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
```

```
Collecting sqlalchemy==1.3.9
  Downloading SQLAlchemy-1.3.9.tar.gz (6.0 MB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 6.0/6.0 MB 102.3 MB/s eta 0:00:00
  Preparing metadata (setup.py) ... one
Building wheels for collected packages: sqlalchemy
  Building wheel for sqlalchemy (setup.py) ...done
  Created wheel for sqlalchemy: filename=SQLAlchemy-1.3.9-cp312-cp312-linux_x86_64.w
hl size=1160111 sha256=682a5e2a1ac305a3bc3d6f9d2df682fc366dfdfddacf26a685353647b0295
a30
  Stored in directory: /home/jupyterlab/.cache/pip/wheels/b3/1c/42/0e26b8d512adc6bce
10ff71a05229366b4ccec641cd3b42111
Successfully built sqlalchemy
Installing collected packages: sqlalchemy
  Attempting uninstall: sqlalchemy
    Found existing installation: SQLAlchemy 2.0.37
    Uninstalling SQLAlchemy-2.0.37:
      Successfully uninstalled SQLAlchemy-2.0.37
ERROR: pip's dependency resolver does not currently take into account all the packag
es that are installed. This behaviour is the source of the following dependency conf
licts.
jupyterhub 5.2.1 requires SQLAlchemy>=1.4.1, but you have sqlalchemy 1.3.9 which is
incompatible.
Successfully installed sqlalchemy-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
!pip install ipython-sql prettytable
```

```
Collecting ipython-sql
  Downloading ipython_sql-0.5.0-py3-none-any.whl.metadata (17 kB)
Collecting prettytable (from ipython-sql)
  Downloading prettytable-3.16.0-py3-none-any.whl.metadata (33 kB)
Requirement already satisfied: ipython in /opt/conda/lib/python3.12/site-packages (f
rom ipython-sql) (8.31.0)
Collecting sqlalchemy>=2.0 (from ipython-sql)
  Downloading sqlalchemy-2.0.43-cp312-cp312-manylinux_2_17_x86_64.manylinux2014_x86_
64.whl.metadata (9.6 kB)
Collecting sqlparse (from ipython-sql)
  Downloading sqlparse-0.5.3-py3-none-any.whl.metadata (3.9 kB)
Requirement already satisfied: six in /opt/conda/lib/python3.12/site-packages (from
ipython-sql) (1.17.0)
Requirement already satisfied: ipython-genutils in /opt/conda/lib/python3.12/site-pa
ckages (from ipython-sql) (0.2.0)
Requirement already satisfied: greenlet>=1 in /opt/conda/lib/python3.12/site-package
s (from sqlalchemy>=2.0->ipython-sql) (3.1.1)
Requirement already satisfied: typing-extensions>=4.6.0 in /opt/conda/lib/python3.1
2/site-packages (from sqlalchemy>=2.0->ipython-sql) (4.12.2)
Requirement already satisfied: decorator in /opt/conda/lib/python3.12/site-packages
(from ipython->ipython-sql) (5.1.1)
Requirement already satisfied: jedi>=0.16 in /opt/conda/lib/python3.12/site-packages
(from ipython->ipython-sql) (0.19.2)
Requirement already satisfied: matplotlib-inline in /opt/conda/lib/python3.12/site-p
ackages (from ipython->ipython-sql) (0.1.7)
Requirement already satisfied: pexpect>4.3 in /opt/conda/lib/python3.12/site-package
s (from ipython->ipython-sql) (4.9.0)
Requirement already satisfied: prompt_toolkit<3.1.0,>=3.0.41 in /opt/conda/lib/pytho
n3.12/site-packages (from ipython->ipython-sql) (3.0.50)
Requirement already satisfied: pygments>=2.4.0 in /opt/conda/lib/python3.12/site-pac
kages (from ipython->ipython-sql) (2.19.1)
Requirement already satisfied: stack_data in /opt/conda/lib/python3.12/site-packages
(from ipython->ipython-sql) (0.6.3)
Requirement already satisfied: traitlets>=5.13.0 in /opt/conda/lib/python3.12/site-p
ackages (from ipython->ipython-sql) (5.14.3)
Requirement already satisfied: wcwidth in /opt/conda/lib/python3.12/site-packages (f
rom prettytable->ipython-sql) (0.2.13)
Requirement already satisfied: parso<0.9.0,>=0.8.4 in /opt/conda/lib/python3.12/site
-packages (from jedi>=0.16->ipython->ipython-sql) (0.8.4)
Requirement already satisfied: ptyprocess>=0.5 in /opt/conda/lib/python3.12/site-pac
kages (from pexpect>4.3->ipython->ipython-sql) (0.7.0)
Requirement already satisfied: executing>=1.2.0 in /opt/conda/lib/python3.12/site-pa
ckages (from stack_data->ipython->ipython-sql) (2.1.0)
Requirement already satisfied: asttokens>=2.1.0 in /opt/conda/lib/python3.12/site-pa
ckages (from stack_data->ipython->ipython-sql) (3.0.0)
Requirement already satisfied: pure_eval in /opt/conda/lib/python3.12/site-packages
(from stack_data->ipython->ipython-sql) (0.2.3)
Downloading ipython_sql-0.5.0-py3-none-any.whl (20 kB)
Downloading sqlalchemy-2.0.43-cp312-cp312-manylinux_2_17_x86_64.manylinux2014_x86_6
4.whl (3.3 MB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 3.3/3.3 MB 106.4 MB/s eta 0:00:00
Downloading prettytable-3.16.0-py3-none-any.whl (33 kB)
Downloading sqlparse-0.5.3-py3-none-any.whl (44 kB)
Installing collected packages: sqlparse, sqlalchemy, prettytable, ipython-sql
  Attempting uninstall: sqlalchemy
    Found existing installation: SQLAlchemy 1.3.9
    Uninstalling SQLAlchemy-1.3.9:
      Successfully uninstalled SQLAlchemy-1.3.9
Successfully installed ipython-sql-0.5.0 prettytable-3.16.0 sqlalchemy-2.0.43 sqlpar
se-0.5.3
Requirement already satisfied: ipython-sql in /opt/conda/lib/python3.12/site-package
s (0.5.0)
Requirement already satisfied: prettytable in /opt/conda/lib/python3.12/site-package
s (3.16.0)
Requirement already satisfied: ipython in /opt/conda/lib/python3.12/site-packages (f
rom ipython-sql) (8.31.0)
Requirement already satisfied: sqlalchemy>=2.0 in /opt/conda/lib/python3.12/site-pac
kages (from ipython-sql) (2.0.43)
Requirement already satisfied: sqlparse in /opt/conda/lib/python3.12/site-packages
(from ipython-sql) (0.5.3)
Requirement already satisfied: six in /opt/conda/lib/python3.12/site-packages (from
ipython-sql) (1.17.0)
Requirement already satisfied: ipython-genutils in /opt/conda/lib/python3.12/site-pa
ckages (from ipython-sql) (0.2.0)
Requirement already satisfied: wcwidth in /opt/conda/lib/python3.12/site-packages (f
rom prettytable) (0.2.13)
Requirement already satisfied: greenlet>=1 in /opt/conda/lib/python3.12/site-package
s (from sqlalchemy>=2.0->ipython-sql) (3.1.1)
Requirement already satisfied: typing-extensions>=4.6.0 in /opt/conda/lib/python3.1
2/site-packages (from sqlalchemy>=2.0->ipython-sql) (4.12.2)
Requirement already satisfied: decorator in /opt/conda/lib/python3.12/site-packages
(from ipython->ipython-sql) (5.1.1)
Requirement already satisfied: jedi>=0.16 in /opt/conda/lib/python3.12/site-packages
(from ipython->ipython-sql) (0.19.2)
Requirement already satisfied: matplotlib-inline in /opt/conda/lib/python3.12/site-p
ackages (from ipython->ipython-sql) (0.1.7)
Requirement already satisfied: pexpect>4.3 in /opt/conda/lib/python3.12/site-package
s (from ipython->ipython-sql) (4.9.0)
Requirement already satisfied: prompt_toolkit<3.1.0,>=3.0.41 in /opt/conda/lib/pytho
n3.12/site-packages (from ipython->ipython-sql) (3.0.50)
Requirement already satisfied: pygments>=2.4.0 in /opt/conda/lib/python3.12/site-pac
kages (from ipython->ipython-sql) (2.19.1)
Requirement already satisfied: stack_data in /opt/conda/lib/python3.12/site-packages
(from ipython->ipython-sql) (0.6.3)
Requirement already satisfied: traitlets>=5.13.0 in /opt/conda/lib/python3.12/site-p
ackages (from ipython->ipython-sql) (5.14.3)
Requirement already satisfied: parso<0.9.0,>=0.8.4 in /opt/conda/lib/python3.12/site
-packages (from jedi>=0.16->ipython->ipython-sql) (0.8.4)
Requirement already satisfied: ptyprocess>=0.5 in /opt/conda/lib/python3.12/site-pac
kages (from pexpect>4.3->ipython->ipython-sql) (0.7.0)
Requirement already satisfied: executing>=1.2.0 in /opt/conda/lib/python3.12/site-pa
ckages (from stack_data->ipython->ipython-sql) (2.1.0)
Requirement already satisfied: asttokens>=2.1.0 in /opt/conda/lib/python3.12/site-pa
ckages (from stack_data->ipython->ipython-sql) (3.0.0)
Requirement already satisfied: pure_eval in /opt/conda/lib/python3.12/site-packages
(from stack_data->ipython->ipython-sql) (0.2.3)
```

In [3]:

```
%load_ext sql
```

In [4]:

```
import csv, sqlite3
import prettytable
prettytable.DEFAULT = 'DEFAULT'
```

```
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.appdomain.clou
df.to_sql("SPACEXTBL", con, if_exists='replace', index=False,method="multi")
```

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 not null
```

```
 * 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 [12]:
```
%%sql
SELECT DISTINCT launch_site FROM SPACEXTBL;
```

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

Out[12]:

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 [13]:

```
%%sql
SELECT * FROM SPACEXTBL WHERE launch_site LIKE 'CCA%' LIMIT 5;
```

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

Out[13]:

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

Task 3

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

In [14]:

```
%%sql
SELECT sum(payload_mass__kg_) AS "Total payload mass (NASA (CRS))" FROM SPACEXTBL W
```

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

Out[14]:

Total payload mass (NASA (CRS))
45596

Task 4

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

In [15]:

```
%%sql
SELECT AVG(payload_mass__kg_) AS "Average payload mass (booster version F9 v1.1)" F
```

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

Out[15]:

Average payload mass (booster version F9 v1.1)
2534.6666666666665

Task 5

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

Hint:Use min function

In [16]:

```
%%sql
SELECT min(DATE) AS "First successful landing outcome in ground pad" FROM SPACEXTBL
```

```
 * sqlite:///my_data1.db
(sqlite3.OperationalError) no such column: landing__outcome
[SQL: SELECT min(DATE) AS "First successful landing outcome in ground pad" FROM SPAC
EXTBL WHERE landing__outcome = 'Success (ground pad)';]
(Background on this error at: https://sqlalche.me/e/20/e3q8)
```

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 [17]:

```
%%sql
SELECT booster_version FROM SPACEXTBL WHERE landing__outcome = 'Success (drone ship
```

```
 * sqlite:///my_data1.db
(sqlite3.OperationalError) no such column: landing__outcome
[SQL: SELECT booster_version FROM SPACEXTBL WHERE landing__outcome = 'Success (drone
ship)' AND payload_mass__kg_ BETWEEN 4000 AND 6000;]
(Background on this error at: https://sqlalche.me/e/20/e3q8)
```

Task 7

List the total number of successful and failure mission outcomes

In [18]:

```
%%sql
SELECT 'Success' AS "Outcome", count(*) AS "Count" FROM SPACEXTBL WHERE landing__ou
```

```
 * sqlite:///my_data1.db
(sqlite3.OperationalError) no such column: landing__outcome
[SQL: SELECT 'Success' AS "Outcome", count(*) AS "Count" FROM SPACEXTBL WHERE landin
g__outcome LIKE 'Success%' UNION ALL SELECT 'Failure' AS "Outcome", count(*) AS "Cou
nt" FROM SPACEXTBL WHERE landing__outcome NOT LIKE 'Success%' UNION ALL SELECT '(Al
l)' AS "Outcome", count(*) AS "Count" FROM SPACEXTBL;]
(Background on this error at: https://sqlalche.me/e/20/e3q8)
```

Task 8

List all the booster_versions that have carried the maximum payload mass, using a subquery with a suitable aggregate function.

In [19]:
```
%%sql
SELECT DISTINCT booster_version FROM SPACEXTBL WHERE payload_mass__kg_ = ( SELECT m
```

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

Out[19]:

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 [20]:
```
%%sql
SELECT booster_version, launch_site,landing__outcome FROM SPACEXTBL WHERE landing__
```

```
 * sqlite:///my_data1.db
(sqlite3.OperationalError) near "FROM": syntax error
[SQL: SELECT booster_version, launch_site,landing__outcome FROM SPACEXTBL WHERE land
ing__outcome = 'Failure (drone ship)' AND EXTRACT(YEAR FROM DATE) = 2015]
(Background on this error at: https://sqlalche.me/e/20/e3q8)
```

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 [21]:

```
%%sql
SELECT landing__outcome, COUNT(*) AS "Count" FROM SPACEXTBL WHERE DATE BETWEEN '201
```

```
 * sqlite:///my_data1.db
(sqlite3.OperationalError) no such column: landing__outcome
[SQL: SELECT landing__outcome, COUNT(*) AS "Count" FROM SPACEXTBL WHERE DATE BETWEEN
'2010-06-04' and '2017-03-20' GROUP BY landing__outcome ORDER BY Count DESC ;]
(Background on this error at: https://sqlalche.me/e/20/e3q8)
```

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