



# Modelagem e análise de dados da F1

## dataWizards

Abílio Nogueira

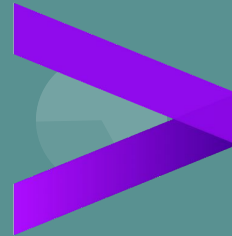
Alana Vieira

Alano Coelho Mauriz Menezes

Antônio Carlos Sandoque dos Santos

Breno Jones Agrelli Dias da Silva

Bruno Ferreira da Silva





# Divisão das atividades

Não houve definição formal de tarefas. Cada membro desenvolvia soluções de forma independente e frequentemente nos reuníamos para construir o projeto. No entanto, alguns membros foram mais proativos e contribuíram mais em determinadas etapas do projeto:

Modelagem Transacional	Modelagem Multidimensional	Análise dos dados no Python
Alano	Alano	Abílio
Breno	Antônio	Breno
	Breno	



# Etapas do projeto

## 1. Modelagem Relacional/Transacional

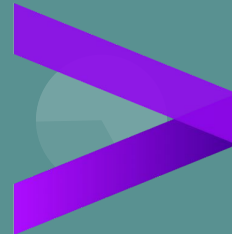
- a. Diagrama Entidade-Relacionamento
- b. Criação do Banco de dados (DDL)
- c. Povoamento do Banco de Dados (DML)

## 2. Modelagem Multidimensional

- a. Diagrama Entidade-Relacionamento (Fatos e Dimensões)
- b. Criação do DW
- c. Tratamento de dados e povoamento do DW
- d. Exportação dos dados para análise

## 3. Análise dos dados em Python

- a. Criação do Notebook
- b. Importação de bibliotecas e bases
- c. Desenvolvimento
- d. Apresentação dos resultados





# 1) Modelagem Transacional

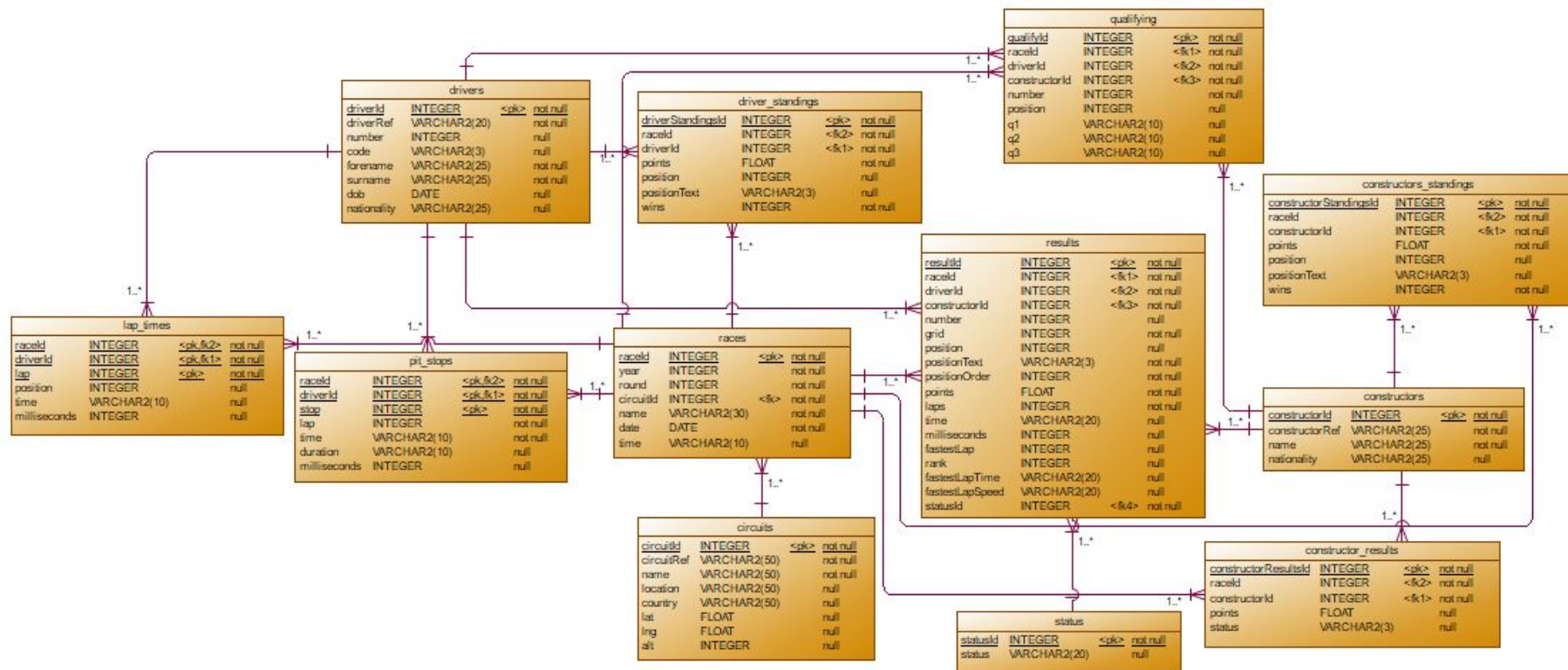
Ferramentas utilizadas:

- SAP Powerdesigner: Utilizado na criação do DER
- Oracle Cloud 19c Autonomous Database (Free trial 30 dias)
- Oracle SQL Developer (ambiente de desenvolvimento)
- Notepad ++ e Excel

O que foi feito:

- Definição das entidades (tabelas), atributos (colunas) e relacionamentos (integridade referencial, cardinalidades)
- Criação das tabelas no banco através de scripts DDL
- Povoamento do banco através de scripts DML e ferramentas do SGBD.

# Diagrama Entidade Relacionamento Transaccional





# Script DDL

No PowerDesigner:

- Database > Generate Database...
- Script gerado automaticamente pelo software levando em conta todas as definições feitas na modelagem.
- Instaciado um banco de dados no Oracle Cloud
- Estabelecimento da conexão com o Oracle SQL Developer através da criação de uma Wallet.

# Script DDL

```
create table CIRCUITS (
  CIRCUITID          INTEGER          not null,
  CIRCUITREF         VARCHAR2(30)     not null,
  NAME               VARCHAR2(50)     not null,
  LOCATION           VARCHAR2(30)     not null,
  COUNTRY            VARCHAR2(30)     not null,
  LAT                NUMBER(9,6)      not null,
  LNG                NUMBER(9,6)      not null,
  ALT                INTEGER          not null,
  constraint PK_CIRCUITS primary key (CIRCUITID)
);

/*=====*/
/* Table: CONSTRUCTORS */
/*=====*/

create table CONSTRUCTORS (
  CONSTRUCTORID      INTEGER          not null,
  CONSTRUCTORREF     VARCHAR2(25)     not null,
  NAME               VARCHAR2(25)     not null,
  NATIONALITY        VARCHAR2(25)     not null,
  constraint PK_CONSTRUCTORS primary key (CONSTRUCTORID)
);

/*=====*/
/* Table: CONSTRUCTORS_STANDINGS */
/*=====*/

create table CONSTRUCTORS_STANDINGS (
  CONSTRUCTORSTANDINGSID INTEGER      not null,
  RACEID             INTEGER          not null,
  CONSTRUCTORID      INTEGER          not null,
  POINTS             INTEGER          not null,
  POSITION            INTEGER          not null,
  POSITIONTEXT        VARCHAR2(3)      not null,
  WINS               INTEGER          not null,
  constraint PK_CONSTRUCTORS_STANDINGS primary key (CONSTRUCTORSTANDINGSID)
);
```

```
alter table CONSTRUCTORS_STANDINGS
  add constraint FK_CON_STAND_REFERENCE_CON foreign key (CONSTRUCTORID)
  references CONSTRUCTORS (CONSTRUCTORID);

alter table CONSTRUCTORS_STANDINGS
  add constraint FK_CON_STAND_REFERENCE_RACES foreign key (RACEID)
  references RACES (RACEID);

alter table CONSTRUCTOR_RESULTS
  add constraint FK_CON_REFERENCE_CON_RESULTS foreign key (CONSTRUCTORID)
  references CONSTRUCTORS (CONSTRUCTORID);

alter table CONSTRUCTOR_RESULTS
  add constraint FK_CON_RESULTS_REFERENCE_RACES foreign key (RACEID)
  references RACES (RACEID);

alter table DRIVER_STANDINGS
  add constraint FK_DRIVERS_REFERENCE_DRIVERS foreign key (DRIVERID)
  references DRIVERS (DRIVERID);

alter table DRIVER_STANDINGS
  add constraint FK_DRIVERS_REFERENCE_RACES foreign key (RACEID)
  references RACES (RACEID);

alter table LAP_TIMES
  add constraint FK_LAP_TIME_REFERENCE_DRIVERS foreign key (DRIVERID)
  references DRIVERS (DRIVERID);

alter table LAP_TIMES
  add constraint FK_LAP_TIME_REFERENCE_RACES foreign key (RACEID)
  references RACES (RACEID);

alter table PIT_STOPS
  add constraint FK_PIT_STOP_REFERENCE_DRIVERS foreign key (DRIVERID)
  references DRIVERS (DRIVERID);
```



# Conexão do ambiente de desenvolvimento com o banco de dados



Novo / Selecionar Conexão do Banco de Dados

Nome da Conexão	Detalhes da Con...
FIA_ACCENTURE	ADMIN@fia_high

Name: FIA\_ACCENTURE ☐ Color

Tipo de Banco de Dados: Oracle

**Informações do Usuário** | Usuário Proxy

Tipo de Autenticação: Padrão

Nome do Usuário: ADMIN Atribuição: padrão

Senha:  ☒ Salvar Senha

Tipo de Conexão: Wallet do Cloud

**Detalhes** | Avançado | Proxy

Arquivo de Configuração: ents\DATAWIZARDS\_ACCENTURE\Wallet\_FIA.zip

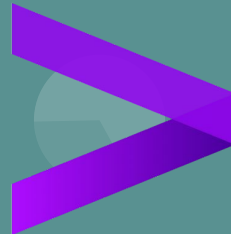
Serviço: fia\_high

Status:





# Povoamento do banco (DML)



Inserção manual através de concatenações no Excel

B2						hamilton					
D	E	F	G	H	I						
1	code	forename	surname	dob	nationality						
2	HAM	Lewis	Hamilton	1985-01-07	British	insert into DRIVERS values (1, 'hamilton', 44, 'HAM', 'Lewis', 'Hamilton', NVL('', TO_DATE('1985-01-07', 'YYYY/MM/DD')), 'British');					
3	HEI	Nick	Heidfeld	1977-05-10	German	insert into DRIVERS values (2, 'heidfeld', \N, 'HEI', 'Nick', 'Heidfeld', NVL('', TO_DATE('1977-05-10', 'YYYY/MM/DD')), 'German');					
4	ROS	Nico	Rosberg	1985-06-27	German	insert into DRIVERS values (3, 'rosberg', 6, 'ROS', 'Nico', 'Rosberg', NVL('', TO_DATE('1985-06-27', 'YYYY/MM/DD')), 'German');					
5	ALO	Fernando	Alonso	1981-07-29	Spanish	insert into DRIVERS values (4, 'alonso', 14, 'ALO', 'Fernando', 'Alonso', NVL('', TO_DATE('1981-07-29', 'YYYY/MM/DD')), 'Spanish');					
6	KOV	Heikki	Kovalainen	1981-10-19	Finnish	insert into DRIVERS values (5, 'kovalainen', \N, 'KOV', 'Heikki', 'Kovalainen', NVL('', TO_DATE('1981-10-19', 'YYYY/MM/DD')), 'Finnish');					
7	NAK	Kazuki	Nakajima	1985-01-11	Japanese	insert into DRIVERS values (6, 'nakajima', \N, 'NAK', 'Kazuki', 'Nakajima', NVL('', TO_DATE('1985-01-11', 'YYYY/MM/DD')), 'Japanese');					
8	BOU	Sébastien	Bourdais	1979-02-28	French	insert into DRIVERS values (7, 'bourdais', \N, 'BOU', 'Sébastien', 'Bourdais', NVL('', TO_DATE('1979-02-28', 'YYYY/MM/DD')), 'French');					
9	RAI	Kimi	Räikkönen	1979-10-17	Finnish	insert into DRIVERS values (8, 'raikkonen', 7, 'RAI', 'Kimi', 'Räikkönen', NVL('', TO_DATE('1979-10-17', 'YYYY/MM/DD')), 'Finnish');					
10	KUB	Robert	Kubica	1984-12-07	Polish	insert into DRIVERS values (9, 'kubica', 88, 'KUB', 'Robert', 'Kubica', NVL('', TO_DATE('1984-12-07', 'YYYY/MM/DD')), 'Polish');					
11	GLO	Timo	Glock	1982-03-18	German	insert into DRIVERS values (10, 'glock', \N, 'GLO', 'Timo', 'Glock', NVL('', TO_DATE('1982-03-18', 'YYYY/MM/DD')), 'German');					
12	SAT	Takuma	Sato	1977-01-28	Japanese	insert into DRIVERS values (11, 'sato', \N, 'SAT', 'Takuma', 'Sato', NVL('', TO_DATE('1977-01-28', 'YYYY/MM/DD')), 'Japanese');					
13	PIQ	Nelson	Piquet Jr.	1985-07-25	Brazilian	insert into DRIVERS values (12, 'piquet_jr', \N, 'PIQ', 'Nelson', 'Piquet Jr.', NVL('', TO_DATE('1985-07-25', 'YYYY/MM/DD')), 'Brazilian');					
14	MAS	Felipe	Massa	1981-04-25	Brazilian	insert into DRIVERS values (13, 'massa', 19, 'MAS', 'Felipe', 'Massa', NVL('', TO_DATE('1981-04-25', 'YYYY/MM/DD')), 'Brazilian');					
15	COU	David	Coulthard	1971-03-27	British	insert into DRIVERS values (14, 'coulthard', \N, 'COU', 'David', 'Coulthard', NVL('', TO_DATE('1971-03-27', 'YYYY/MM/DD')), 'British');					
16	TRU	Jarno	Trulli	1974-07-13	Italian	insert into DRIVERS values (15, 'trulli', \N, 'TRU', 'Jarno', 'Trulli', NVL('', TO_DATE('1974-07-13', 'YYYY/MM/DD')), 'Italian');					
17	SUT	Adrian	Sutil	1983-01-11	German	insert into DRIVERS values (16, 'sutil', 99, 'SUT', 'Adrian', 'Sutil', NVL('', TO_DATE('1983-01-11', 'YYYY/MM/DD')), 'German');					
18	WEB	Mark	Webber	1976-08-27	Australian	insert into DRIVERS values (17, 'webber', \N, 'WEB', 'Mark', 'Webber', NVL('', TO_DATE('1976-08-27', 'YYYY/MM/DD')), 'Australian');					
19	BUT	Jenson	Button	1980-01-19	British	insert into DRIVERS values (18, 'button', 22, 'BUT', 'Jenson', 'Button', NVL('', TO_DATE('1980-01-19', 'YYYY/MM/DD')), 'British');					
20	DAV	Anthony	Davidson	1979-04-18	British	insert into DRIVERS values (19, 'davidson', \N, 'DAV', 'Anthony', 'Davidson', NVL('', TO_DATE('1979-04-18', 'YYYY/MM/DD')), 'British');					
21	VET	Sebastian	Vettel	1987-07-03	German	insert into DRIVERS values (20, 'vettel', 5, 'VET', 'Sebastian', 'Vettel', NVL('', TO_DATE('1987-07-03', 'YYYY/MM/DD')), 'German');					

# Através do Assistente de Importação de Dados do Oracle SQL Developer

Assistente de Importação de Dados - Etapa 1 de 4

## Visualização de Dados

**Visualização de Dados**  
Método de Importação  
Definição de Coluna  
Concluir

Restaurar Estado

Origem: Arquivo Local

Arquivo: C:\Users\bjagr\Documents\DATAWIZARDS\_ACCENTURE\TABELAS CSV\results.csv Procurar...

Formato do Arquivo

☒ Cabeç. Após Ignorar Ignorar Linhas: 0

Formato: csv ☒ Visualizar Limite de Linhas: 100

Codificação: Cp1252

Delimitador: , Finalizador de Linha: padrão: CR LF, CR ou LF

Delimitador Esquerdo: " Delimitador Direito: "

Conteúdo do Arquivo

resultId	raceId	driverId	constructorId	number	grid	position	positionText	positionOrder	points	laps
1	18	1	1	22	1	1	1	1	10	58
2	18	2	2	3	5	2	2	2	8	58
3	18	3	3	7	7	3	3	3	6	58
4	18	4	4	5	11	4	4	4	5	58
5	18	5	1	23	3	5	5	5	4	58
6	18	6	3	8	13	6	6	6	3	57
7	18	7	5	14	17	7	7	7	2	55
8	18	8	6	1	15	8	8	8	1	53
9	18	9	2	4	2	null	R	9	0	47
10	18	10	7	12	18	null	R	10	0	43

Ajuda < Voltar Próximo > Finalizar Cancelar



## 2) Modelagem Multidimensional

Ferramentas utilizadas:

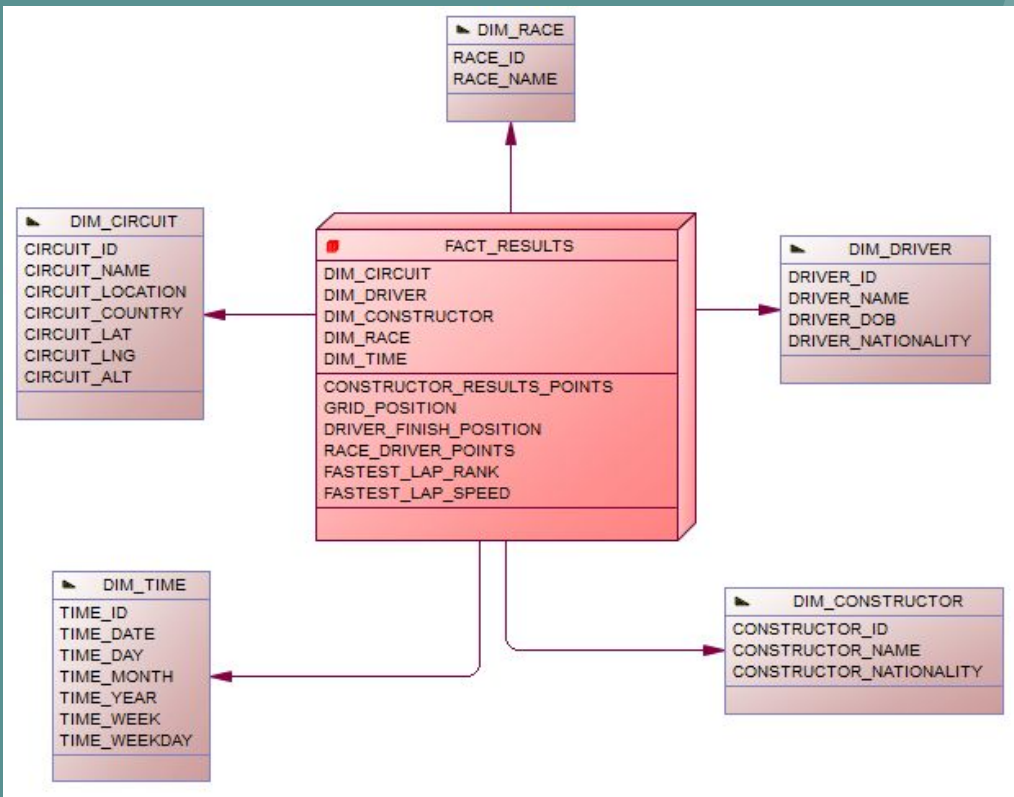
- SAP Powerdesigner: Utilizado na criação do DER
- Oracle Cloud 19c Autonomous Database (Free trial 30 dias)
- Oracle SQL Developer (ambiente de desenvolvimento)

O que foi feito:

- Definição do esquema (star schema), da granularidade e das tabelas fato e dimensão.
- Criação e povoamento das tabelas no banco através de SELECTS e JOINS



## 2) Diagrama Entidade Relacionamento Multidimensional



Esquema: Star Schema

Granularidade: Piloto por Corrida

Fato:  
Resultados das corridas

Dimensões:  
Piloto  
Tempo  
Corrida  
Circuito  
Construtora





# DDL DW

```
-- factResults
CREATE TABLE FACT_RESULTS AS (
SELECT
RESULTS.RESULTID AS "RESULT_ID",
DRIVERS.DRIVERID AS "DRIVER_ID",
CONSTRUCTORS.CONSTRUCTORID AS "CONSTRUCTOR_ID",
RACES.RACEID AS "RACE_ID",
CIRCUITS.CIRCUITID AS "CIRCUIT_ID",
TO_CHAR(RACES."DATE", 'YYYYMMDD') AS "DATE_ID",
NVL(CONSTRUCTOR_RESULTS.POINTS, 0) AS "CONSTRUCTOR_RESULTS_POINTS",
RESULTS.GRID AS "GRID_POSITION",
RESULTS.POSITIONORDER AS "DRIVER_FINISH_POSITION",
RESULTS.POINTS AS "RACE_DRIVER_POINTS",
NVL(RESULTS."RANK", 0) AS "FASTEST_LAP_RANK",
RESULTS.FASTESTLAPSPEED AS "FASTEST_LAP_SPEED"

FROM RESULTS
LEFT JOIN RACES
ON RESULTS.RACEID = RACES.RACEID
LEFT JOIN DRIVERS
ON RESULTS.DRIVERID = DRIVERS.DRIVERID
LEFT JOIN CONSTRUCTORS
ON RESULTS.CONSTRUCTORID = CONSTRUCTORS.CONSTRUCTORID
LEFT JOIN CIRCUITS
ON RACES.CIRCUITID = CIRCUITS.CIRCUITID
LEFT JOIN CONSTRUCTOR_RESULTS
ON CONSTRUCTORS.CONSTRUCTORID = CONSTRUCTOR_RESULTS.CONSTRUCTORID
AND RACES.RACEID = CONSTRUCTOR_RESULTS.RACEID
);
```

```
CREATE TABLE DIM_RACES AS (
SELECT
RACEID AS "RACE_ID",
CONCAT(NAME, CONCAT(' ', YEAR)) AS "RACE_NAME"
FROM
RACES
);

CREATE TABLE DIM_CIRCUITS AS (
SELECT
CIRCUITID AS "CIRCUIT_ID",
"NAME" AS "CIRCUIT_NAME",
LOCATION AS "CIRCUIT_LOCATION",
COUNTRY AS "CIRCUIT_COUNTRY",
LAT AS "CIRCUIT_LATITUDE",
LNG AS "CIRCUIT_LONGITUDE",
ALT AS "CIRCUIT_ALTITUDE"
FROM
CIRCUITS
);

CREATE TABLE DIM_DRIVERS AS (
SELECT
DRIVERID AS "DRIVER_ID",
CONCAT(FORENAME, CONCAT(' ', SURNAME)) AS "DRIVER_NAME",
NATIONALITY AS "DRIVER_NATIONALITY"
FROM
DRIVERS
);

CREATE TABLE DIM_CONSTRUCTORS AS (
SELECT
CONSTRUCTORID AS "CONSTRUCTOR_ID",
"NAME" AS "CONSTRUCTOR_NAME",
NATIONALITY AS "CONSTRUCTOR_NATIONALITY"
FROM
CONSTRUCTORS
);
```

# Exportação do DW para análise

**Assistente de Exportação - Etapa 1 de 3**

**Origem/Destino**

Conexão: FIA\_ACCENTURE

☐ Exportar DDL

☒ Adicionar palavra-chave BYTE ☒ Adicionar Imposição às Views ☐ Colocar Eliminações em Cascata

☐ Elimina ☒ Congede ☒ Particionamento

☒ Mostrar Esquema ☒ Armazenamento ☒ Tablespace

Vers... COMPATIBLE

☒ Exportar Dados

Formato: csv ☒ Cabeç.

Delimitador: ; Finalizador de Linha: ambiente padrão

Delimitador Esquerdo: \* Delimitador Direito: \*

Salvar como: Arquivo Único ☐ Compactado Codificação: UTF-8

Arquivo: nts\DATAWIZARDS\_ACCENTURE\sqldeveloper\sqldeveloper\bin\sqlcsv Procurar...

☐ Ir para resumo.

**Relatórios**

Todos os Relatórios

- Relatórios Definidos pelo Usuário
- Relatórios de View Analítica
- Relatórios do Data Modeler
- Relatórios do Dicionário de Dados
- Relatórios do TimesTen
- Relatórios OLAP

1	2	3	4	5	6	7	8	9	10	11
19700109	09/01/70	Sexta-Feira	1	1	1	1	1	1	1	1
19700110	10/01/70	Sábado	1	1	1	1	1	1	1	1
19700111	11/01/70	Domingo	2	1	1	1	1	1	1	1



# Análise dos dados em Python

Notebook desenvolvido no Google Colab

