persona) VALUES (3,2);

INSERT INTO compite(codigo\_campeonato, codigo\_persona) VALUES (3,3);

INSERT INTO compite(codigo\_campeonato, codigo\_persona) VALUES (3,4);

INSERT INTO compite(codigo\_campeonato, codigo\_persona) VALUES (3,5);

INSERT INTO compite(codigo\_campeonato, codigo\_persona) VALUES (3,6);

INSERT INTO compite(codigo\_campeonato, codigo\_persona) VALUES (3,8);

INSERT INTO compite(codigo\_campeonato, codigo\_persona) VALUES (4,1);

INSERT INTO compite(codigo\_campeonato, codigo\_persona) VALUES (4,2);

INSERT INTO compite(codigo\_campeonato, codigo\_persona) VALUES (4,3);

INSERT INTO compite(codigo\_campeonato, codigo\_persona) VALUES (4,10);

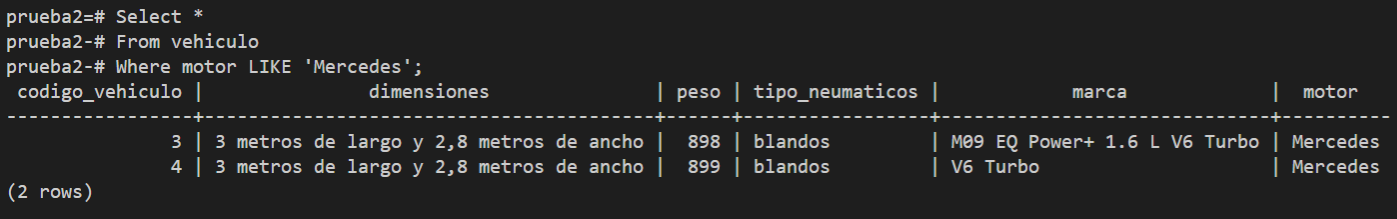
# 5. Consultas SQL

**1º Vehículo cuyo motor sea Mercedes**

Select \*

From vehiculo

Where motor LIKE 'Mercedes';



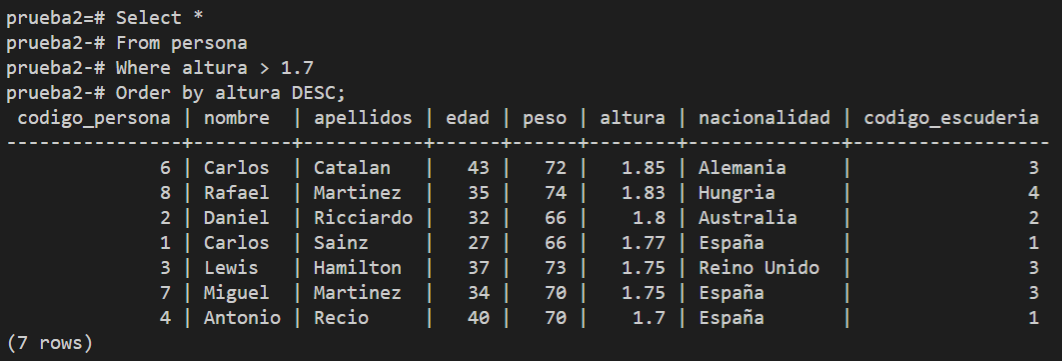
**2º Piloto con altura superior a 1,7 ordenados de más alto a más bajo.**

Select \*

From persona

Where altura > 1.7

Order by altura DESC;



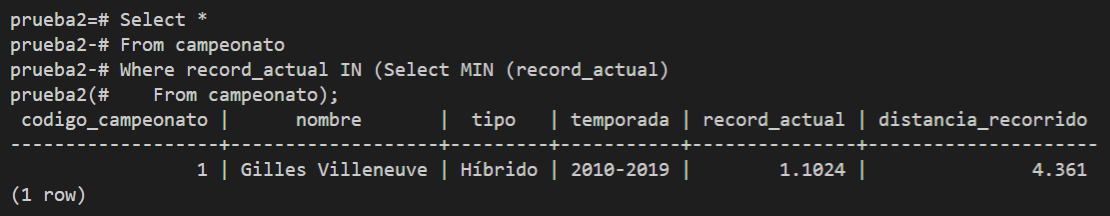
**3º Selecciona el circuito con mejor récord.**

Select \*

From campeonato

Where record\_actual IN (Select MIN (record\_actual)

From campeonato);



**4º Campeonato donde no haya pilotos**

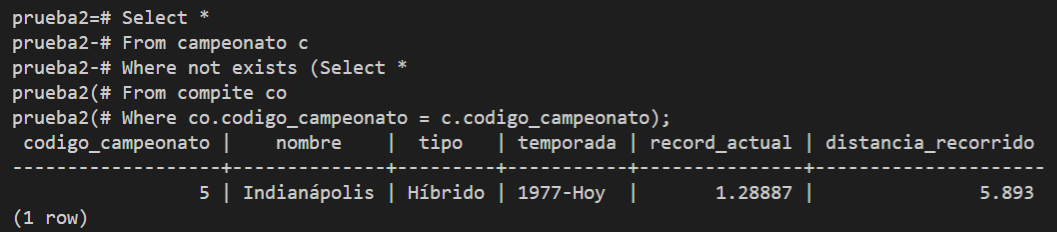
Select \*

From campeonato c

Where not exists (Select \*

From compite co

Where co.codigo\_campeonato = c.codigo\_campeonato);



**5º Escudería donde no haya pilotos**

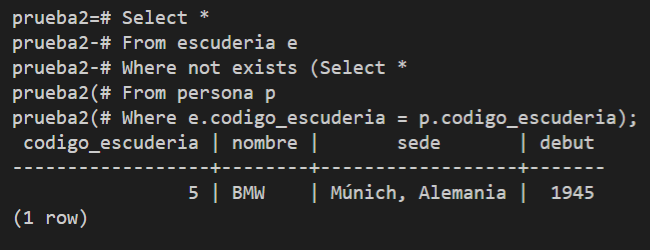
Select \*

From escuderia e

Where not exists (Select \*

From persona p

Where e.codigo\_escuderia = p.codigo\_escuderia);



**6º Piloto que participa en todos los campeonatos**

Para obtener los pilotos que participan en todos los campeonatos debemos relizar dos consultas:

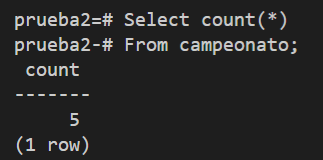
1º Para saber el número de campeonatos

2º Para saber si el piloto ha participado en el mismo número de campeonatos.

1º

Select count(\*)

From campeonato;



2º

Select pe.codigo\_persona, pe.nombre, pe.apellidos, count(\*)

From piloto p, compite c, campeonato ca, persona pe

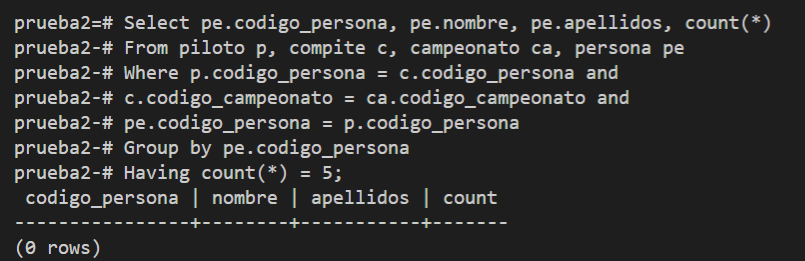
Where p.codigo\_persona = c.codigo\_persona and

c.codigo\_campeonato = ca.codigo\_campeonato and

pe.codigo\_persona = p.codigo\_persona

Group by pe.codigo\_persona

Having count(\*) = 5;



No hay ningún piloto que participe en todos los campeonatos ya que en el campeonato con nombre “Indianápolis” no hay Pilotos apuntados.

**7º Mecánico que arregla el coche Ferrari**

Select p.nombre, p.apellidos

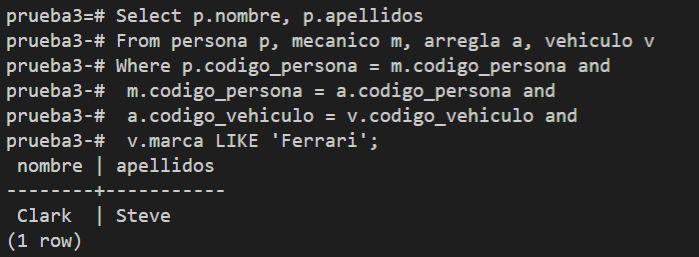
From persona p, mecanico m, arregla a, vehiculo v

Where p.codigo\_persona = m.codigo\_persona and

m.codigo\_persona = a.codigo\_persona and

a.codigo\_vehiculo = v.codigo\_vehiculo and

v.marca LIKE 'Ferrari';



**8º Piloto con mayor palmares**

Select p.nombre, p.apellidos, p1.palmares

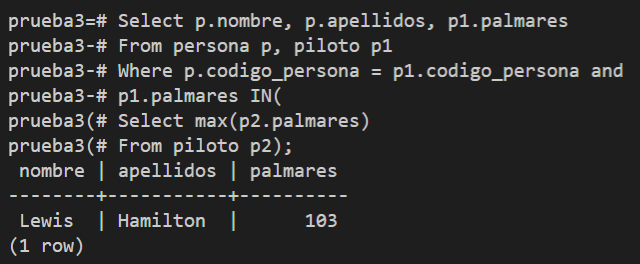
From persona p, piloto p1

Where p.codigo\_persona = p1.codigo\_persona and

p1.palmares IN(

Select max(p2.palmares)

From piloto p2);



**9º Piloto que use neumáticos duros, solo si es el único que usa este tipo de neumáticos**

Select pe.codigo\_persona, pe.nombre, pe.apellidos

From piloto p, persona pe

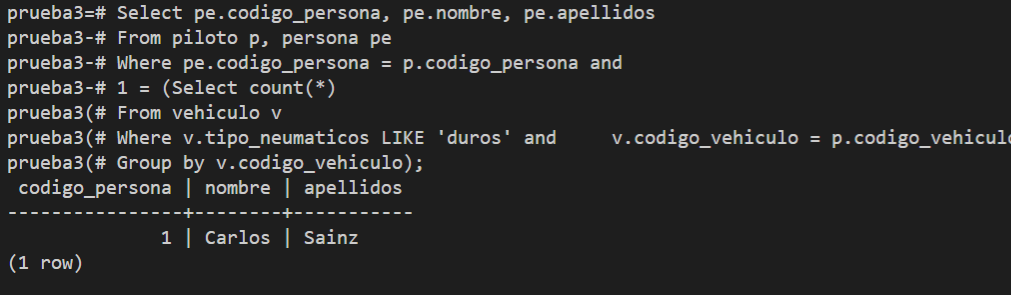
Where pe.codigo\_persona = p.codigo\_persona and

1 = (Select count(\*)

From vehiculo v

Where v.tipo\_neumaticos LIKE 'duros' and v.codigo\_vehiculo = p.codigo\_vehiculo

Group by v.codigo\_vehiculo);



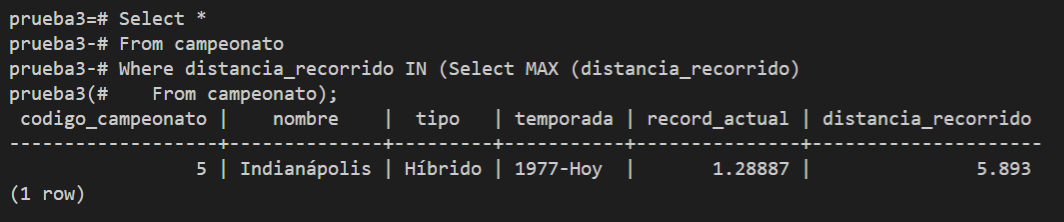
**10º Circuito con mayor distancia**

Select \*

From campeonato

Where distancia\_recorrido IN (Select MAX (distancia\_recorrido)

From campeonato);



**11º Nombre piloto que tienen un campeonato en su país**

Select pe.nombre, pe.apellidos, pe.nacionalidad, ci.nombre, ci.pais

From persona pe, piloto p, compite c, campeonato ca, ciudad ci

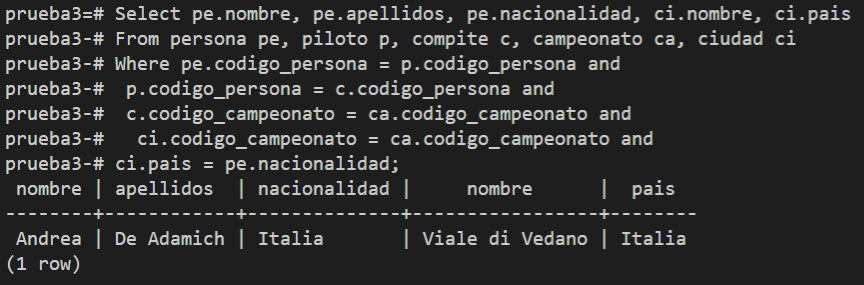
Where pe.codigo\_persona = p.codigo\_persona and

p.codigo\_persona = c.codigo\_persona and

c.codigo\_campeonato = ca.codigo\_campeonato and

ci.codigo\_campeonato = ca.codigo\_campeonato and

ci.pais = pe.nacionalidad;



**12º Mecánico que no arregla ningún vehículo**

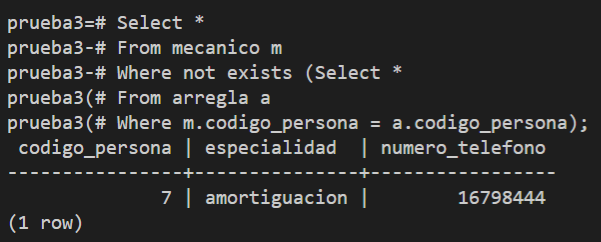
Select \*

From mecanico m

Where not exists (Select \*

From arregla a

Where m.codigo\_persona = a.codigo\_persona);



**13º Todas las escuderías que participan en algún campeonato**

Select e.nombre

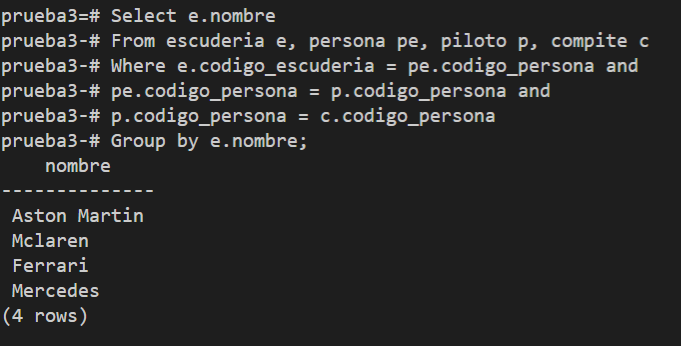
From escuderia e, persona pe, piloto p, compite c

Where e.codigo\_escuderia = pe.codigo\_persona and

pe.codigo\_persona = p.codigo\_persona and

p.codigo\_persona = c.codigo\_persona

Group by e.nombre;



**14º Todas las escuderías con pilotos que no participan en ningún campeonato**

Select e.nombre, e.sede

From escuderia e, persona pe

Where e.codigo\_escuderia = pe.codigo\_escuderia and

pe.codigo\_persona IN (

Select p.codigo\_persona

From piloto p

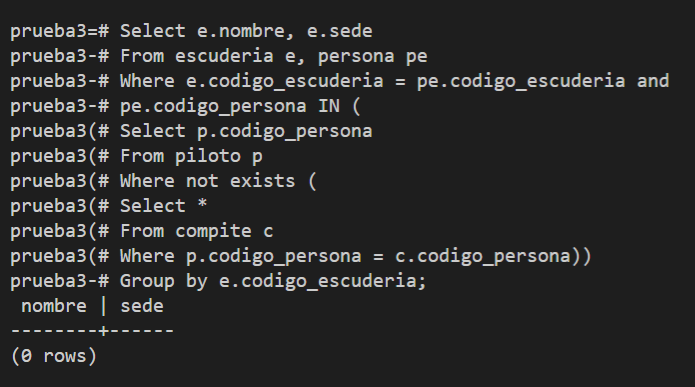
Where not exists (

Select \*

From compite c

Where p.codigo\_persona = c.codigo\_persona))

Group by e.codigo\_escuderia;



**14º Todos los vehículos que participan en más de dos campeonatos**

Select v.codigo\_vehiculo, v.motor, v.marca

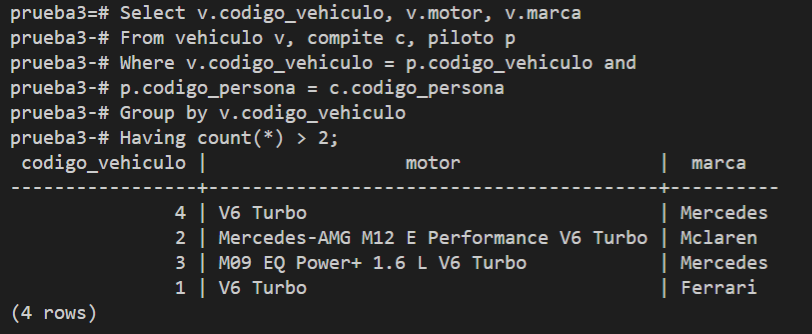
From vehiculo v, compite c, piloto p

Where v.codigo\_vehiculo = p.codigo\_vehiculo and

p.codigo\_persona = c.codigo\_persona

Group by v.codigo\_vehiculo

Having count(\*) > 2;

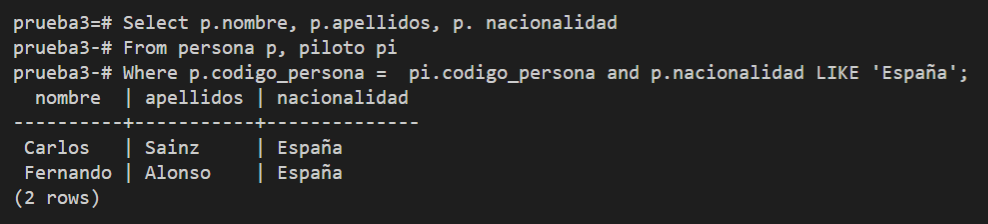


**15º Pilotos españoles**

Select p.nombre, p.apellidos, p. nacionalidad

From persona p, piloto pi

Where p.codigo\_persona = pi.codigo\_persona and p.nacionalidad LIKE 'España';



**16º Pilotos que conducen mercedes**

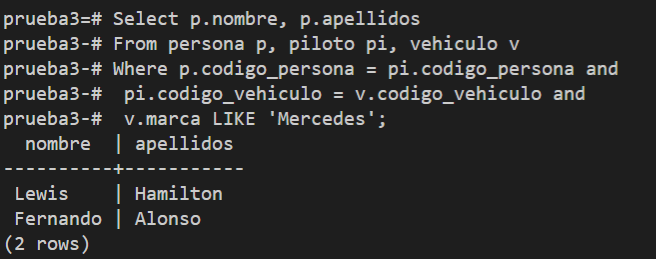
Select p.nombre, p.apellidos

From persona p, piloto pi, vehiculo v

Where p.codigo\_persona = pi.codigo\_persona and

pi.codigo\_vehiculo = v.codigo\_vehiculo and

v.marca LIKE 'Mercedes';



**17º Mecánicos que reparan mercedes**

Select p.nombre, p.apellidos

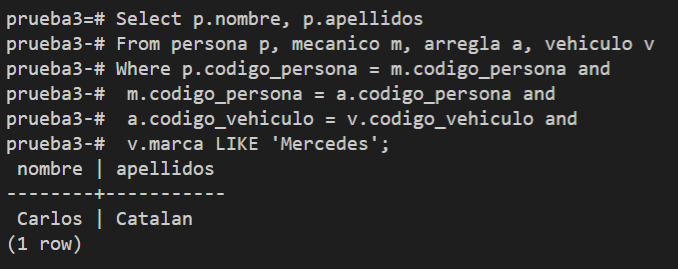
From persona p, mecanico m, arregla a, vehiculo v

Where p.codigo\_persona = m.codigo\_persona and

m.codigo\_persona = a.codigo\_persona and

a.codigo\_vehiculo = v.codigo\_vehiculo and

v.marca LIKE 'Mercedes';



**18º Escuderias sin pilotos**

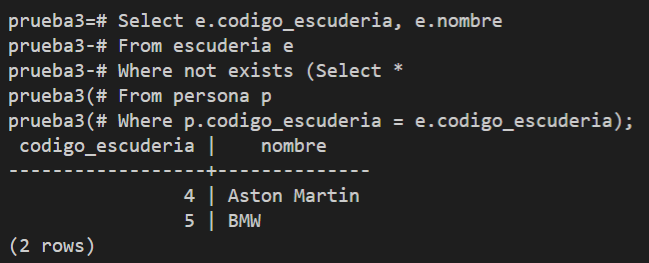
Select e.codigo\_escuderia, e.nombre

From escuderia e

Where not exists (Select \*

From persona p

Where p.codigo\_escuderia = e.codigo\_escuderia);



**19º Pilotos que no compiten en ningún campeonato**

Select pe.nombre, pe.apellidos

From persona pe

Where pe.codigo\_persona IN (

Select p.codigo\_persona

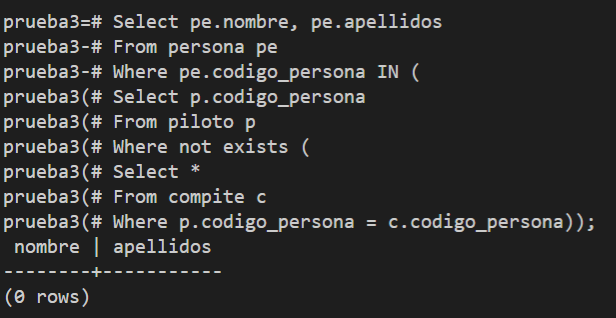
From piloto p

Where not exists (

Select \*

From compite c

Where p.codigo\_persona = c.codigo\_persona));



**20º Pilotos sin escudería**

Select p.nombre, p.apellidos

From persona p, piloto pi

Where pi.codigo\_persona = p.codigo\_persona and p.codigo\_escuderia is null;

