BBDD F1



***url Github***

***https://github.com/Abraham-HUB-777/Tarefa-Final-BD-1-DAM***

**Nome Alumno/a: Abraham Fernández Bande**

***Abraham Fernandez Bande***

**Curso: *1º DAM* Materia:** ***Bases de Datos – Proyecto Final 24/25***

Contido

[1. Introducción 2](#_Toc197794426)

[2. Descripción del Problema / Requisitos 2](#_Toc197794427)

[3. Modelo Conceptual 2](#_Toc197794428)

[4. Modelo Relacional 2](#_Toc197794429)

[5. Proceso de Normalización 2](#_Toc197794430)

[6. Script de Creación de la Base de Datos 2](#_Toc197794431)

[7. Carga de Datos Inicial 2](#_Toc197794432)

[8. Funciones y Procedimientos Almacenados 2](#_Toc197794433)

[9. Triggers 2](#_Toc197794434)

[10. Consultas SQL 2](#_Toc197794435)

[11. Casos de Prueba y Simulación 2](#_Toc197794436)

[12. Resultados y Verificación 2](#_Toc197794437)

[13. Capturas de Pantalla (opcional) 2](#_Toc197794438)

[14. Conclusiones y Mejoras Futuras 2](#_Toc197794439)

[15. Enlace al Repositorio en GitHub 3](#_Toc197794440)

# Introducción

Necesidad del cliente:

 Nuestro negocio se centra en proporcionar información precisa y exhaustiva sobre cada Gran Premio, incluyendo resultados, tiempos de vuelta, posiciones en parrilla, estrategias de carrera y estadísticas de pilotos y equipos.

# Descripción del Problema / Requisitos

Para crear esta BD necesitaremos tablas para almacenar

**- Piloto**(id, nombre, abreviatura, nacionalidad, altura, peso, fecha\_nacimiento, numero\_licencia, foto\_url);

**- Equipo**(id, nombre, pais, color, sede\_principal, director\_equipo, año\_fundacion, logo\_url);

**- Temporada**(id, año, idPilotoGANADOR, idEquipoGANADOR);

**- CocheTemporada**(id, nombre, motor, idEquipo, idTemporada, especificaciones\_tecnicas, peso\_coche, potencia);

**- PilotoTemporadaEquipo**(id, idTemporada, idPiloto, idEquipo, idCocheTemporada, numeroCoche);

**- Circuito**(id, nombre, localizacion, pais, kilometros, numero\_vueltas, record\_vuelta, tipo\_circuito, longitud\_pit\_lane);

**- GranPremio**(id, nombreGranPremio, idTemporada, idCircuito, fecha, clima, hora\_inicio, vueltas\_completadas, estado);

**- ClasificacionPiloto**(id, idTemporada, idPiloto, posicion, puntos);

**- ClasificacionEquipo**(id, idTemporada, idEquipo, posicion, puntos);

**- Qualy**(id, idGranPremio, idPiloto, idEquipo, q1, q2, q3, posicion, tipo\_clasificacion, condicion\_pista);

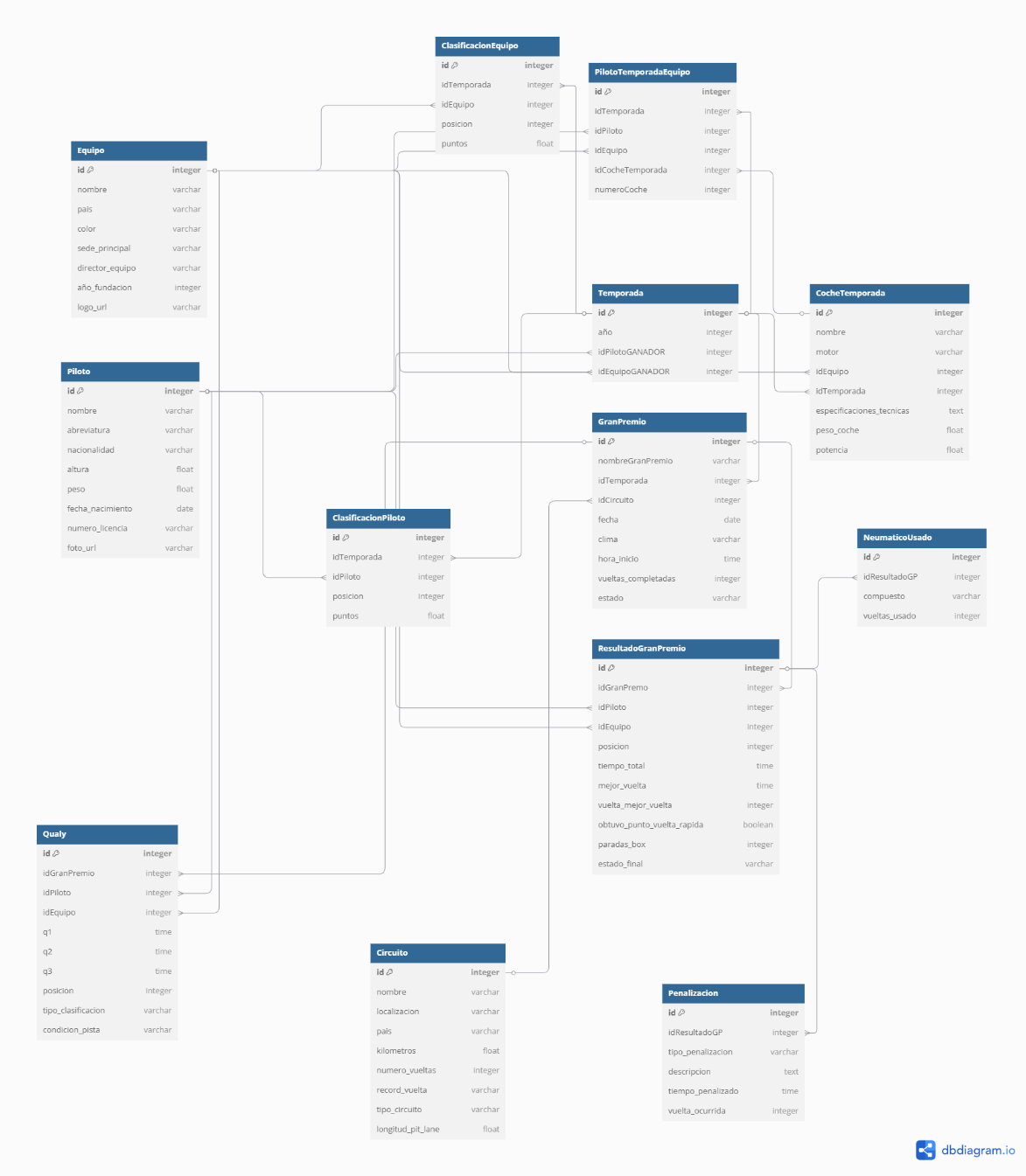
**- ResultadoGranPremio**(id, idGranPremo, idPiloto, idEquipo, posicion, tiempo\_total, mejor\_vuelta, vuelta\_mejor\_vuelta, obtuvo\_punto\_vuelta\_rapida, paradas\_box, estado\_final);

**- NeumaticoUsado**(id, idResultadoGP, compuesto, vueltas\_usado);

**- Penalizacion**(id, idResultadoGP, tipo\_penalizacion, descripcion, tiempo\_penalizado, vuelta\_ocurrida)

# Modelo Conceptual C:\Users\abraham\Downloads\Mapa conceptual (1).png

# Modelo Relacional



# Proceso de Normalización

Paso 1:

* Todas las tablas cumplen con el 1fn (No hay campos multivaluados)

**Piloto**(id, nombre, abreviatura, edad, nacionalidad, altura, peso, fecha\_nacimiento, numero\_licencia, foto\_url);

**Piloto(id, nombre, abreviatura, nacionalidad, altura, peso, fecha\_nacimiento, numero\_licencia, foto\_url).**

Vamos a eliminar edad ya que puede ser calculada con fecha Nacimiento.

Paso 2:

* Todas las tablas cumplen con 2FN (No hay dependencias de claves compuestas)

Paso 3:

# Script de Creación de la Base de Datos

drop database if exists Formula1;

create database if not exists Formula1;

use Formula1;

CREATE TABLE Piloto (

id INT AUTO\_INCREMENT PRIMARY KEY,

nombre VARCHAR(255),

abreviatura VARCHAR(50),

nacionalidad VARCHAR(100),

altura FLOAT,

peso FLOAT,

fecha\_nacimiento DATE,

numero\_licencia VARCHAR(50),

foto\_url VARCHAR(255)

);

CREATE TABLE Equipo (

id INT AUTO\_INCREMENT PRIMARY KEY,

nombre VARCHAR(255),

pais VARCHAR(100),

color VARCHAR(50),

sede\_principal VARCHAR(255),

director\_equipo VARCHAR(255),

año\_fundacion INT,

logo\_url VARCHAR(255)

);

CREATE TABLE Temporada (

id INT AUTO\_INCREMENT PRIMARY KEY,

año INT,

idPilotoGANADOR INT,

idEquipoGANADOR INT,

CONSTRAINT fk\_temporada\_piloto FOREIGN KEY (idPilotoGANADOR) REFERENCES Piloto(id),

CONSTRAINT fk\_temporada\_equipo FOREIGN KEY (idEquipoGANADOR) REFERENCES Equipo(id)

);

CREATE TABLE CocheTemporada (

id INT AUTO\_INCREMENT PRIMARY KEY,

nombre VARCHAR(255),

motor VARCHAR(255),

idEquipo INT,

idTemporada INT,

especificaciones\_tecnicas TEXT,

peso\_coche FLOAT,

potencia FLOAT,

CONSTRAINT fk\_coche\_equipo FOREIGN KEY (idEquipo) REFERENCES Equipo(id),

CONSTRAINT fk\_coche\_temporada FOREIGN KEY (idTemporada) REFERENCES Temporada(id)

);

CREATE TABLE PilotoTemporadaEquipo (

id INT AUTO\_INCREMENT PRIMARY KEY,

idTemporada INT,

idPiloto INT,

idEquipo INT,

idCocheTemporada INT,

numeroCoche INT,

CONSTRAINT fk\_pte\_temporada FOREIGN KEY (idTemporada) REFERENCES Temporada(id),

CONSTRAINT fk\_pte\_piloto FOREIGN KEY (idPiloto) REFERENCES Piloto(id),

CONSTRAINT fk\_pte\_equipo FOREIGN KEY (idEquipo) REFERENCES Equipo(id),

CONSTRAINT fk\_pte\_coche FOREIGN KEY (idCocheTemporada) REFERENCES CocheTemporada(id)

);

CREATE TABLE Circuito (

id INT AUTO\_INCREMENT PRIMARY KEY,

nombre VARCHAR(255),

localizacion VARCHAR(255),

pais VARCHAR(100),

kilometros FLOAT,

numero\_vueltas INT,

record\_vuelta VARCHAR(50),

tipo\_circuito VARCHAR(100),

longitud\_pit\_lane FLOAT

);

CREATE TABLE GranPremio (

id INT AUTO\_INCREMENT PRIMARY KEY,

nombreGranPremio VARCHAR(255),

idTemporada INT,

idCircuito INT,

fecha DATE,

clima VARCHAR(100),

hora\_inicio TIME,

vueltas\_completadas INT,

estado VARCHAR(100),

CONSTRAINT fk\_gp\_temporada FOREIGN KEY (idTemporada) REFERENCES Temporada(id),

CONSTRAINT fk\_gp\_circuito FOREIGN KEY (idCircuito) REFERENCES Circuito(id)

);

CREATE TABLE ClasificacionPiloto (

id INT AUTO\_INCREMENT PRIMARY KEY,

idTemporada INT,

idPiloto INT,

posicion INT,

puntos FLOAT,

CONSTRAINT fk\_cp\_temporada FOREIGN KEY (idTemporada) REFERENCES Temporada(id),

CONSTRAINT fk\_cp\_piloto FOREIGN KEY (idPiloto) REFERENCES Piloto(id)

);

CREATE TABLE ClasificacionEquipo (

id INT AUTO\_INCREMENT PRIMARY KEY,

idTemporada INT,

idEquipo INT,

posicion INT,

puntos FLOAT,

CONSTRAINT fk\_ce\_temporada FOREIGN KEY (idTemporada) REFERENCES Temporada(id),

CONSTRAINT fk\_ce\_equipo FOREIGN KEY (idEquipo) REFERENCES Equipo(id)

);

CREATE TABLE Qualy (

id INT AUTO\_INCREMENT PRIMARY KEY,

idGranPremio INT,

idPiloto INT,

idEquipo INT,

q1 TIME,

q2 TIME,

q3 TIME,

posicion INT,

tipo\_clasificacion VARCHAR(100),

condicion\_pista VARCHAR(100),

CONSTRAINT fk\_qualy\_gp FOREIGN KEY (idGranPremio) REFERENCES GranPremio(id),

CONSTRAINT fk\_qualy\_piloto FOREIGN KEY (idPiloto) REFERENCES Piloto(id),

CONSTRAINT fk\_qualy\_equipo FOREIGN KEY (idEquipo) REFERENCES Equipo(id)

);

CREATE TABLE ResultadoGranPremio (

id INT AUTO\_INCREMENT PRIMARY KEY,

idGranPremo INT,

idPiloto INT,

idEquipo INT,

posicion INT,

tiempo\_total TIME,

mejor\_vuelta TIME,

vuelta\_mejor\_vuelta INT,

obtuvo\_punto\_vuelta\_rapida BOOLEAN,

paradas\_box INT,

estado\_final VARCHAR(100),

CONSTRAINT fk\_resultado\_gp FOREIGN KEY (idGranPremo) REFERENCES GranPremio(id),

CONSTRAINT fk\_resultado\_piloto FOREIGN KEY (idPiloto) REFERENCES Piloto(id),

CONSTRAINT fk\_resultado\_equipo FOREIGN KEY (idEquipo) REFERENCES Equipo(id)

);

CREATE TABLE NeumaticoUsado (

id INT AUTO\_INCREMENT PRIMARY KEY,

idResultadoGP INT,

compuesto VARCHAR(100),

vueltas\_usado INT,

CONSTRAINT fk\_neumatico\_resultado FOREIGN KEY (idResultadoGP) REFERENCES ResultadoGranPremio(id)

);

CREATE TABLE Penalizacion (

id INT AUTO\_INCREMENT PRIMARY KEY,

idResultadoGP INT,

tipo\_penalizacion VARCHAR(100),

descripcion TEXT,

tiempo\_penalizado TIME,

vuelta\_ocurrida INT,

CONSTRAINT fk\_penalizacion\_resultado FOREIGN KEY (idResultadoGP) REFERENCES ResultadoGranPremio(id)

);

# Carga de Datos Inicial

Describe aquí...

# Funciones y Procedimientos Almacenados

Describe aquí...

# Triggers

Describe aquí...

# Consultas SQL

Describe aquí...

# Casos de Prueba y Simulación

Describe aquí...

# Resultados y Verificación

# Capturas de Pantalla (opcional)

Describe aquí...

# Conclusiones y Mejoras Futuras

Describe aquí...

# Enlace al Repositorio en GitHub

Describe aquí..