Space X Falcon 9 First Stage Landing Prediction

EDA with SQL

-Understand the Spacex DataSet

-Load the dataset into the corresponding table in a Db2 database

-Execute SQL queries to answer assignment questions

```
import sqlite3
import pandas as pd
print(sqlite3.version)
print(sqlite3.sqlite_version)
```

```
2.6.0
3.47.0
```

```
# get dataset into pandas dataframe
url = "https://cf-courses-data.s3.us.cloud-object-
storage.appdomain.cloud/IBM-DS0321EN-SkillsNetwork/labs/module_2/
data/Spacex.csv?
utm_medium=Exinfluencer&utm_source=Exinfluencer&utm_content=000026UJ&u
tm_term=10006555&utm_id=NA-SkillsNetwork-Channel-
SkillsNetworkCoursesIBMDS0321ENSkillsNetwork26802033-2022-01-01"
df = pd.read_csv(url)
# remove spaces in columns name
df.columns = df.columns.str.replace(' ','_')

# convert timestamp strings to date and time format
# df['Date'] = pd.to_datetime(df['Date'], errors="coerce", format="%d-
%m-%Y")

print(df.dtypes)
print(df.head())
```

```
Date                  object
Time_(UTC)            object
Booster_Version       object
Launch_Site           object
Payload               object
PAYLOAD_MASS__KG_      int64
Orbit                 object
Customer              object
Mission_Outcome       object
Landing_Outcome       object
dtype: object
```

```
         Date Time_(UTC) Booster_Version  Launch_Site  \
0  2010-06-04   18:45:00  F9 v1.0  B0003  CCAFS LC-40
1  2010-12-08   15:43:00  F9 v1.0  B0004  CCAFS LC-40
2  2012-05-22    7:44:00  F9 v1.0  B0005  CCAFS LC-40
3  2012-10-08    0:35:00  F9 v1.0  B0006  CCAFS LC-40
4  2013-03-01   15:10:00  F9 v1.0  B0007  CCAFS LC-40

                                              Payload
PAYLOAD_MASS__KG_  \
0              Dragon Spacecraft Qualification Unit
0
1  Dragon demo flight C1, two CubeSats, barrel of...
0
2                             Dragon demo flight C2
525
3                                       SpaceX CRS-1
500
4                                       SpaceX CRS-2
677

       Orbit         Customer Mission_Outcome      Landing_Outcome
0        LEO           SpaceX         Success  Failure (parachute)
1  LEO (ISS)  NASA (COTS) NRO         Success  Failure (parachute)
2  LEO (ISS)      NASA (COTS)         Success           No attempt
3  LEO (ISS)       NASA (CRS)         Success           No attempt
4  LEO (ISS)       NASA (CRS)         Success           No attempt
```

```
# create sqlite table and upload data into it
conn = sqlite3.connect(':memory:')  # in memory database
df.to_sql(name="spacexdata", con=conn, if_exists="replace")

query = pd.read_sql('select * from spacexdata', conn)
query
```

```
     index        Date Time_(UTC) Booster_Version   Launch_Site  \
0        0  2010-06-04   18:45:00  F9 v1.0  B0003   CCAFS LC-40
1        1  2010-12-08   15:43:00  F9 v1.0  B0004   CCAFS LC-40
2        2  2012-05-22    7:44:00  F9 v1.0  B0005   CCAFS LC-40
3        3  2012-10-08    0:35:00  F9 v1.0  B0006   CCAFS LC-40
4        4  2013-03-01   15:10:00  F9 v1.0  B0007   CCAFS LC-40
..     ...         ...        ...              ...           ...
96      96  2020-11-05   23:24:23    F9 B5B1062.1  CCAFS SLC-40
97      97  2020-11-16    0:27:00   F9 B5B1061.1     KSC LC-39A
98      98  2020-11-21   17:17:08    F9 B5B1063.1   VAFB SLC-4E
99      99  2020-11-25    2:13:00  F9 B5 B1049.7   CCAFS SLC-40
100    100  2020-12-06   16:17:08  F9 B5 B1058.4     KSC LC-39A

                                                Payload
PAYLOAD_MASS__KG_  \
0                  Dragon Spacecraft Qualification Unit
```

```
0
1    Dragon demo flight C1, two CubeSats, barrel of...
0
2                                 Dragon demo flight C2
525
3                                          SpaceX CRS-1
500
4                                          SpaceX CRS-2
677
..                                                   ...
...
96                                GPS III-04 , Crew-1
4311
97                Crew-1, Sentinel-6 Michael Freilich
12500
98      Sentinel-6 Michael Freilich, Starlink 15 v1.0
1192
99                    Starlink 15 v1.0, SpaceX CRS-21
15600
100                                        SpaceX CRS-21
2972

         Orbit                        Customer Mission_Outcome  \
0          LEO                          SpaceX         Success
1    LEO (ISS)                NASA (COTS) NRO         Success
2    LEO (ISS)                    NASA (COTS)         Success
3    LEO (ISS)                     NASA (CRS)         Success
4    LEO (ISS)                     NASA (CRS)         Success
..         ...                            ...             ...
96         MEO                           USSF         Success
97   LEO (ISS)                     NASA (CCP)         Success
98         LEO  NASA / NOAA / ESA / EUMETSAT         Success
99         LEO                          SpaceX         Success
100  LEO (ISS)                     NASA (CRS)         Success

         Landing_Outcome
0    Failure (parachute)
1    Failure (parachute)
2             No attempt
3             No attempt
4             No attempt
..                   ...
96               Success
97               Success
98               Success
99               Success
100              Success

[101 rows x 11 columns]
```

```
# create sqlite table and upload data into it
conn = sqlite3.connect(':memory:')  # in memory database
df.to_sql(name="spacexdata", con=conn, if_exists="replace")

q = pd.read_sql('select * from spacexdata', conn)
q
```

```
     index        Date Time_(UTC) Booster_Version   Launch_Site  \
0        0  2010-06-04   18:45:00  F9 v1.0  B0003   CCAFS LC-40
1        1  2010-12-08   15:43:00  F9 v1.0  B0004   CCAFS LC-40
2        2  2012-05-22    7:44:00  F9 v1.0  B0005   CCAFS LC-40
3        3  2012-10-08    0:35:00  F9 v1.0  B0006   CCAFS LC-40
4        4  2013-03-01   15:10:00  F9 v1.0  B0007   CCAFS LC-40
..     ...         ...        ...              ...           ...
96      96  2020-11-05   23:24:23    F9 B5B1062.1  CCAFS SLC-40
97      97  2020-11-16    0:27:00   F9 B5B1061.1     KSC LC-39A
98      98  2020-11-21   17:17:08    F9 B5B1063.1   VAFB SLC-4E
99      99  2020-11-25    2:13:00  F9 B5 B1049.7   CCAFS SLC-40
100    100  2020-12-06   16:17:08  F9 B5 B1058.4     KSC LC-39A

                                                  Payload
PAYLOAD_MASS__KG_  \
0                 Dragon Spacecraft Qualification Unit
0
1    Dragon demo flight C1, two CubeSats, barrel of...
0
2                                Dragon demo flight C2
525
3                                          SpaceX CRS-1
500
4                                          SpaceX CRS-2
677
..                                                   ...
...
96                                 GPS III-04 , Crew-1
4311
97                Crew-1, Sentinel-6 Michael Freilich
12500
98      Sentinel-6 Michael Freilich, Starlink 15 v1.0
1192
99                    Starlink 15 v1.0, SpaceX CRS-21
15600
100                                        SpaceX CRS-21
2972

         Orbit                       Customer Mission_Outcome  \
0          LEO                         SpaceX         Success
1    LEO (ISS)                NASA (COTS) NRO         Success
2    LEO (ISS)                    NASA (COTS)         Success
3    LEO (ISS)                     NASA (CRS)         Success
```

```
4    LEO (ISS)                   NASA (CRS)         Success
..         ...                         ...             ...
96         MEO                        USSF         Success
97   LEO (ISS)                   NASA (CCP)         Success
98         LEO  NASA / NOAA / ESA / EUMETSAT         Success
99         LEO                      SpaceX         Success
100  LEO (ISS)                   NASA (CRS)         Success

         Landing_Outcome
0    Failure (parachute)
1    Failure (parachute)
2             No attempt
3             No attempt
4             No attempt
..                   ...
96               Success
97               Success
98               Success
99               Success
100              Success

[101 rows x 11 columns]
```

Task 1

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

```
query = pd.read_sql('select distinct Launch_Site from spacexdata',
conn)
query
```

```
    Launch_Site
0   CCAFS LC-40
1   VAFB SLC-4E
2    KSC LC-39A
3  CCAFS SLC-40
```

Task 2

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

```
query = pd.read_sql("select * from spacexdata where Launch_Site like
'CCA%' limit 5", conn)
query
```

```
   index        Date Time_(UTC) Booster_Version  Launch_Site  \
0      0  2010-06-04   18:45:00  F9 v1.0  B0003  CCAFS LC-40
```

```
1      1  2010-12-08   15:43:00  F9 v1.0  B0004  CCAFS LC-40
2      2  2012-05-22    7:44:00  F9 v1.0  B0005  CCAFS LC-40
3      3  2012-10-08    0:35:00  F9 v1.0  B0006  CCAFS LC-40
4      4  2013-03-01   15:10:00  F9 v1.0  B0007  CCAFS LC-40

                                                  Payload
PAYLOAD_MASS__KG_  \
0              Dragon Spacecraft Qualification Unit
0
1  Dragon demo flight C1, two CubeSats, barrel of...
0
2                               Dragon demo flight C2
525
3                                    SpaceX CRS-1
500
4                                    SpaceX CRS-2
677

       Orbit         Customer Mission_Outcome      Landing_Outcome
0        LEO           SpaceX         Success  Failure (parachute)
1  LEO (ISS)  NASA (COTS) NRO         Success  Failure (parachute)
2  LEO (ISS)      NASA (COTS)         Success           No attempt
3  LEO (ISS)       NASA (CRS)         Success           No attempt
4  LEO (ISS)       NASA (CRS)         Success           No attempt
```

Task 3

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

```
query = pd.read_sql("select sum(PAYLOAD_MASS__KG_) from spacexdata
where Customer='NASA (CRS)'", conn)
query
```

```
   sum(PAYLOAD_MASS__KG_)
0                   45596
```

Task 4

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

```
q = pd.read_sql("select avg(PAYLOAD_MASS__KG_) from spacexdata where
Booster_Version='F9 v1.1'", conn)
q
```

```
   avg(PAYLOAD_MASS__KG_)
0                  2928.4
```

Task 5

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

Hint:Use min function

```
q = pd.read_sql("select min(Date) from spacexdata where
Landing_Outcome='Success (ground pad)'", conn)
q
```

```
    min(Date)
0  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

```
q = pd.read_sql("select distinct(Booster_Version) from spacexdata
where Landing_Outcome='Success (drone ship)' and PAYLOAD_MASS__KG_
between 4000 and 6000", conn)
q
```

```
  Booster_Version
0     F9 FT B1022
1     F9 FT B1026
2  F9 FT  B1021.2
3  F9 FT  B1031.2
```

Task 7

List the total number of successful and failure mission outcomes¶

```
q = pd.read_sql("select substr(Mission_Outcome,1,7) as
Mission_Outcome, count(*) from spacexdata  group by 1", conn)
q
```

```
  Mission_Outcome  count(*)
0         Failure         1
1         Success       100
```

Task 8

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

```
q = pd.read_sql("select distinct Booster_Version from spacexdata where
PAYLOAD_MASS__KG_ = (select max(PAYLOAD_MASS__KG_) from spacexdata)",
conn)
q
```

```
   Booster_Version
0    F9 B5 B1048.4
1    F9 B5 B1049.4
2    F9 B5 B1051.3
3    F9 B5 B1056.4
4    F9 B5 B1048.5
5    F9 B5 B1051.4
6    F9 B5 B1049.5
7   F9 B5 B1060.2
8   F9 B5 B1058.3
9    F9 B5 B1051.6
10   F9 B5 B1060.3
11  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.

```
q = pd.read_sql("select distinct(Landing_Outcome), Booster_Version,
Launch_Site from spacexdata where Landing_Outcome='Failure (drone
ship)'", conn)
q
```

```
        Landing_Outcome Booster_Version  Launch_Site
0  Failure (drone ship)   F9 v1.1 B1012  CCAFS LC-40
1  Failure (drone ship)   F9 v1.1 B1015  CCAFS LC-40
2  Failure (drone ship)   F9 v1.1 B1017  VAFB SLC-4E
3  Failure (drone ship)     F9 FT B1020  CCAFS LC-40
4  Failure (drone ship)     F9 FT B1024  CCAFS LC-40
```

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.

```
q = pd.read_sql("select Landing_Outcome, count(*) from spacexdata
where Date between '2011-06-04' and '2017-03-20' group by
Landing_Outcome order by 2 desc", conn)
q
```

```
          Landing_Outcome  count(*)
0              No attempt        10
1    Success (drone ship)         5
2    Failure (drone ship)         5
3    Success (ground pad)         3
```

```
4      Controlled (ocean)         3
5    Uncontrolled (ocean)         2
6  Precluded (drone ship)         1
```