Assignment: SQL Notebook for Peer Assignment

Estimated time needed: **60** minutes.

Introduction

Using this Python notebook you will:

1. Understand the Spacex DataSet
2. Load the dataset into the corresponding table in a Db2 database
3. Execute SQL queries to answer assignment questions

Overview of the DataSet

SpaceX has gained worldwide attention for a series of historic milestones.

It is the only private company ever to return a spacecraft from low-earth orbit, which it first accomplished in December 2010. SpaceX advertises Falcon 9 rocket launches on its website with a cost of 62 million dollars wheras other providers cost upward of 165 million dollars each, much of the savings is because Space X can reuse the first stage.

Therefore if we can determine if the first stage will land, we can determine the cost of a launch.

This information can be used if an alternate company wants to bid against SpaceX for a rocket launch.

This dataset includes a record for each payload carried during a SpaceX mission into outer space.

Download the datasets

This assignment requires you to load the spacex dataset.

In many cases the dataset to be analyzed is available as a .CSV (comma separated values) file, perhaps on the internet. Click on the link below to download and save the dataset (.CSV file):

Spacex DataSet

In [1]:

```
!pip install sqlalchemy==1.3.9
```

```
Requirement already satisfied: sqlalchemy==1.3.9 in /opt/conda/lib/python
3.11/site-packages (1.3.9)
```

Connect to the database

Let us first load the SQL extension and establish a connection with the database

In [2]:

```
!pip install ipython-sql
```

```
Requirement already satisfied: ipython-sql in /opt/conda/lib/python3.11/si
te-packages (0.5.0)
Requirement already satisfied: prettytable in /opt/conda/lib/python3.11/si
te-packages (from ipython-sql) (3.11.0)
Requirement already satisfied: ipython in /opt/conda/lib/python3.11/site-p
ackages (from ipython-sql) (8.22.2)
Collecting sqlalchemy>=2.0 (from ipython-sql)
  Using cached SQLAlchemy-2.0.35-cp311-cp311-manylinux_2_17_x86_64.manylin
ux2014_x86_64.whl.metadata (9.6 kB)
Requirement already satisfied: sqlparse in /opt/conda/lib/python3.11/site-
packages (from ipython-sql) (0.5.1)
Requirement already satisfied: six in /opt/conda/lib/python3.11/site-packa
ges (from ipython-sql) (1.16.0)
Requirement already satisfied: ipython-genutils in /opt/conda/lib/python3.
11/site-packages (from ipython-sql) (0.2.0)
Requirement already satisfied: typing-extensions>=4.6.0 in /opt/conda/lib/
python3.11/site-packages (from sqlalchemy>=2.0->ipython-sql) (4.11.0)
Requirement already satisfied: greenlet!=0.4.17 in /opt/conda/lib/python3.
11/site-packages (from sqlalchemy>=2.0->ipython-sql) (3.0.3)
Requirement already satisfied: decorator in /opt/conda/lib/python3.11/site
-packages (from ipython->ipython-sql) (5.1.1)
Requirement already satisfied: jedi>=0.16 in /opt/conda/lib/python3.11/sit
e-packages (from ipython->ipython-sql) (0.19.1)
Requirement already satisfied: matplotlib-inline in /opt/conda/lib/python
3.11/site-packages (from ipython->ipython-sql) (0.1.7)
Requirement already satisfied: prompt-toolkit<3.1.0,>=3.0.41 in /opt/cond
a/lib/python3.11/site-packages (from ipython->ipython-sql) (3.0.42)
Requirement already satisfied: pygments>=2.4.0 in /opt/conda/lib/python3.1
1/site-packages (from ipython->ipython-sql) (2.18.0)
Requirement already satisfied: stack-data in /opt/conda/lib/python3.11/sit
e-packages (from ipython->ipython-sql) (0.6.2)
Requirement already satisfied: traitlets>=5.13.0 in /opt/conda/lib/python
3.11/site-packages (from ipython->ipython-sql) (5.14.3)
Requirement already satisfied: pexpect>4.3 in /opt/conda/lib/python3.11/si
te-packages (from ipython->ipython-sql) (4.9.0)
Requirement already satisfied: wcwidth in /opt/conda/lib/python3.11/site-p
ackages (from prettytable->ipython-sql) (0.2.13)
Requirement already satisfied: parso<0.9.0,>=0.8.3 in /opt/conda/lib/pytho
n3.11/site-packages (from jedi>=0.16->ipython->ipython-sql) (0.8.4)
Requirement already satisfied: ptyprocess>=0.5 in /opt/conda/lib/python3.1
1/site-packages (from pexpect>4.3->ipython->ipython-sql) (0.7.0)
Requirement already satisfied: executing>=1.2.0 in /opt/conda/lib/python3.
11/site-packages (from stack-data->ipython->ipython-sql) (2.0.1)
Requirement already satisfied: asttokens>=2.1.0 in /opt/conda/lib/python3.
11/site-packages (from stack-data->ipython->ipython-sql) (2.4.1)
Requirement already satisfied: pure-eval in /opt/conda/lib/python3.11/site
-packages (from stack-data->ipython->ipython-sql) (0.2.2)
Using cached SQLAlchemy-2.0.35-cp311-cp311-manylinux_2_17_x86_64.manylinux
2014_x86_64.whl (3.2 MB)
Installing collected packages: sqlalchemy
  Attempting uninstall: sqlalchemy
    Found existing installation: SQLAlchemy 1.3.9
    Uninstalling SQLAlchemy-1.3.9:
      Successfully uninstalled SQLAlchemy-1.3.9
Successfully installed sqlalchemy-2.0.35
```

In [3]:

```
%load_ext sql
```

In [4]:

```
import csv, sqlite3
```

```
con = sqlite3.connect("my_data1.db")
cur = con.cursor()
```

In [5]:
```
!pip install -q pandas
```

In [6]:
```
%sql sqlite:///my_data1.db
```

In [7]:
```
import pandas as pd
df = pd.read_csv("https://cf-courses-data.s3.us.cloud-object-storage.appd
df.to_sql("SPACEXTBL", con, if_exists='replace', index=False,method="mult
```

Out[7]:
```
101
```

Note:This below code is added to remove blank rows from table

In [8]:
```
#DROP THE TABLE IF EXISTS

%sql DROP TABLE IF EXISTS SPACEXTABLE;
```

```
 * sqlite:///my_data1.db
Done.
```

Out[8]:
```
[]
```

In [9]:
```
%sql create table SPACEXTABLE as select * from SPACEXTBL where Date is no
```

```
 * sqlite:///my_data1.db
Done.
```

Out[9]:
```
[]
```

Tasks

Now write and execute SQL queries to solve the assignment tasks.

Note: If the column names are in mixed case enclose it in double quotes For Example "Landing_Outcome"

Task 1

Display the names of the unique launch sites in the space mission

In [13]:
```
%sql SELECT DISTINCT "launch_site" FROM SPACEXTBL;
```

```
 * sqlite:///my_data1.db
Done.
```

Out[13]:

Launch_Site
CCAFS LC-40
VAFB SLC-4E
KSC LC-39A
CCAFS SLC-40

Task 2

Display 5 records where launch sites begin with the string 'CCA'

In [14]:
```
%sql SELECT * FROM SPACEXTBL WHERE "launch_site" LIKE 'CCA%' LIMIT 5;
```

```
 * sqlite:///my_data1.db
Done.
```

Out[14]:

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_
2010-06-04	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0
2010-12-08	15:43:00	F9 v1.0 B0004	CCAFS LC-40	Dragon demo flight C1, two CubeSats, barrel of Brouere cheese	0
2012-05-22	7:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525
2012-10-08	0:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500
2013-03-01	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677

Task 3

Display the total payload mass carried by boosters launched by NASA (CRS)

In [17]:
```
%sql SELECT SUM(PAYLOAD_MASS__KG_) AS total_payload_mass FROM SPACEXTBL W
```

```
 * sqlite:///my_data1.db
Done.
```

Out[17]:

total_payload_mass
45596

Task 4

Display average payload mass carried by booster version F9 v1.1

In [20]:
```
%sql SELECT AVG(PAYLOAD_MASS__KG_) AS avg_payload_mass FROM SPACEXTBL WHE
```

```
 * sqlite:///my_data1.db
Done.
```

Out[20]:

avg_payload_mass
2928.4

Task 5

List the date when the first succesful landing outcome in ground pad was acheived.

Hint:Use min function

```
In [21]: %sql SELECT MIN(Date) AS First_Successful_Landing FROM SPACEXTBL WHERE La
```

```
 * sqlite:///my_data1.db
Done.
```

Out[21]:

First_Successful_Landing
2015-12-22

Task 6

List the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000

```
In [23]: %sql SELECT Booster_Version FROM SPACEXTBL WHERE Landing_Outcome = 'Succe
```

```
 * sqlite:///my_data1.db
Done.
```

Out[23]:

Booster_Version
F9 FT B1022
F9 FT B1026
F9 FT B1021.2
F9 FT B1031.2

Task 7

List the total number of successful and failure mission outcomes

```
In [24]: %sql SELECT Landing_Outcome, COUNT(*) AS Outcome_Count FROM SPACEXTBL GRO
```

```
 * sqlite:///my_data1.db
Done.
```

Out[24]:

Landing_Outcome	Outcome_Count
Controlled (ocean)	5
Failure	3
Failure (drone ship)	5
Failure (parachute)	2
No attempt	21
No attempt	1
Precluded (drone ship)	1
Success	38
Success (drone ship)	14
Success (ground pad)	9
Uncontrolled (ocean)	2

Task 8

List the names of the booster_versions which have carried the maximum payload mass. Use a subquery

In [25]:

```
%sql SELECT Booster_Version FROM SPACEXTBL WHERE PAYLOAD_MASS__KG_ = (SEL
```

```
 * sqlite:///my_data1.db
Done.
```

Out[25]:

Booster_Version
F9 B5 B1048.4
F9 B5 B1049.4
F9 B5 B1051.3
F9 B5 B1056.4
F9 B5 B1048.5
F9 B5 B1051.4
F9 B5 B1049.5
F9 B5 B1060.2
F9 B5 B1058.3
F9 B5 B1051.6
F9 B5 B1060.3
F9 B5 B1049.7

Task 9

List the records which will display the month names, failure landing_outcomes in drone ship ,booster versions, launch_site for the months in year 2015.

Note: SQLLite does not support monthnames. So you need to use substr(Date, 6,2) as month to get the months and substr(Date,0,5)='2015' for year.

In [26]:

```
%sql SELECT SUBSTR(Date, 6, 2) AS Month, Booster_Version, Launch_Site, La
```

```
 * sqlite:///my_data1.db
Done.
```

Out[26]:

Month	Booster_Version	Launch_Site	Landing_Outcome
01	F9 v1.1 B1012	CCAFS LC-40	Failure (drone ship)
04	F9 v1.1 B1015	CCAFS LC-40	Failure (drone ship)

Task 10

Rank the count of landing outcomes (such as Failure (drone ship) or Success (ground pad)) between the date 2010-06-04 and 2017-03-20, in descending order.

In [28]:

```
%sql SELECT Landing_Outcome, COUNT(*) AS Outcome_Count FROM SPACEXTBL WHE
```

```
 * sqlite:///my_data1.db
Done.
```

Out[28]:

Landing_Outcome	Outcome_Count
No attempt	10
Success (drone ship)	5
Failure (drone ship)	5
Success (ground pad)	3
Controlled (ocean)	3
Uncontrolled (ocean)	2
Failure (parachute)	2
Precluded (drone ship)	1

Reference Links

- Hands-on Lab : String Patterns, Sorting and Grouping
- Hands-on Lab: Built-in functions
- Hands-on Lab : Sub-queries and Nested SELECT Statements
- Hands-on Tutorial: Accessing Databases with SQL magic
- Hands-on Lab: Analyzing a real World Data Set

Author(s)

Lakshmi Holla

Other Contributors

Rav Ahuja