**Documentação Projeto Integrador 3**

**Grupo:** Davi Barbosa, Eduardo Rizzi, Gabriel Alves, Joaquim.

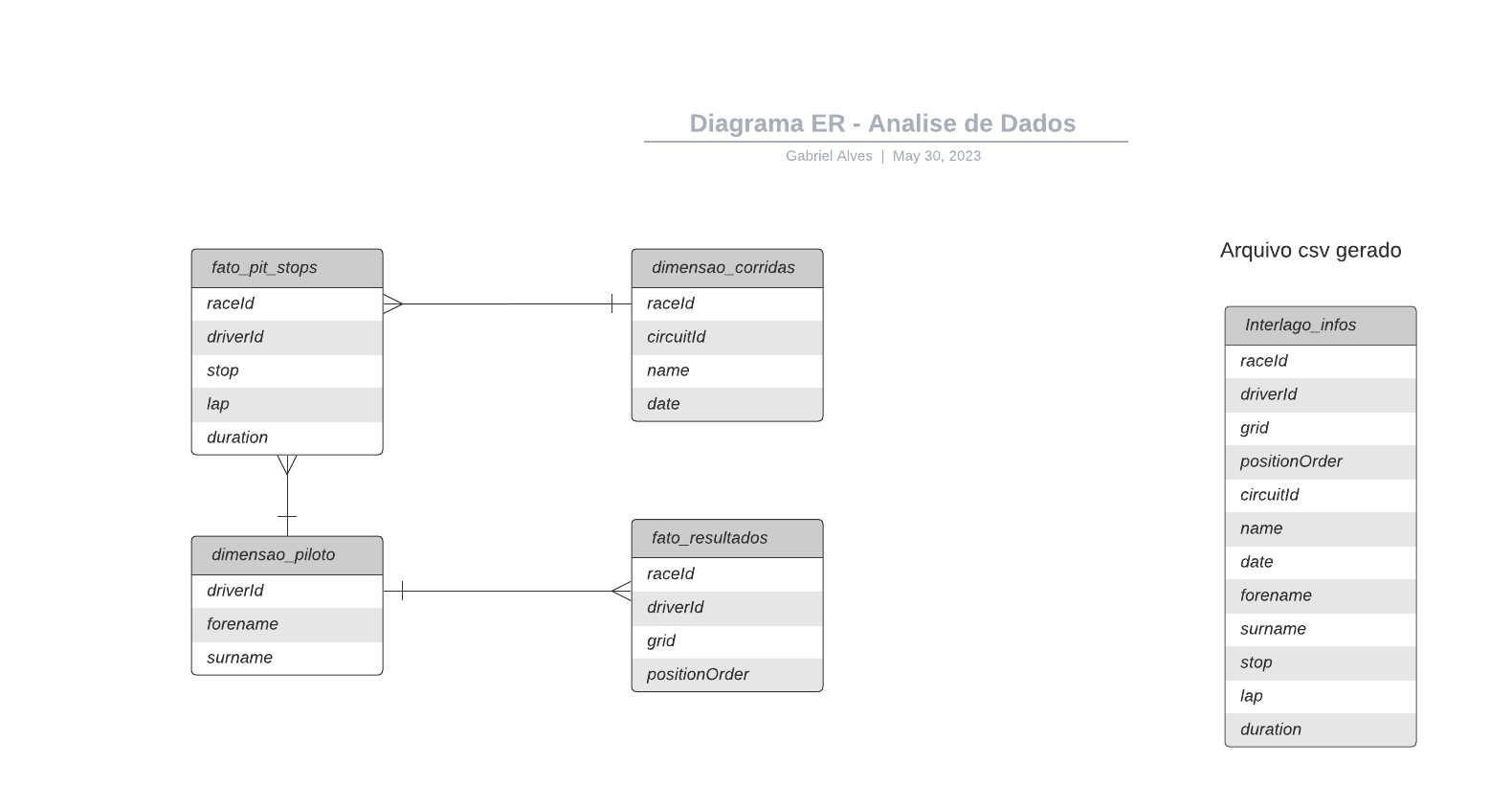
Nosso projeto será uma análise de corridas de formula 1 desde 1950 até 2023. O ETL foi feito no Jupyter notebook, utilizando bibliotecas para lidar com DataFrames e facilitar a conexão com o banco de dados.

O foco do projeto é prever possíveis respostas para a seguinte pergunta:  
**“É possível prever o próximo campeão da formula 1 com base em dados categóricos?”**

**OBS: Filtrado por um só circuito, sendo o escolhido para o projeto o circuito de Interlagos.**

**Iremos extrair as fontes do Kaggle:** <https://www.kaggle.com/datasets/rohanrao/formula-1-world-championship-1950-2020>.

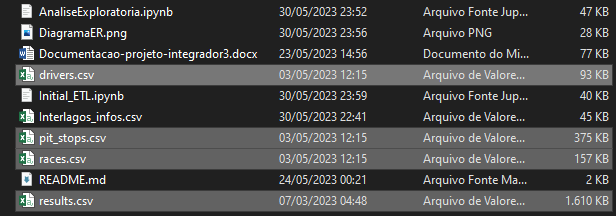
**Diagrama ER:**

****

Dessa base de dados do Kaggle, utilizaremos 4 arquivos, sendo eles ***drivers.csv***, ***pit\_stops.csv***, ***races.csv***, ***results.csv***.

Para fazer o ETL, é necessário ter esses arquivos baixados e salvos no mesmo diretório do arquivo principal (Initial\_ETL.ipynb).

**Exemplo:**



Utilizamos docker para lidar iniciar a conexão com o banco de dados localmente.

O banco de dados escolhido foi o Postgres.

As tabelas fato e dimensão foram geradas em um database chamado ‘dbanalise’

Na ETL, fizemos uma query no banco de dados (que contém os dataframes citados anteriormente e filtrados apenas com as colunas necessárias) para retornar apenas um dataframe.

**Query**:

***SELECT \****

***FROM fato\_resultados***

***INNER JOIN dimensao\_corridas***

***ON fato\_resultados."raceId" = dimensao\_corridas."raceId"***

***INNER JOIN dimensao\_pilotos***

***ON dimensao\_pilotos."driverId" = fato\_resultados."driverId"***

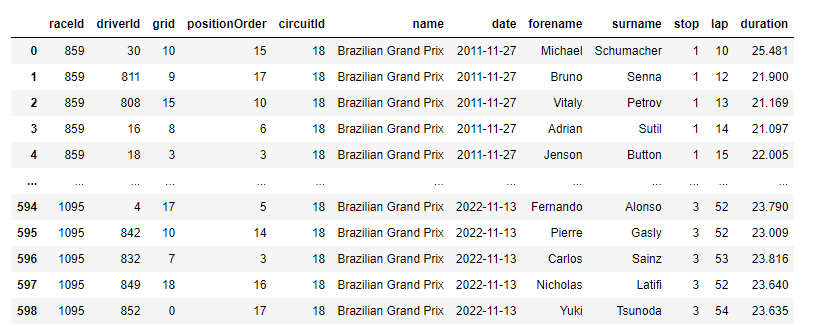
***INNER JOIN fato\_pit\_stops***

***ON fato\_resultados."raceId" = fato\_pit\_stops."raceId"***

***AND***

***fato\_resultados."driverId" = fato\_pit\_stops."driverId"***

**WHERE dimensao\_corridas."circuitId" = 18**

Após isso, recuperamos uma tabela mais ou menos assim:

E então a salvamos em um arquivo .csv para trabalharmos somente nele para os próximos passos.

No arquivo **AnaliseExploratoria.ipynb** começamos a fazer uma análise mais profunda sobre os dados obtidos, como média de duração de pit\_stops e desvio padrão

**DICIONARIO DE DADOS:**

|  |  |  |
| --- | --- | --- |
| DADO | TIPO DE DADO | DESCRIÇÃO |
| circuitId | Int64 | Número de identificação do circuito. |
| circuitRef | Object | Nome de referência do circuito. |
| name | Object | Atual nome do circuito. |
| location | Object | Cidade do circuito. |
| country | Object | País do circuito. |
| lat | Float64 | Latitude do circuito. |
| lng | Float64 | Longitude do circuito. |
| alt | Object | Altitude do circuito. |
| driverId | Int6 | Número de identificação do piloto. |
| driverRef | Object | Nome de referência do piloto. |
| number | Object | Número do piloto. |
| code | Object | Código de 3 letras do alfabeto representando o piloto. |
| forename | Object | Primeiro nome do piloto. |
| surname | Object | Último nome do piloto. |
| dob | Object | Data de nascimento do piloto. |
| nationality | Object | Nacionalidade do piloto. |
| driverStandingsId | Int64 | Número de identificação da classificação do piloto. |
| raceId | Int64 | Número de identificação da corrida. |
| points | Float64 | Pontuação do piloto. |
| position | Int64 | Posição final do piloto. |
| positionText | Object | Posição final do piloto em texto. |
| wins | Int64 | Número de vitórias do piloto. |
| year | Int64 | Ano da corrida. |
| round | Int64 | Número da corrida. |
| date | Object | Data da corrida. |
| time | Object | Tempo da corrida. |
| stop | Int64 | Número do pit stop. |
| lap | Int64 | Número da volta. |
| duration | Object | Duração do pit stop. |
| milliseconds | Int64 | Duração do pit stop em milissegundos. |