

Git: <https://github.com/leonardoamorim/arquiteturadebigdata>

## Atividade 01 - Obtenção de dados e inserção de dados no HDFS

The screenshot shows the Ambari Sandbox interface with the HDFS file browser. The path is /user/maria\_dev/flightdelays. The table lists three files:

Name	Size	Last Modified	Owner	Group	Permission
flight_delays1.csv	934.1 kB	2025-06-23 16:42	maria_dev	hdfs	-rW-r--r--
flight_delays2.csv	939.3 kB	2025-06-23 16:42	maria_dev	hdfs	-rW-r--r--
flight_delays3.csv	950.1 kB	2025-06-23 16:42	maria_dev	hdfs	-rW-r--r--

The screenshot shows the Ambari Sandbox interface with the Hive Query Editor. The Database Explorer shows the default database selected. The Query Editor has a new worksheet named '1' open. The status bar at the bottom right shows 'Selected database : default'.

The screenshot shows an SSH session in MobaXterm Personal Edition v25.2. The title bar says 'localhost (maria\_dev)'. The terminal window displays the following text:

```
• MobaXterm Personal Edition v25.2 •
(SSH client, X server and network tools)

▶ SSH session to maria_dev@DESKTOP-FPAF2KH
  • Direct SSH : ✓
  • SSH compression : ✓
  • SSH-browser : ✓
  • X11-forwarding : ✘ (disabled or not supported by server)

▶ For more info, ctrl+click on help or visit our website.

Last login: Mon Jun 23 19:50:27 2025 from 10.0.2.2
[maria_dev@sandbox-hdp ~]$
```

## Atividade 02 - Criação de tabela externa no Hive

```
DROP TABLE IF EXISTS flightdelays;  
  
CREATE EXTERNAL TABLE flightdelays (  
    Year INT,  
    Month INT,  
    DayofMonth INT,  
    DayOfWeek INT,  
    DepTime INT,  
    CRSDepTime INT,  
    ArrTime INT,  
    CRSArrTime INT,  
    UniqueCarrier STRING,  
    FlightNum INT,  
    TailNum STRING,  
    ActualElapsedTime INT,  
    CRSElapsedTime INT,  
    AirTime INT,  
    ArrDelay INT,  
    DepDelay INT,  
    Origin STRING,  
    Dest STRING,  
    Distance INT,  
    TaxiIn INT,  
    TaxiOut INT,  
    Cancelled INT,  
    CancellationCode STRING,  
    Diverted INT,
```

```

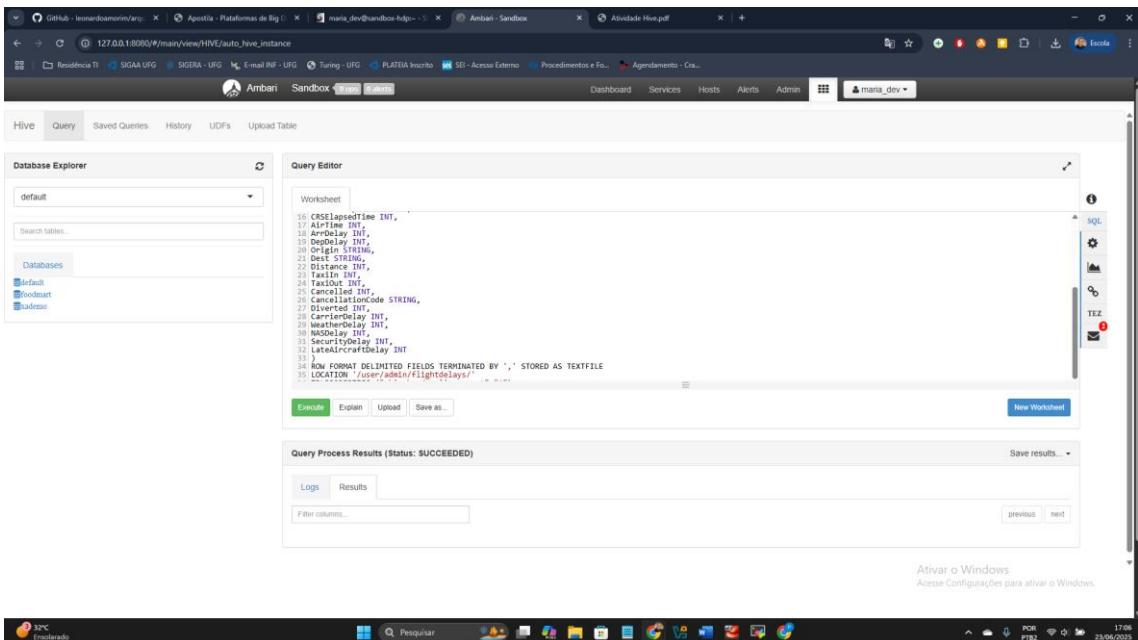
CarrierDelay INT,
WeatherDelay INT,
NASDelay INT,
SecurityDelay INT,
LateAircraftDelay INT

)

ROW FORMAT DELIMITED FIELDS TERMINATED BY ',' STORED AS TEXTFILE
LOCATION '/user/maria_dev/flightdelays/'

TBLPROPERTIES ("skip.header.line.count"="1");

```



The screenshot shows the Apache Ambari interface for managing Hadoop clusters. The main window is the 'Query Editor' for the 'Hive' service. In the 'Worksheet' tab, a complex SQL query is displayed, starting with creating a temporary table and defining various integer and string columns. It then specifies the row format as delimited fields terminated by a comma and stored as a textfile at a specific location. Finally, it sets a table property to skip the first header line. Below the worksheet, the 'Logs' tab of the 'Query Process Results' section shows a 'SUCCEEDED' status. The Ambari navigation bar at the top includes links for GitHub, Agenzia - Plataformas de Big Data, E-mail INF - UFG, SIGAA - UFG, SIGER - UFG, Procedimentos e Fórum, and Agendamento - Criação.

```

CREATE TEMPORARY TABLE #temp_table (
    16 CRSElapsedTime INT,
    17 AirTime INT,
    18 ArrDelay INT,
    19 DepDelay INT,
    20 Origin STRING,
    21 Dest STRING,
    22 Distance INT,
    23 TailNum INT,
    24 Carrier INT,
    25 Cancelled INT,
    26 CarrierFlightMode STRING,
    27 Diverted INT,
    28 CarrierDelay INT,
    29 WeatherDelay INT,
    30 NASDelay INT,
    31 SecurityDelay INT,
    32 LateAircraftDelay INT
)
ROW FORMAT DELIMITED FIELDS TERMINATED BY ',' STORED AS TEXTFILE
LOCATION '/user/admin/flightdelays/'

TBLPROPERTIES ("skip.header.line.count"="1");

```

## Atividade 03 - Analisando dados com o Hive

The screenshot shows the Ambari Sandbox interface. In the top navigation bar, there are several tabs: Residência TI, SIGAA UFG, SIGERA - UFG, E-mail INF - UFG, Turnig - UFG, PLATEIA Insrito, SEI - Acesso Externo, Procedimentos e Fo..., and Agendamento - Cr... The main area has tabs for Hive, Query, Saved Queries, History, UDF's, and Upload Table. The Database Explorer on the left shows a dropdown set to 'default' and a list of databases: default, foodmart, and addemo. The Query Editor contains a worksheet with the following SQL query:

```
1: SELECT AVG(arrdelay) FROM flightdelays WHERE dest = 'DEN'
```

Below the editor are buttons for Execute, Explain, Upload, and Save as. The Query Process Results section shows the status as SUCCEEDED and displays the results in a table:

_c0
7.264646464646464

At the bottom right of the results table, there is a message: "Ativar o Windows" and "Acesse Configurações para ativar o Windows."

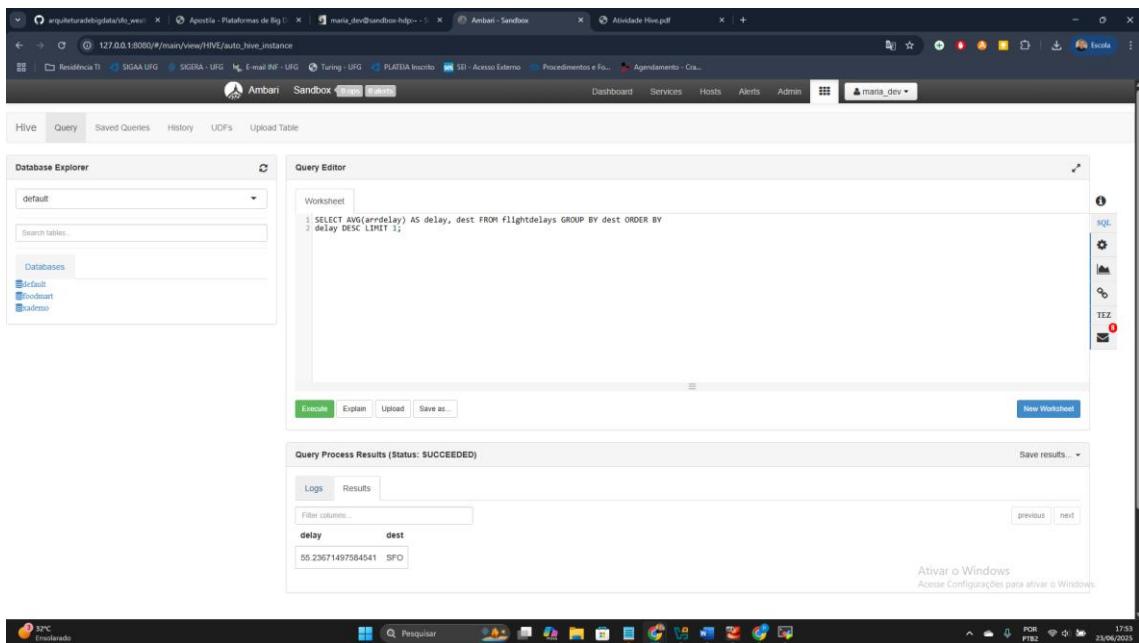
This screenshot shows the same Ambari Sandbox interface as the previous one. The Database Explorer shows the 'default' database selected. The Query Editor contains a worksheet with the following SQL query:

```
1: SELECT AVG(arrdelay) FROM flightdelays WHERE origin = 'LAX' AND dest = 'SFO';
```

The Query Process Results section shows the status as SUCCEEDED and displays the results in a table:

_c0
63.660377358490564

At the bottom right of the results table, there is a message: "Ativar o Windows" and "Acesse Configurações para ativar o Windows."



## Atividade 04 - Definir e preencher uma tabela ORCFile

-- Cria tabela temporária em formato texto

DROP TABLE IF EXISTS sfo\_weather\_txt;

CREATE TABLE sfo\_weather\_txt (

station\_name STRING,

year INT,

month INT,

dayofmonth INT,

precipitation INT,

temperature\_max INT,

temperature\_min INT

)

ROW FORMAT DELIMITED

FIELDS TERMINATED BY ',';

STORED AS TEXTFILE;

-- Carrega dados do arquivo CSV local

```
LOAD DATA LOCAL INPATH  
'/home/maria_dev/arquiteturadebigdata/sfo_weather.csv'  
OVERWRITE INTO TABLE sfo_weather_txt;
```

```
-- Cria tabela definitiva em formato ORC
```

```
DROP TABLE IF EXISTS sfo_weather;
```

```
CREATE TABLE sfo_weather (
```

```
    station_name STRING,
```

```
    year INT,
```

```
    month INT,
```

```
    dayofmonth INT,
```

```
    precipitation INT,
```

```
    temperature_max INT,
```

```
    temperature_min INT
```

```
)
```

```
STORED AS ORC;
```

```
-- Insere dados da tabela texto para a tabela ORC
```

```
INSERT INTO TABLE sfo_weather
```

```
SELECT * FROM sfo_weather_txt;
```

```
-- Consulta os dados da tabela ORC
```

```
SELECT * FROM sfo_weather;
```

Screenshot of a web-based Hadoop/Hive interface showing a query editor and results.

**Database Explorer**

- default
- Search tables...
- Databases:
  - default
  - goodmart
  - academe

**Query Editor**

```
8 month INT,
9 precipitation INT,
10 temperature INT,
11 temperature_min INT,
12
13 ROW FORMAT DELIMITED
14 FIELDS TERMINATED BY ','
15 STORED AS TEXTFILE;
16
17 -- Cria tabela ORC se não existir
18 LOAD DATA INPATH '/user/maria_dev/flightdelays/sfo_weather.csv'
19 OVERWRITE INTO TABLE sfo_weather;
20
21 -- Remover tabela ORC se já existir
22 DROP TABLE IF EXISTS sfo_weather;
23
24 -- Criar tabela ORC otimizada
25 CREATE TABLE sfo_weather (
26 station_name STRING,
```

**Query Process Results (Status: SUCCEEDED)**

sfo_weather.station_name	sfo_weather.year	sfo_weather.month	sfo_weather.dayofmonth	sfo_weather.precipitation	sfo_weather.temperature_max	sfo_weather.temperature_min
SAN FRANCISCO INTERNATIONAL AIRPORT CA US	2008	1	1	0	122	39
SAN FRANCISCO INTERNATIONAL AIRPORT CA US	2008	1	2	0	117	39
SAN FRANCISCO INTERNATIONAL AIRPORT CA US	2008	1	3	43	150	94
SAN FRANCISCO INTERNATIONAL AIRPORT CA US	2008	1	4	533	150	100

Screenshot of a web-based Hadoop/Hive interface showing a query editor and results.

**Query Process Results (Status: SUCCEEDED)**

sfo_weather.station_name	sfo_weather.year	sfo_weather.month	sfo_weather.dayofmonth	sfo_weather.precipitation	sfo_weather.temperature_max	sfo_weather.temperature_min
SAN FRANCISCO INTERNATIONAL AIRPORT CA US	2008	1	1	0	122	39
SAN FRANCISCO INTERNATIONAL AIRPORT CA US	2008	1	2	0	117	39
SAN FRANCISCO INTERNATIONAL AIRPORT CA US	2008	1	3	43	150	94
SAN FRANCISCO INTERNATIONAL AIRPORT CA US	2008	1	4	533	150	100
SAN FRANCISCO INTERNATIONAL AIRPORT CA US	2008	1	5	196	122	78
SAN FRANCISCO INTERNATIONAL AIRPORT CA US	2008	1	6	15	106	50
SAN FRANCISCO INTERNATIONAL AIRPORT CA US	2008	1	7	0	111	67
SAN FRANCISCO INTERNATIONAL AIRPORT CA US	2008	1	8	20	128	61
SAN FRANCISCO INTERNATIONAL AIRPORT CA US	2008	1	9	3	106	67
SAN FRANCISCO INTERNATIONAL AIRPORT CA US	2008	1	10	25	100	89
SAN FRANCISCO INTERNATIONAL AIRPORT CA US	2008	1	11	0	117	89
SAN FRANCISCO INTERNATIONAL AIRPORT CA US	2008	1	12	0	133	67
SAN FRANCISCO INTERNATIONAL AIRPORT CA US	2008	1	13	0	144	67

SAN FRANCISCO INTERNATIONAL AIRPORT CA US	2008	2	6	3	122	83
SAN FRANCISCO INTERNATIONAL AIRPORT CA US	2008	2	7	0	139	50
SAN FRANCISCO INTERNATIONAL AIRPORT CA US	2008	2	8	0	161	78
SAN FRANCISCO INTERNATIONAL AIRPORT CA US	2008	2	9	0	189	56
SAN FRANCISCO INTERNATIONAL AIRPORT CA US	2008	2	10	0	189	72
SAN FRANCISCO INTERNATIONAL AIRPORT CA US	2008	2	11	0	189	72
SAN FRANCISCO INTERNATIONAL AIRPORT CA US	2008	2	12	0	189	72
SAN FRANCISCO INTERNATIONAL AIRPORT CA US	2008	2	13	0	156	78
SAN FRANCISCO INTERNATIONAL AIRPORT CA US	2008	2	14	0	156	83
SAN FRANCISCO INTERNATIONAL AIRPORT CA US	2008	2	15	0	156	50
SAN FRANCISCO INTERNATIONAL AIRPORT CA US	2008	2	16	0	150	56
SAN FRANCISCO INTERNATIONAL AIRPORT CA US	2008	2	17	0	122	56
SAN FRANCISCO INTERNATIONAL AIRPORT CA US	2008	2	18	0	122	83
SAN FRANCISCO INTERNATIONAL AIRPORT CA US	2008	2	19	163	122	89

Ativar o Windows  
Acesse Configurações para ativar o Windows.



## Atividade 05 - Junção de dados do Hive

```
SET hive.execution.engine= tez;
```

```
DROP TABLE IF EXISTS flights_weather;
```

```
CREATE TABLE flights_weather STORED AS TEXTFILE AS SELECT fd.*,
sw.temperature_max, sw.temperature_min FROM flightdelays fd JOIN sfo_weather
SW
ON fd.year = sw.year AND fd.month = sw.month AND fd.dayofmonth =
sw.dayofmonth
WHERE fd.origin = 'SFO' OR fd.dest = 'SFO';
```

```
SELECT * FROM flights_weather;
```

Screenshot of a web-based Hadoop/Hive interface showing a query editor and results.

**Database Explorer:**

- Selected Database: default
- Search tables: flights\_weather
- Databases: default, rodamart, academe

**Query Editor:**

```
Worksheet
1 SET Hive.execution.engine=tez;
2 DROP TABLE IF EXISTS flights_weather;
3 CREATE TABLE flights_weather AS TEXTFILE AS SELECT fd.*,
4   sw.temperature
5   FROM flights fd JOIN temperature sw ON fd.year = sw.year AND fd.month = sw.month AND fd.dayofmonth = sw.dayofmonth
6   WHERE fd.origin = 'SFO' OR fd.dest = 'SFO';
7 SELECT * FROM flights_weather;
```

**Buttons:** Execute, Explain, Upload, Save as..., New Worksheet

**Query Process Results (Status: SUCCEEDED):**

flights_weather.year	flights_weather.month	flights_weather.dayofmonth	flights_weather.dayofweek	flights_weather.deptime	flights_weather.crsdepetime	flights_weather.arrrtime	flights_weather.wt
2008	1	3	4	1426	1355	1605	1530
2008	1	3	4	1009	910	1148	1045
2008	1	3	4	2021	1700	2303	1835
2008	1	3	4	2025	1905	2208	2040
2008	1	3	4	603	605	729	740
2008	1	3	4	2301	2105	59	2240
2008	1	3	4	1518	1215	1645	1350
2008	1	3	4	708	710	829	835
2008	1	3	4	null	905	null	1025
2008	1	3	4	2321	1955	38	2115
2008	1	3	4	null	1620	null	1740
2008	1	3	4	2008	1805	2139	1930
2008	1	3	4	1625	1430	1748	1550
2008	1	3	4	1305	1050	1421	1210
2008	1	3	4	1558	1245	1709	1405
2008	1	3	4	2131	1915	29	2205
2008	1	3	4	1736	1305	2031	1555
2008	1	3	4	1319	950	1615	1240
2008	1	3	4	1716	1440	1854	1620
2008	1	3	4	2202	2110	2344	2250
2008	1	3	4	1839	1720	1900	1900
2008	1	3	4	1319	1230	1445	1410

Ativar o Windows Access 2008 configurações para ativar o Windows.

Screenshot of a second instance of the web-based Hadoop/Hive interface showing a query editor and results.

**Database Explorer:**

- Selected Database: default
- Search tables: flights\_weather
- Databases: default, rodamart, academe

**Query Editor:**

```
Worksheet
1 SET Hive.execution.engine=tez;
2 DROP TABLE IF EXISTS flights_weather;
3 CREATE TABLE flights_weather AS TEXTFILE AS SELECT fd.*,
4   sw.temperature
5   FROM flights fd JOIN temperature sw ON fd.year = sw.year AND fd.month = sw.month AND fd.dayofmonth = sw.dayofmonth
6   WHERE fd.origin = 'SFO' OR fd.dest = 'SFO';
7 SELECT * FROM flights_weather;
```

**Buttons:** Execute, Explain, Upload, Save as..., New Worksheet

**Query Process Results (Status: SUCCEEDED):**

flights_weather.year	flights_weather.month	flights_weather.dayofmonth	flights_weather.dayofweek	flights_weather.deptime	flights_weather.crsdepetime	flights_weather.arrrtime	flights_weather.wt
2008	1	3	4	1426	1355	1605	1530
2008	1	3	4	1009	910	1148	1045
2008	1	3	4	2021	1700	2303	1835
2008	1	3	4	2025	1905	2208	2040
2008	1	3	4	603	605	729	740
2008	1	3	4	2301	2105	59	2240
2008	1	3	4	1518	1215	1645	1350
2008	1	3	4	708	710	829	835
2008	1	3	4	null	905	null	1025
2008	1	3	4	2321	1955	38	2115
2008	1	3	4	null	1620	null	1740
2008	1	3	4	2008	1805	2139	1930
2008	1	3	4	1625	1430	1748	1550
2008	1	3	4	1305	1050	1421	1210
2008	1	3	4	1558	1245	1709	1405
2008	1	3	4	2131	1915	29	2205
2008	1	3	4	1736	1305	2031	1555
2008	1	3	4	1319	950	1615	1240
2008	1	3	4	1716	1440	1854	1620
2008	1	3	4	2202	2110	2344	2250
2008	1	3	4	1839	1720	1900	1900
2008	1	3	4	1319	1230	1445	1410

Ativar o Windows Access 2008 configurações para ativar o Windows.

year	month	dayofmonth	precipitation	temperature_max	temperature_min	null
2008	1	3	4	1516	1440	1646
2008	1	3	4	844	645	1020
2008	1	3	4	2059	1620	2216
2008	1	3	4	2225	2105	2350
2008	1	3	4	1800	1805	1929
2008	1	3	4	852	650	1009
2008	1	3	4	1053	1050	1213
2008	1	3	4	1801	1615	1919
2008	1	3	4	1447	1235	1558
2008	1	3	4	2165	1955	2308
2008	1	3	4	null	710	840
2008	1	3	4	null	1430	1555
2008	1	3	4	1657	1310	2247
2008	1	3	4	1911	1650	103
2008	1	3	4	703	705	1316
2008	1	3	4	1221	1040	1353
2008	1	3	4	803	805	929
2008	1	3	4	619	620	742
2008	1	3	4	1324	1220	1447
2008	1	3	4	2325	1900	null
2008	1	3	4	1711	1415	1831
2008	1	3	4			1545

## Atividade 06 - Tabelas particionadas do Hive

-- Remover a tabela particionada se já existir

```
DROP TABLE IF EXISTS weather_partitioned;
```

-- Criar tabela particionada por ano e mês

```
CREATE TABLE weather_partitioned (
```

```
station_name STRING,
```

```
dayofmonth INT,
```

```
precipitation INT,
```

```
temperature_max INT,
```

```
temperature_min INT
```

```
)
```

```
PARTITIONED BY (year INT, month INT)
```

```
STORED AS ORC;
```

-- Inserir dados na partição do ano de 2008 e mês 1 (Janeiro)

```
INSERT INTO TABLE weather_partitioned
```

```
PARTITION (year=2008, month=1)
```

```
SELECT
```

```
station_name,  
dayofmonth,  
precipitation,  
temperature_max,  
temperature_min  
FROM sfo_weather  
WHERE year = 2008 AND month = 1;
```

-- Consultar todos os dados da tabela particionada

```
SELECT * FROM weather_partitioned;
```

weather_partitioned.station_name	weather_partitioned.dayofmonth	weather_partitioned.precipitation	weather_partitioned.temperature_max	weather_partitioned.temperature_min	weather_partitioned.year
SAN FRANCISCO INTERNATIONAL AIRPORT CA US	1	0	122	39	2008
SAN FRANCISCO INTERNATIONAL AIRPORT CA US	2	0	117	39	2008
SAN FRANCISCO INTERNATIONAL AIRPORT CA US	3	43	150	94	2008
SAN FRANCISCO INTERNATIONAL AIRPORT CA US	4	533	150	100	2008
SAN FRANCISCO INTERNATIONAL AIRPORT CA US	5	196	122	78	2008
SAN FRANCISCO INTERNATIONAL AIRPORT CA US	6	15	106	50	2008
SAN FRANCISCO INTERNATIONAL AIRPORT CA US	7	0	111	67	2008
SAN FRANCISCO INTERNATIONAL AIRPORT CA US	8	20	128	61	2008
SAN FRANCISCO INTERNATIONAL AIRPORT CA US	9	3	106	67	2008
SAN FRANCISCO INTERNATIONAL AIRPORT CA US	10	25	100	89	2008
SAN FRANCISCO INTERNATIONAL AIRPORT CA US	11	0	117	89	2008
SAN FRANCISCO INTERNATIONAL AIRPORT CA US	12	0	133	83	2008
SAN FRANCISCO INTERNATIONAL AIRPORT CA US	13	0	144	67	2008

The screenshot shows a Windows desktop environment. At the top, there is a taskbar with several pinned icons: Residência TI, SIGAA - UFG, SIGERA - UFG, E-mail INF - UFG, Turing - UFG, PLATINA Inscrição, S3 - Acesso Externo, Procedimentos e Fo..., Agendamento - Cr., and Escola. Below the taskbar is a window titled "arquitetura de big data/lo\_web" which displays a table of data. The table has columns for ID, Name, Value, and Date. The data consists of 31 rows, all of which are identical: SAN FRANCISCO INTERNATIONAL AIRPORT CA US. The table is styled with alternating row colors. At the bottom right of the table, there is a message: "Ativar o Windows. Acesse Configurações para ativar Windows." The system tray at the bottom right shows the date as 23/06/2025 and the time as 18:06.

	ID	Name	Value	Date
1	10	SAN FRANCISCO INTERNATIONAL AIRPORT CA US	0	153
2	17	SAN FRANCISCO INTERNATIONAL AIRPORT CA US	0	139
3	18	SAN FRANCISCO INTERNATIONAL AIRPORT CA US	0	150
4	19	SAN FRANCISCO INTERNATIONAL AIRPORT CA US	0	122
5	20	SAN FRANCISCO INTERNATIONAL AIRPORT CA US	0	111
6	21	SAN FRANCISCO INTERNATIONAL AIRPORT CA US	152	83
7	22	SAN FRANCISCO INTERNATIONAL AIRPORT CA US	25	89
8	23	SAN FRANCISCO INTERNATIONAL AIRPORT CA US	15	83
9	24	SAN FRANCISCO INTERNATIONAL AIRPORT CA US	76	78
10	25	SAN FRANCISCO INTERNATIONAL AIRPORT CA US	645	117
11	26	SAN FRANCISCO INTERNATIONAL AIRPORT CA US	58	144
12	27	SAN FRANCISCO INTERNATIONAL AIRPORT CA US	81	133
13	28	SAN FRANCISCO INTERNATIONAL AIRPORT CA US	38	100
14	29	SAN FRANCISCO INTERNATIONAL AIRPORT CA US	20	100
15	30	SAN FRANCISCO INTERNATIONAL AIRPORT CA US	28	117
16	31	SAN FRANCISCO INTERNATIONAL AIRPORT CA US	13	117