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

```
!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 64.8 MB/s eta 0:00:00:00:0100:01
  Preparing metadata (setup.py) ... done
Building wheels for collected packages: sqlalchemy
  Building wheel for sqlalchemy (setup.py) ... done
  Created wheel for sqlalchemy: filename=SQLAlchemy-1.3.9-cp37-cp37m-linux_x86_64.whl size=1159121 sha256=8da1ae504b42e206ad1ab3050e2da6386dd20dba96dbe9a3b6
1e4111e4e19123
  Stored in directory: /home/jupyterlab/.cache/pip/wheels/03/71/13/010faf12246f72dc76b4150e6e599d13a85b4435e06fb9e51f
Successfully built sqlalchemy
Installing collected packages: sqlalchemy
  Attempting uninstall: sqlalchemy
    Found existing installation: SQLAlchemy 1.3.24
    Uninstalling SQLAlchemy-1.3.24:
      Successfully uninstalled SQLAlchemy-1.3.24
Successfully installed sqlalchemy-1.3.9
```

Connect to the database

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

```
#Please uncomment and execute the code below if you are working locally.

#!pip install ipython-sql
```

```
%load_ext sql
```

```
import csv, sqlite3

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

```
!pip install -q pandas==1.1.5
```

```
%sql sqlite:///my_data1.db
```

```
'Connected: @my_data1.db'
```

```
import pandas as pd
df = pd.read_csv("https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBM-DS0321EN-SkillsNetwork/labs/module_2/data/Spacex.csv")
df.to_sql("SPACEXTBL", con, if_exists='replace', index=False,method="multi")
```

```
/home/jupyterlab/conda/envs/python/lib/python3.7/site-packages/pandas/core/generic.py:2882: UserWarning: The spaces in these column names will not be change
d. In pandas versions < 0.14, spaces were converted to underscores.
  both result in 0.1234 being formatted as 0.12.
```

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

```
%sql create table SPACEXTABLE as select * from SPACEXTBL where Date is not null
```

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

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

```
%sql SELECT DISTINCT Launch_Site from SPACEXTABLE
```

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

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'

```sql
%%sql
select* from SPACEXTABLE WHERE Launch_Site LIKE 'CCA%' limit 5
```

 * sqlite:///my_data1.db
Done.

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing_Outcome
2010-06-04	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (parachute)
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)	NASA (COTS) NRO	Success	Failure (parachute)
2012-05-22	7:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt
2012-10-08	0:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	No attempt
2013-03-01	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	No attempt

Task 3

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

```sql
%%sql
select sum(payload_mass__kg_) from SPACEXTABLE  where customer LIKE '%CRS%'
```

 * sqlite:///my_data1.db
Done.

sum(payload_mass__kg_)
48213

Task 4

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

```sql
%%sql
select avg(payload_mass__kg_) from SPACEXTABLE  where booster_version='F9 v1.1'
```

 * sqlite:///my_data1.db
Done.

avg(payload_mass__kg_)
2928.4

Task 5

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

Hint:Use min function

```sql
%%sql
select *  from  SPACEXTABLE
```

 * sqlite:///my_data1.db
Done.

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing_Outcome
2010-06-04	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (parachute)
2010-12-08	15:43:00	F9 v1.0 B0004	CCAFS LC-40	Dragon demo flight C1, two CubeSats, barrel of Brouere	0	LEO (ISS)	NASA (COTS) NRO	Success	Failure (parachute)

```
%%sql
```
```
select * from  SPACEXTABLE where Landing_Outcome="Success (ground pad)" ;
```

* sqlite:///my_data1.db
Done.

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing_Outcome
2015-12-22	1:29:00	F9 FT B1019	CCAFS LC-40	OG2 Mission 2 11 Orbcomm-OG2 satellites	2034	LEO	Orbcomm	Success	Success (ground pad)
2016-07-18	4:45:00	F9 FT B1025.1	CCAFS LC-40	SpaceX CRS-9	2257	LEO (ISS)	NASA (CRS)	Success	Success (ground pad)
2017-02-19	14:39:00	F9 FT B1031.1	KSC LC-39A	SpaceX CRS-10	2490	LEO (ISS)	NASA (CRS)	Success	Success (ground pad)
2017-05-01	11:15:00	F9 FT B1032.1	KSC LC-39A	NROL-76	5300	LEO	NRO	Success	Success (ground pad)
2017-						LEO	NASA		Success (ground

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

```
%%sql
select Booster_Version from SPACEXTABLE
        where Landing_Outcome="Success (ground pad)" and PAYLOAD_MASS__KG_>4000 and PAYLOAD_MASS__KG_<6000
```

* sqlite:///my_data1.db
Done.

Booster_Version
F9 FT B1032.1
F9 B4 B1040.1
F9 B4 B1043.1

Task 7 ¶

List the total number of successful and failure mission outcomes

```
%%sql
SELECT Count(mission_outcome) from SPACEXTABLE where mission_outcome LIKE '%Success%'
```

* sqlite:///my_data1.db
Done.

Count(mission_outcome)
100

```
%%sql
SELECT Count(mission_outcome) from SPACEXTABLE where mission_outcome LIKE '%Failure%'
```

* sqlite:///my_data1.db
Done.

Count(mission_outcome)
1

Task 8

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

```sql
%%sql

SELECT  booster_version FROM SPACEXTABLE  where  payload_mass__kg_ = (Select Max(payload_mass__kg_) from SPACEXTABLE)
```

* sqlite:///my_data1.db
Done.

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.

```sql
%%sql
select booster_version, launch_site, Landing_Outcome,DATE from SPACEXTABLE where Landing_Outcome LIKE '%Failure (drone ship)%'
```

* sqlite:///my_data1.db
Done.

Booster_Version	Launch_Site	Landing_Outcome	Date
F9 v1.1 B1012	CCAFS LC-40	Failure (drone ship)	2015-01-10
F9 v1.1 B1015	CCAFS LC-40	Failure (drone ship)	2015-04-14
F9 v1.1 B1017	VAFB SLC-4E	Failure (drone ship)	2016-01-17
F9 FT B1020	CCAFS LC-40	Failure (drone ship)	2016-03-04
F9 FT B1024	CCAFS LC-40	Failure (drone ship)	2016-06-15

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. ¶

```sql
%%sql
select Landing_Outcome, Date, count(Landing_Outcome) from SPACEXTABLE  where DATE>'2010-06-04' AND DATE<='2017-03-20' Group by Landing_Outcome
```

* sqlite:///my_data1.db
Done.

Landing_Outcome	Date	count(Landing_Outcome)
Controlled (ocean)	2014-04-18	3
Failure (drone ship)	2015-01-10	5
Failure (parachute)	2010-12-08	1
No attempt	2012-05-22	10
Precluded (drone ship)	2015-06-28	1
Success (drone ship)	2016-04-08	5
Success (ground pad)	2015-12-22	3
Uncontrolled (ocean)	2013-09-29	2