Applied data Science Capstone

Part 1

By Tapiwa Terrence Munodawafa

27/07/2025

OUTLINE

- Executive Summary
- Introduction
- Methodology
- Results
  - Visualization – Charts
  - Dashboard
- Discussion
  - Findings & Implications
- Conclusion
- Appendix

EXECUTIVE SUMMARY

- Several APIs were used for:
  - Data collection
  - Web scrapping
- EDA was performed
- ML techniques were applied fore prediction

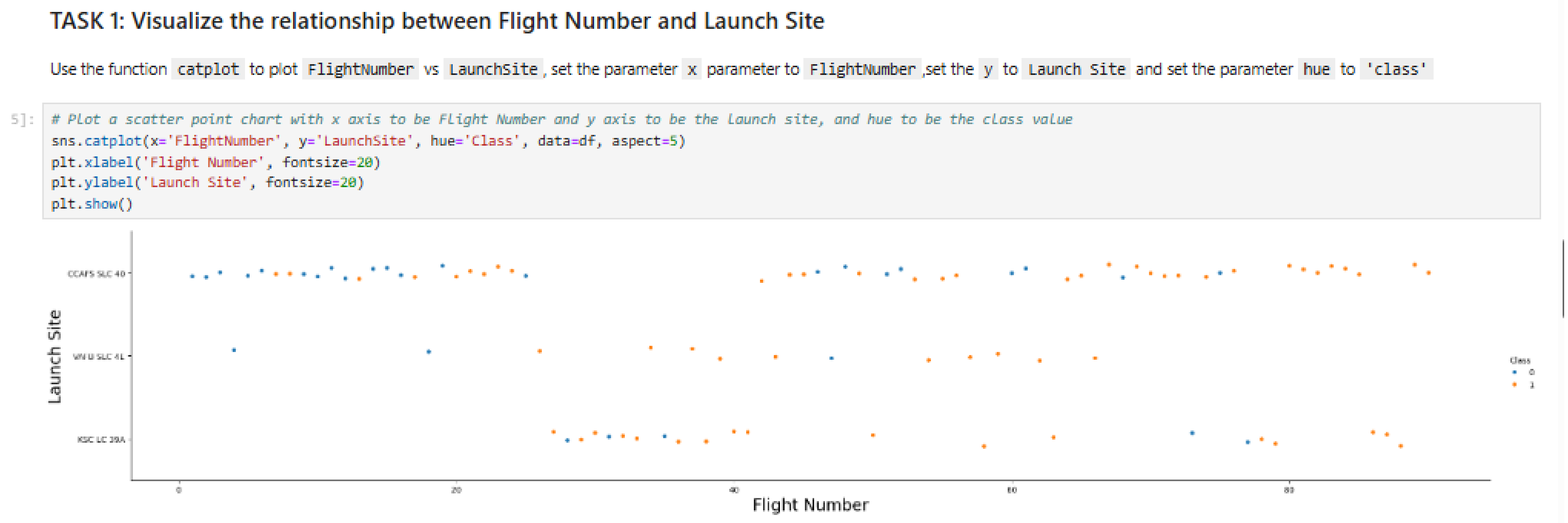
INTRODUCTION

- Prediction of Falcon 9’s first stage successful landing
- This data is crucial for performance enhancement and safety
- Exploratory Data Analysis (EDA) was also made
  - SQL
  - Visualization

METHODOLOGY

- Data collection
  - SpaceX API
  - Web scrapping
- Data wrangling
  - Dropping irrelevant columns
- EDA using SQL and visualization
- Interactive visual analytics using Plotly Dash and Folium
- Performed prediction using classification models

RESULTS – EDA Visualization

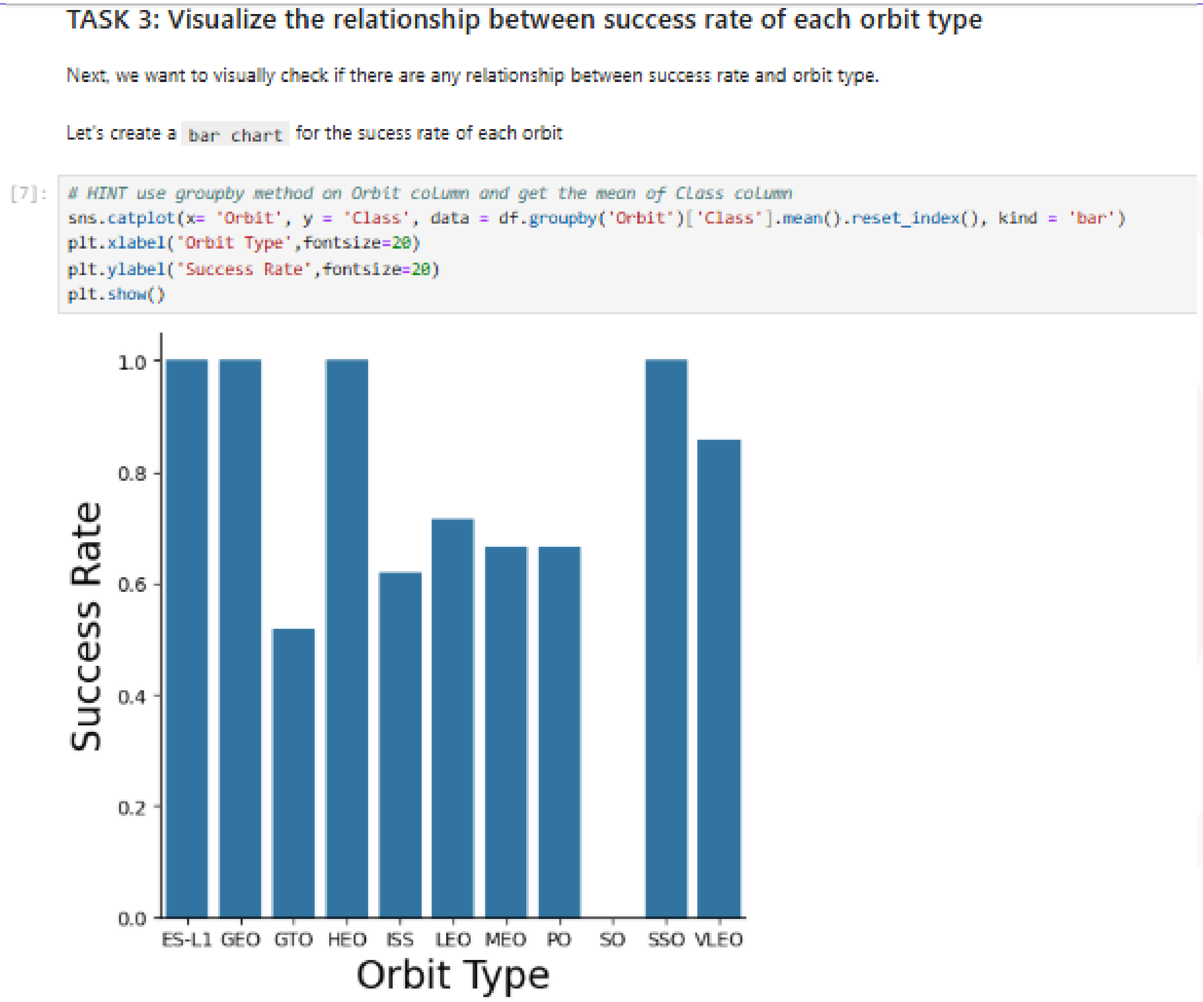


TASK 1: Visualize the relationship between Flight Number and Launch Site

Use the function `catplot` to plot `FlightNumber` vs `LaunchSite`, set the parameter `x` parameter to `FlightNumber`,set the `y` to `Launch Site` and set the parameter `hue` to `'class'`

```
5]: # Plot a scatter point chart with x axis to be Flight Number and y axis to be the launch site, and hue to be the class value
    sns.catplot(x='FlightNumber', y='LaunchSite', hue='Class', data=df, aspect=5)
    plt.xlabel('Flight Number', fontsize=20)
    plt.ylabel('Launch Site', fontsize=20)
    plt.show()
```

RESULTS – EDA Visualization



TASK 3: Visualize the relationship between success rate of each orbit type

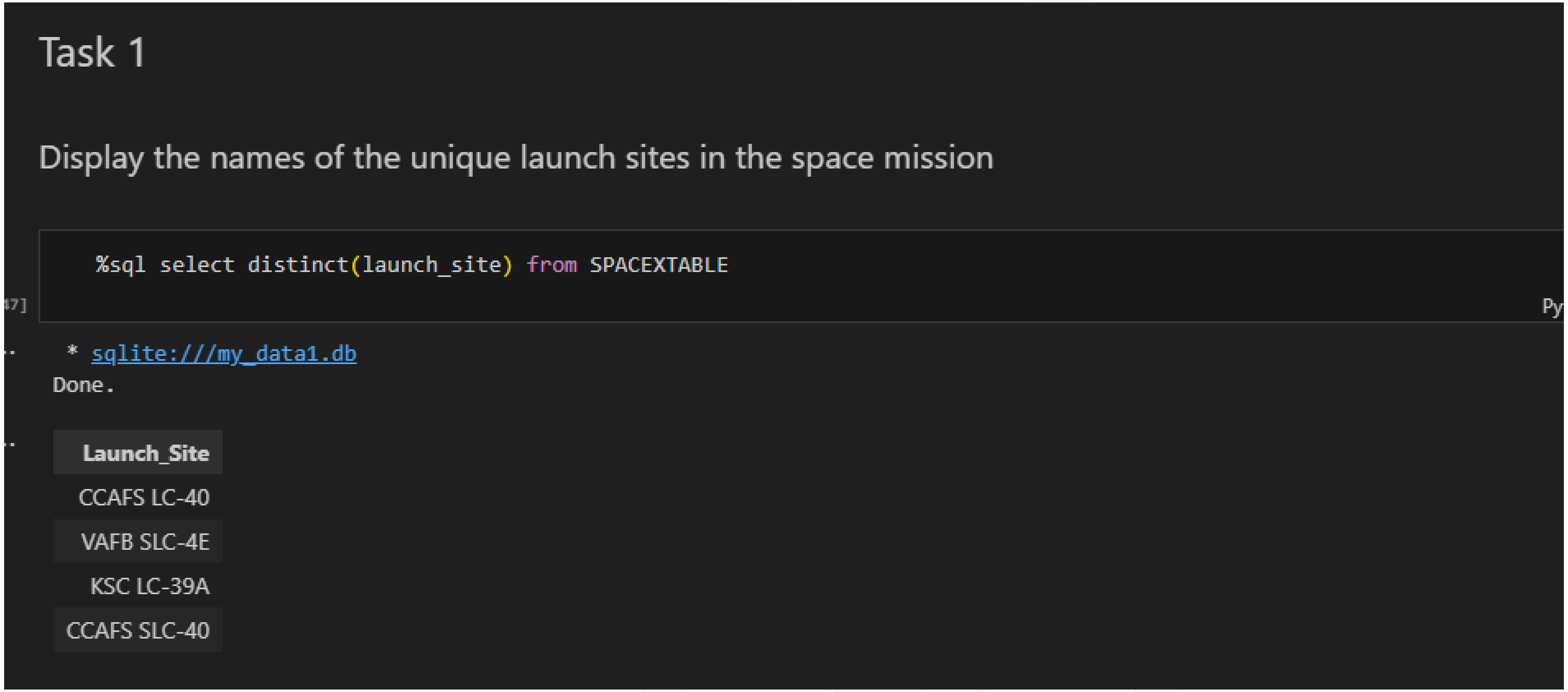
Next, we want to visually check if there are any relationship between success rate and orbit type.

Let's create a `bar chart` for the sucess rate of each orbit

[7]:

```
# HINT use groupby method on Orbit column and get the mean of Class column
sns.catplot(x= 'Orbit', y = 'Class', data = df.groupby('Orbit')['Class'].mean().reset_index(), kind = 'bar')
plt.xlabel('Orbit Type',fontsize=20)
plt.ylabel('Success Rate',fontsize=20)
plt.show()
```

RESULTS – Sample SQL



Task 1

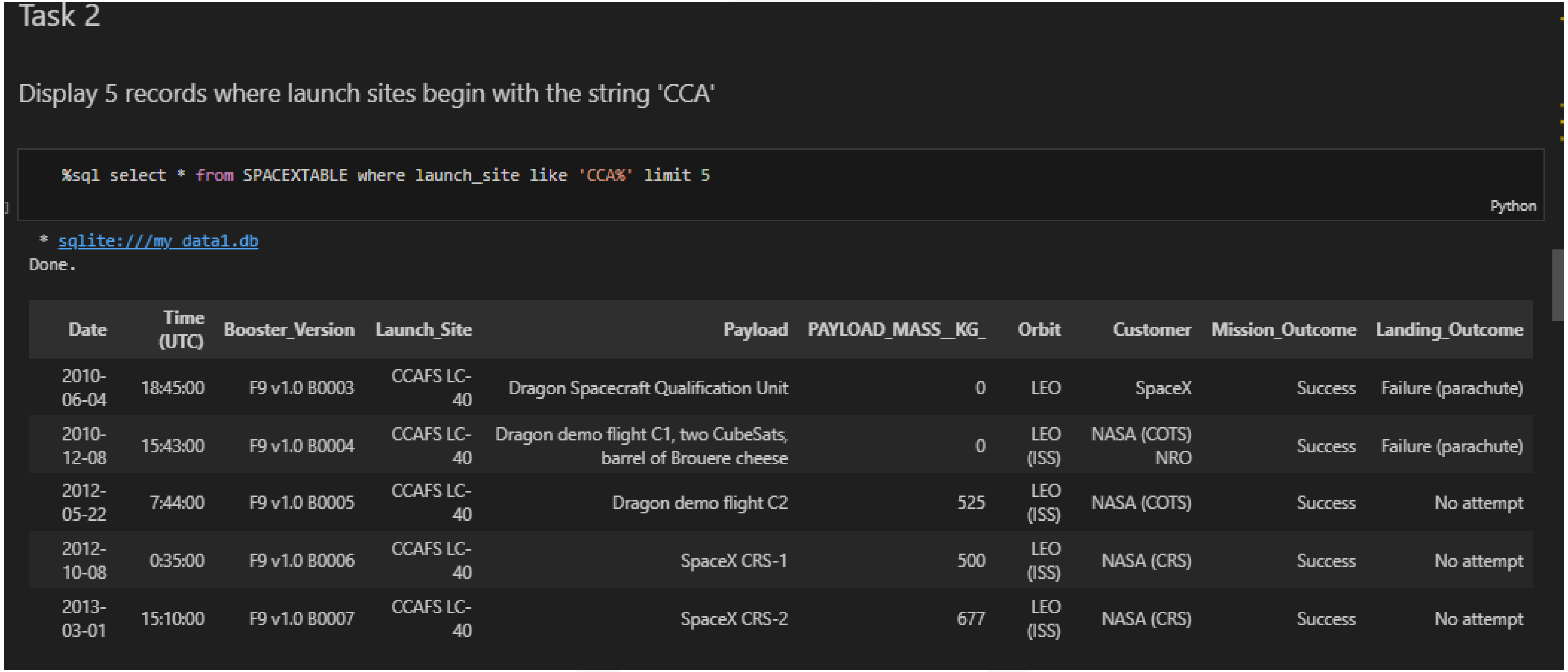
Display the names of the unique launch sites in the space mission

```
%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

RESULTS – Sample SQL



Task 2

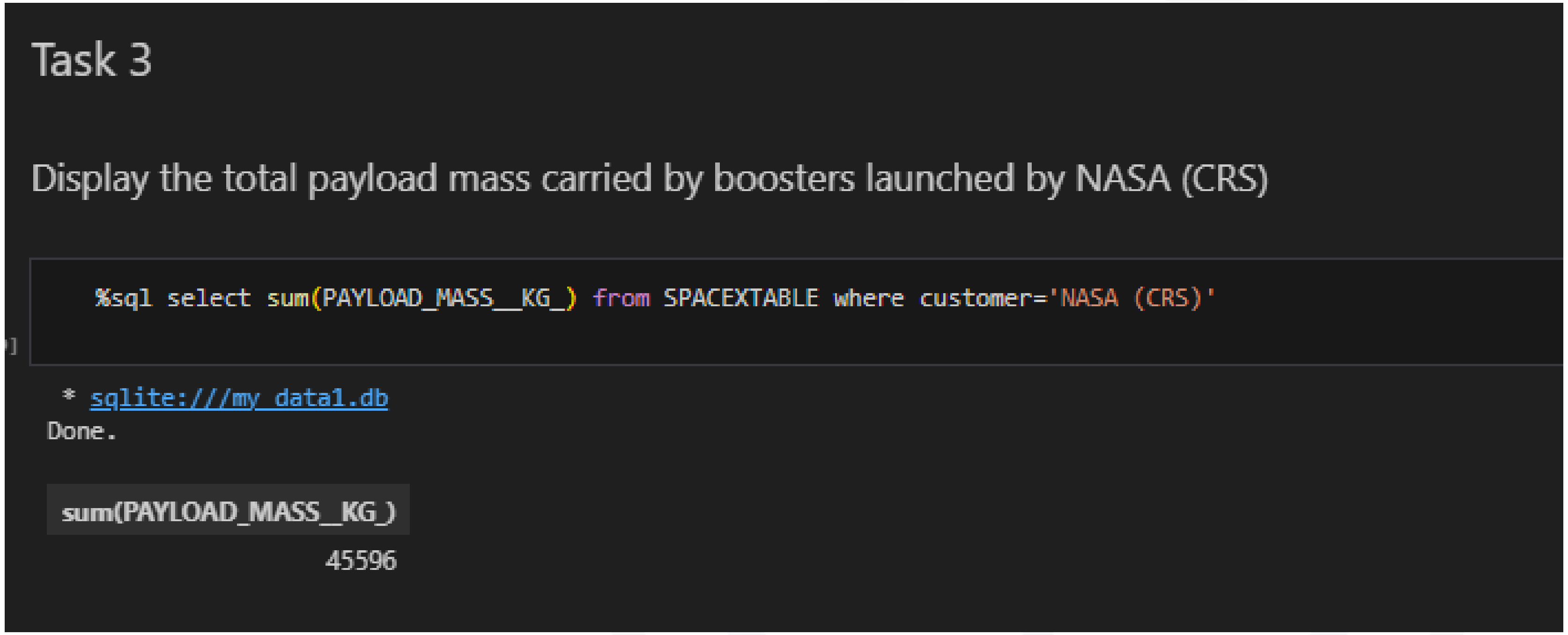
Display 5 records where launch sites begin with the string 'CCA'

```
%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

RESULTS – Sample SQL



Task 3

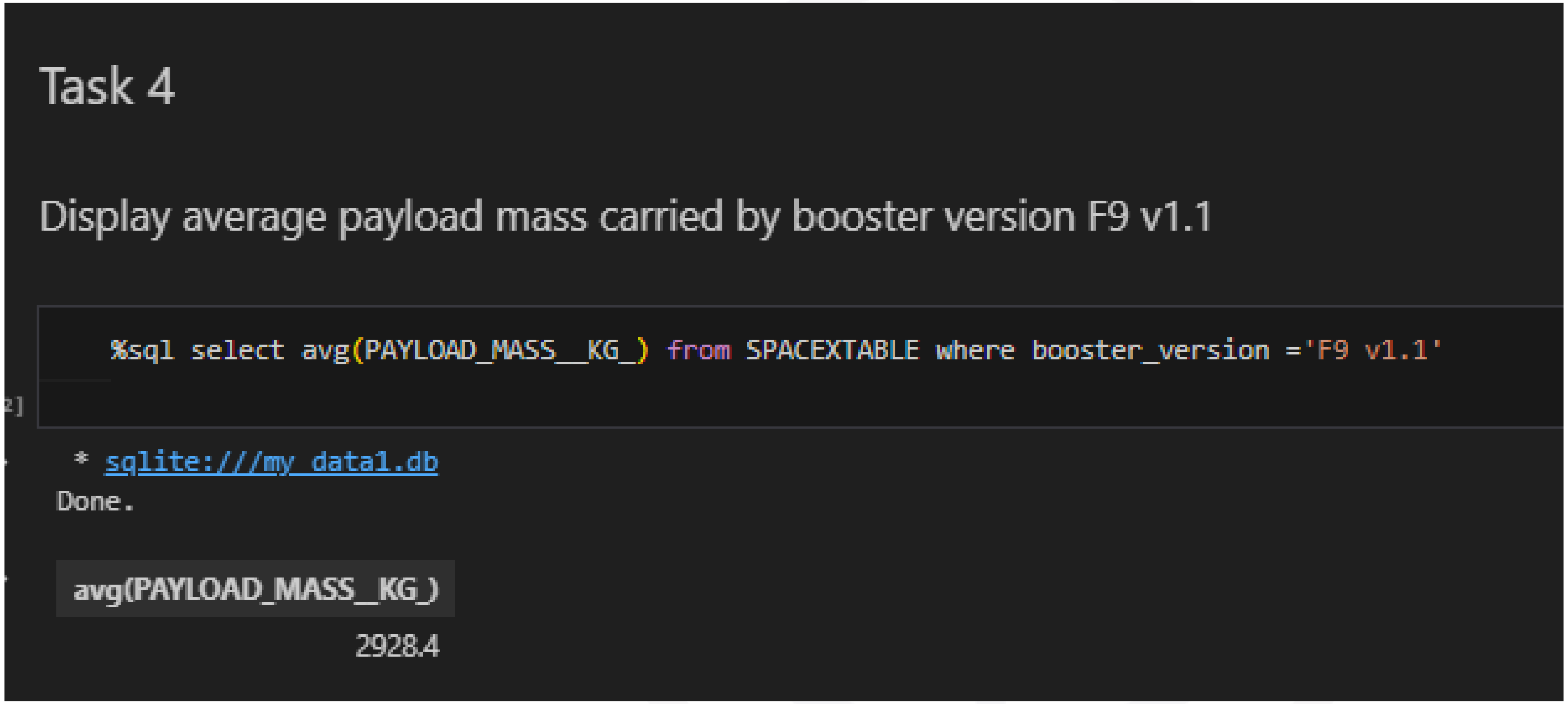
Display the total payload mass carried by boosters launched by NASA (CRS)

```
%sql select sum(PAYLOAD_MASS__KG_) from SPACEXTABLE where customer='NASA (CRS)'
```

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

sum(PAYLOAD_MASS__KG_)
45596

RESULTS – Sample SQL



Task 4

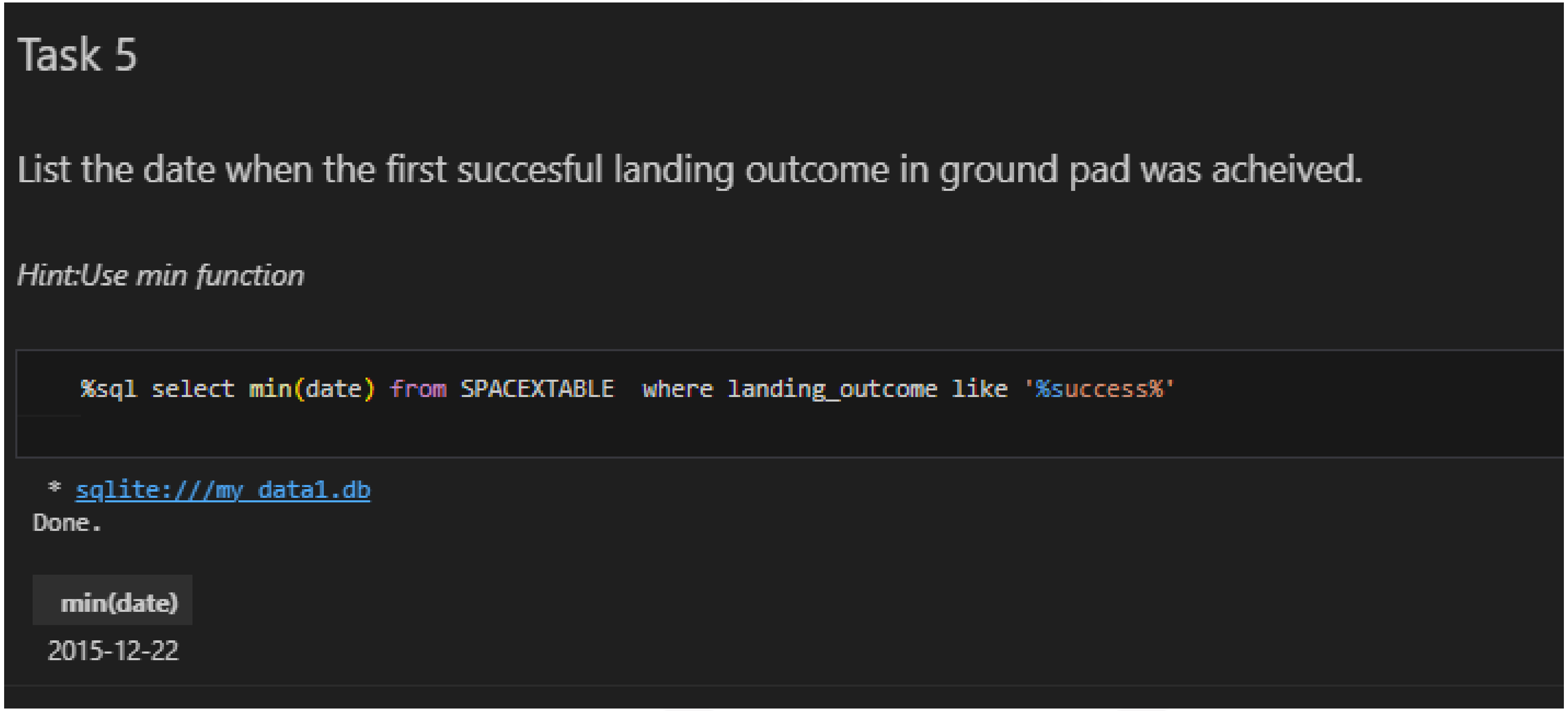
Display average payload mass carried by booster version F9 v1.1

```
%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

RESULTS – Sample SQL



Task 5

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

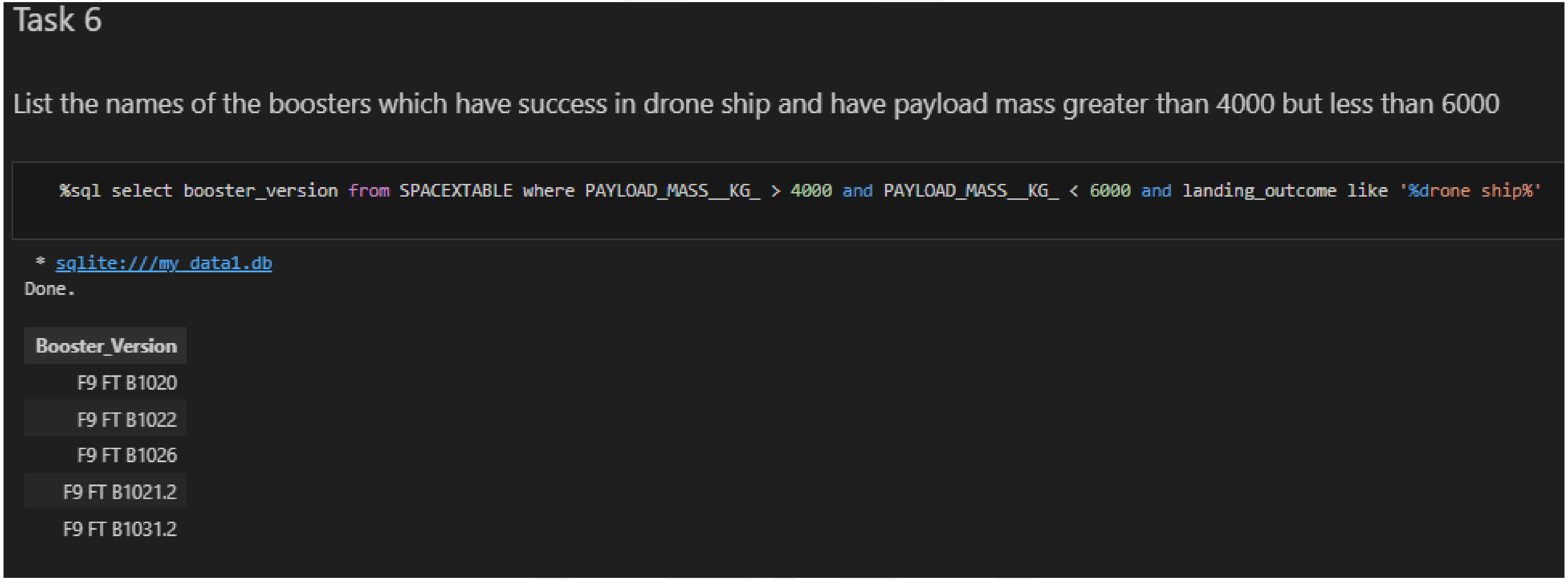
Hint:Use min function

```
%sql select min(date) from SPACEXTABLE  where landing_outcome like '%success%'
```

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

min(date)
2015-12-22

RESULTS – Sample SQL



Task 6

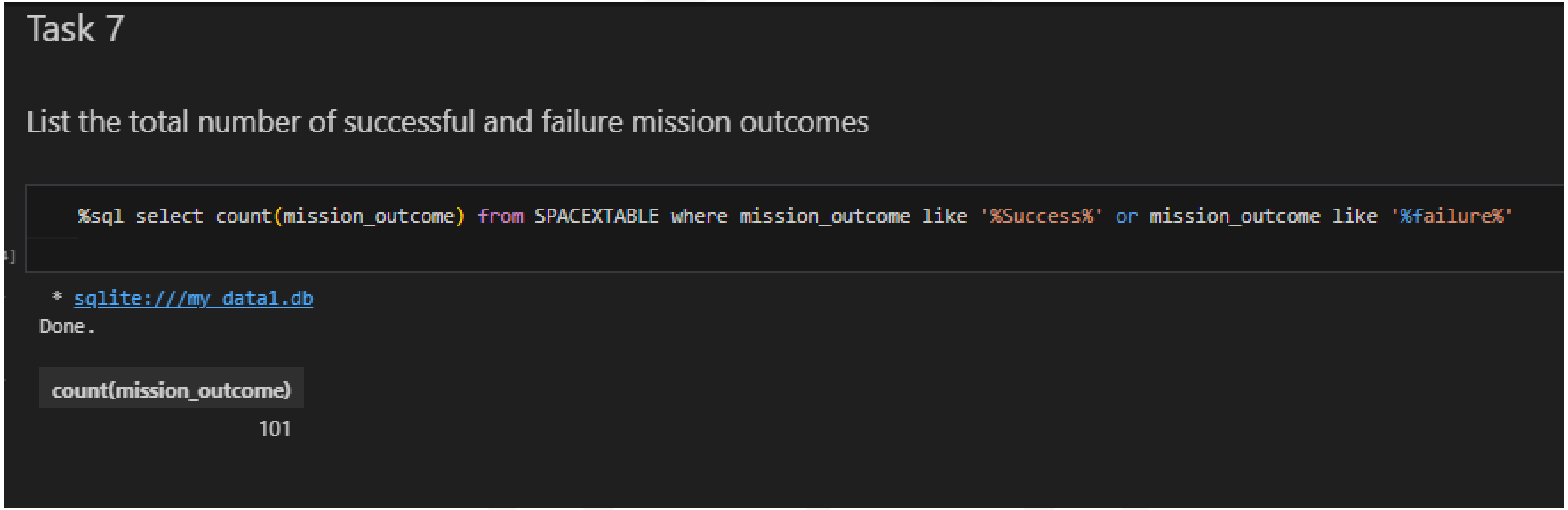
List the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000

```
%sql select booster_version from SPACEXTABLE where PAYLOAD_MASS__KG_ > 4000 and PAYLOAD_MASS__KG_ < 6000 and landing_outcome like '%drone ship%'
```

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

Booster_Version
F9 FT B1020
F9 FT B1022
F9 FT B1026
F9 FT B1021.2
F9 FT B1031.2

RESULTS – Sample SQL



Task 7

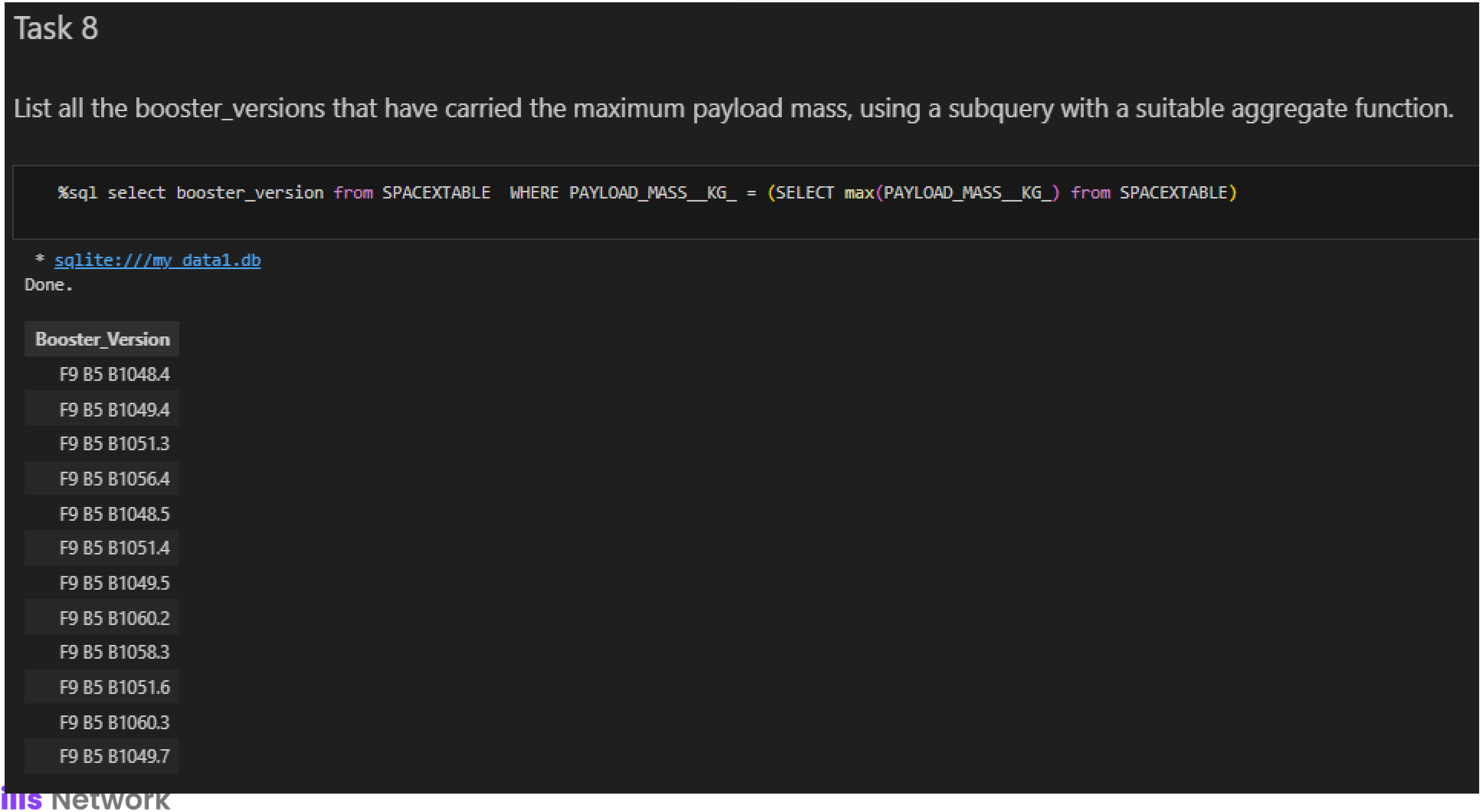
List the total number of successful and failure mission outcomes

```
%sql select count(mission_outcome) from SPACEXTABLE where mission_outcome like '%Success%' or mission_outcome like '%failure%'
```

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

count(mission_outcome)
101

RESULTS – Sample SQL



Task 8

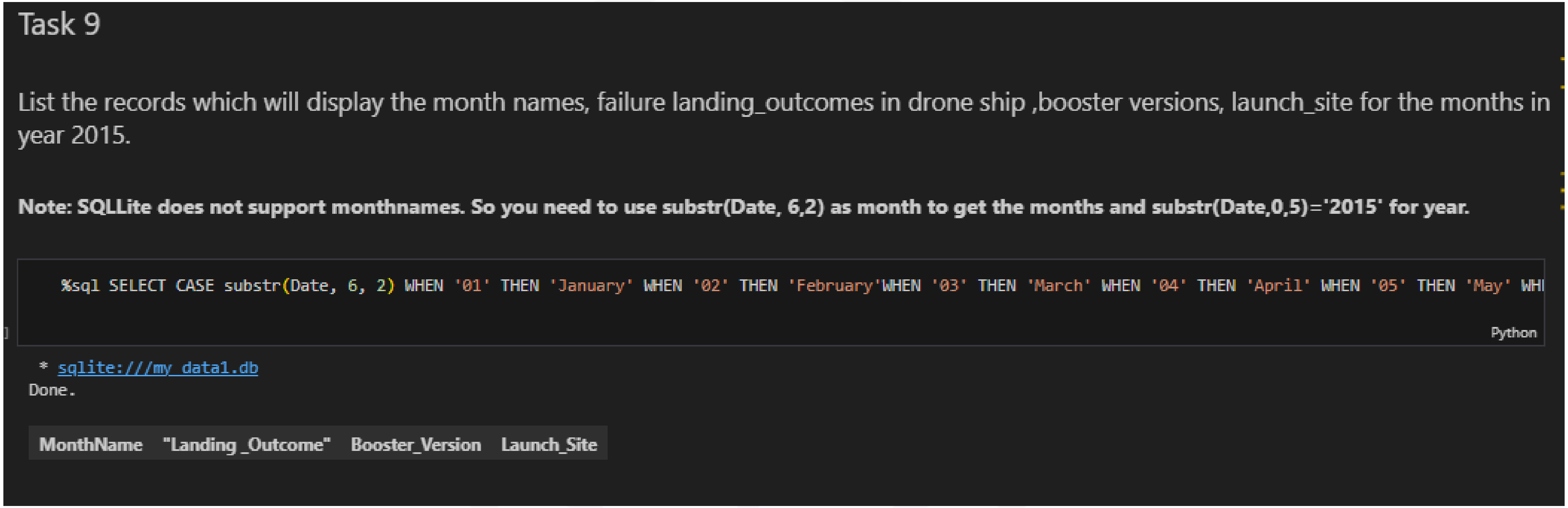
List all the booster_versions that have carried the maximum payload mass, using a subquery with a suitable aggregate function.

```
%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

RESULTS – Sample SQL



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.

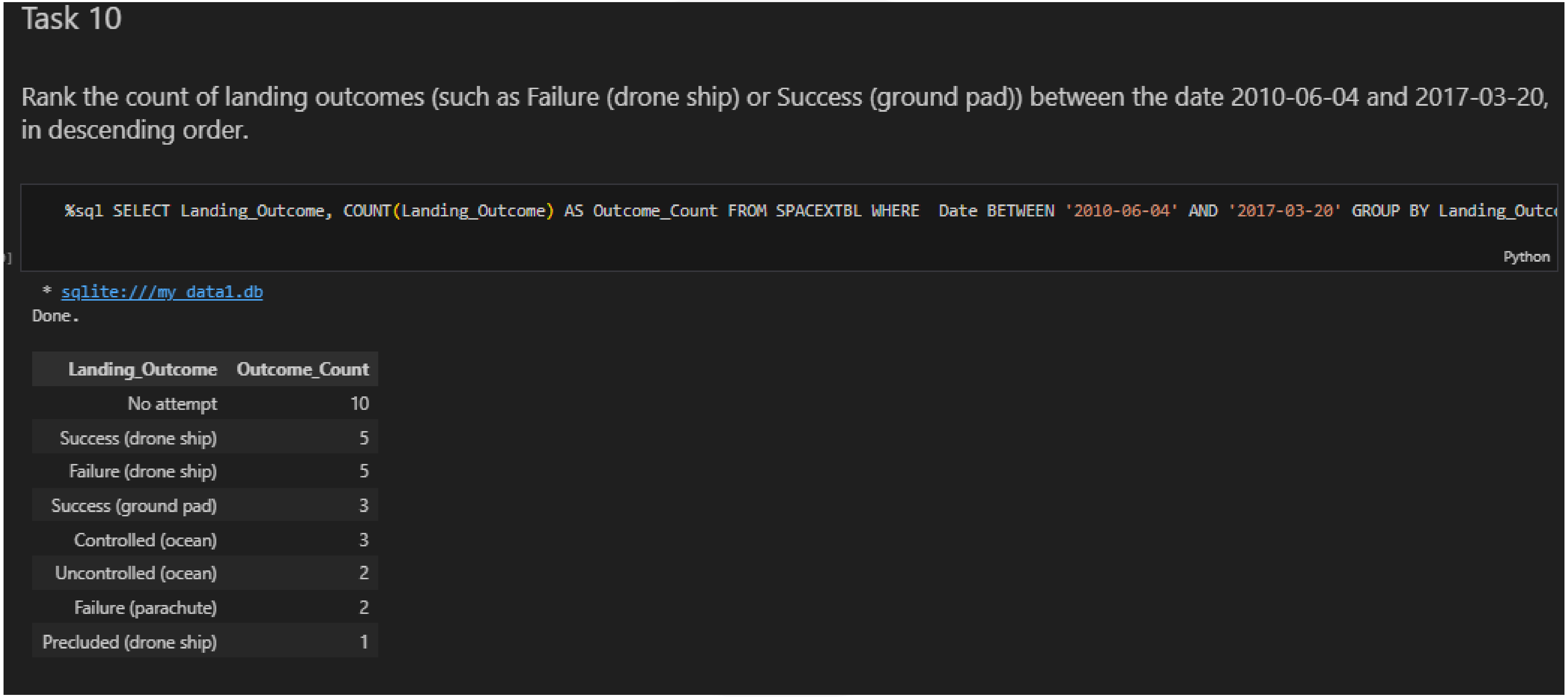
Note: SQLLite does not support monthnames. So you need to use substr(Date, 6,2) as month to get the months and substr(Date,0,5)='2015' for year.

```
%sql SELECT CASE substr(Date, 6, 2) WHEN '01' THEN 'January' WHEN '02' THEN 'February'WHEN '03' THEN 'March' WHEN '04' THEN 'April' WHEN '05' THEN 'May' WH
```

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

MonthName	"Landing _Outcome"	Booster_Version	Launch_Site

RESULTS – Sample SQL



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.

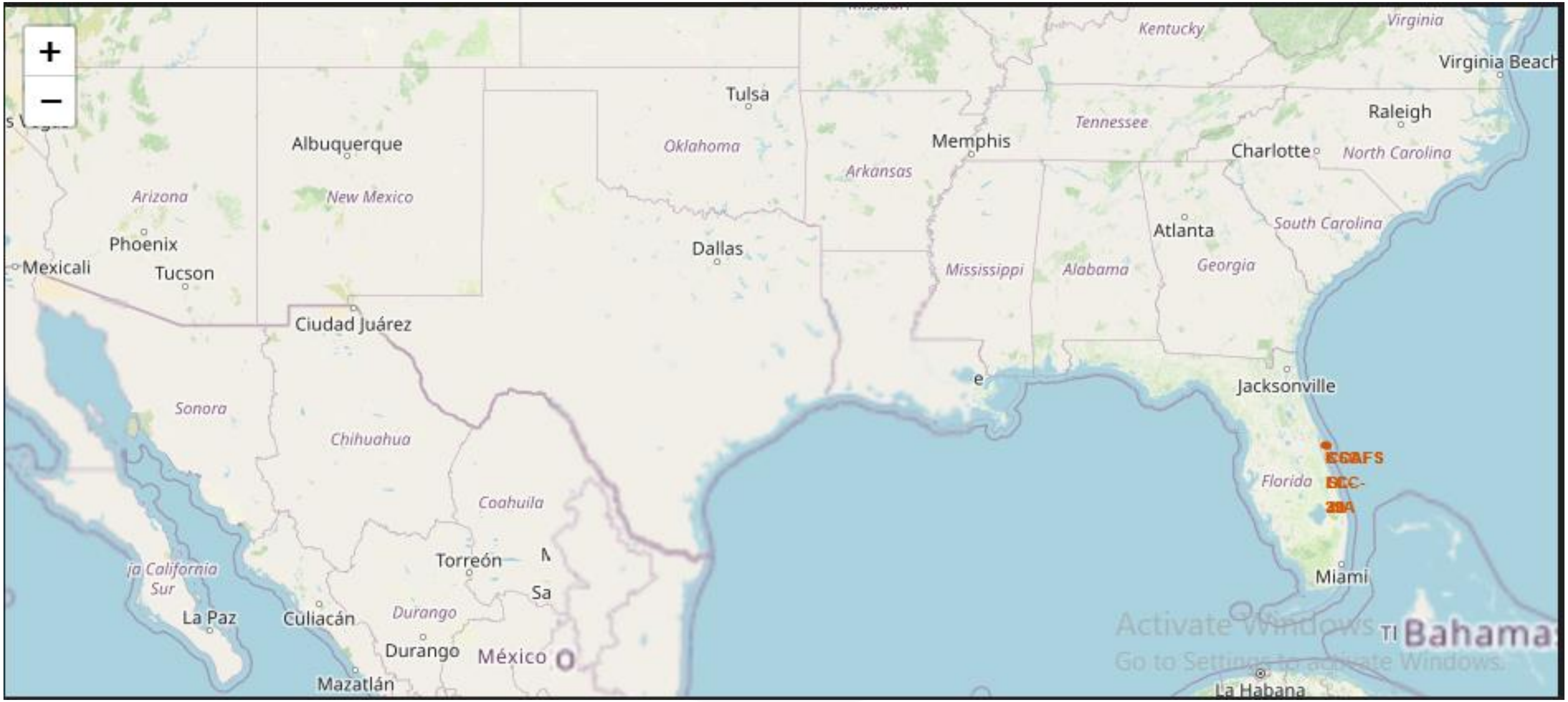
```
%sql SELECT Landing_Outcome, COUNT(Landing_Outcome) AS Outcome_Count FROM SPACEXTBL WHERE  Date BETWEEN '2010-06-04' AND '2017-03-20' GROUP BY Landing_Outc
```

Python

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

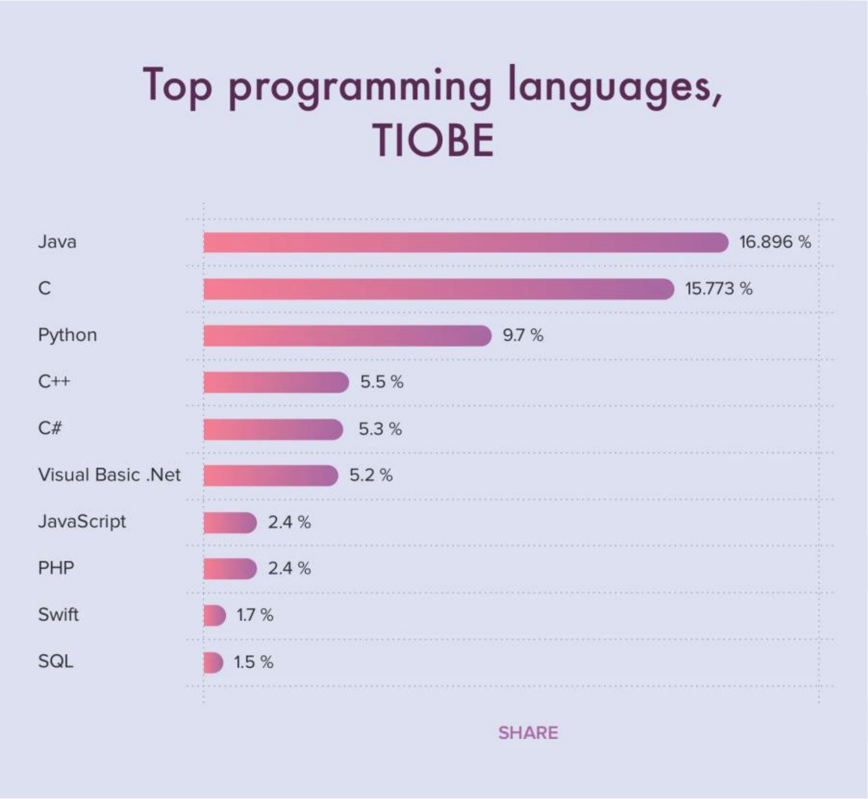
Landing_Outcome	Outcome_Count
No attempt	10
Success (drone ship)	5
Failure (drone ship)	5
Success (ground pad)	3
Controlled (ocean)	3
Uncontrolled (ocean)	2
Failure (parachute)	2
Precluded (drone ship)	1

RESULTS – Sample Folium

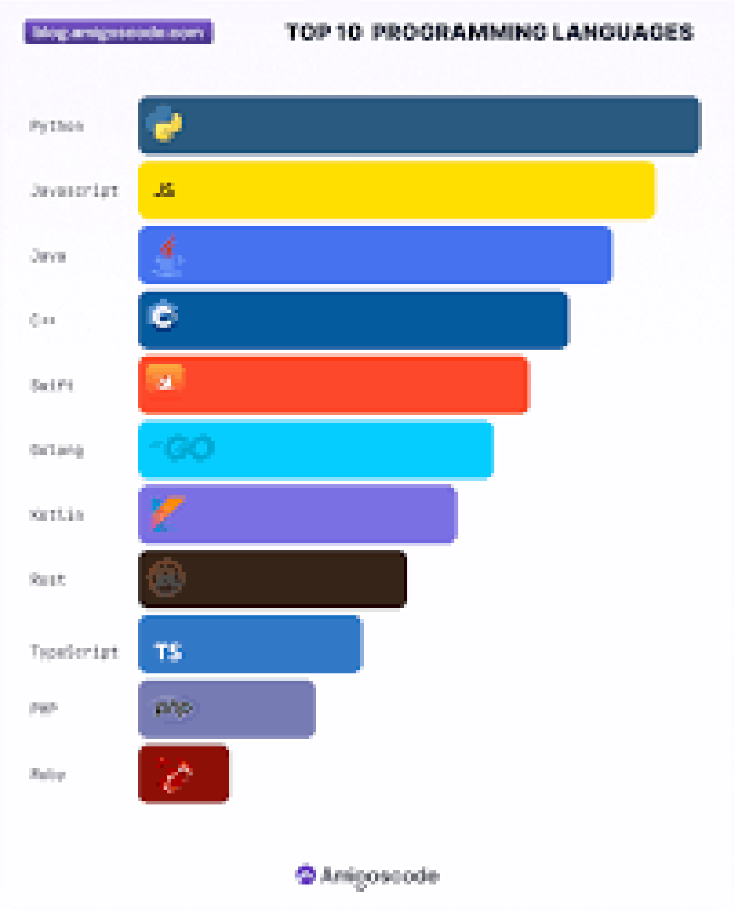


PROGRAMMING LANGUAGE TRENDS

Current Year



Next Year



PROGRAMMING LANGUAGE TRENDS - FINDINGS & IMPLICATIONS

Findings

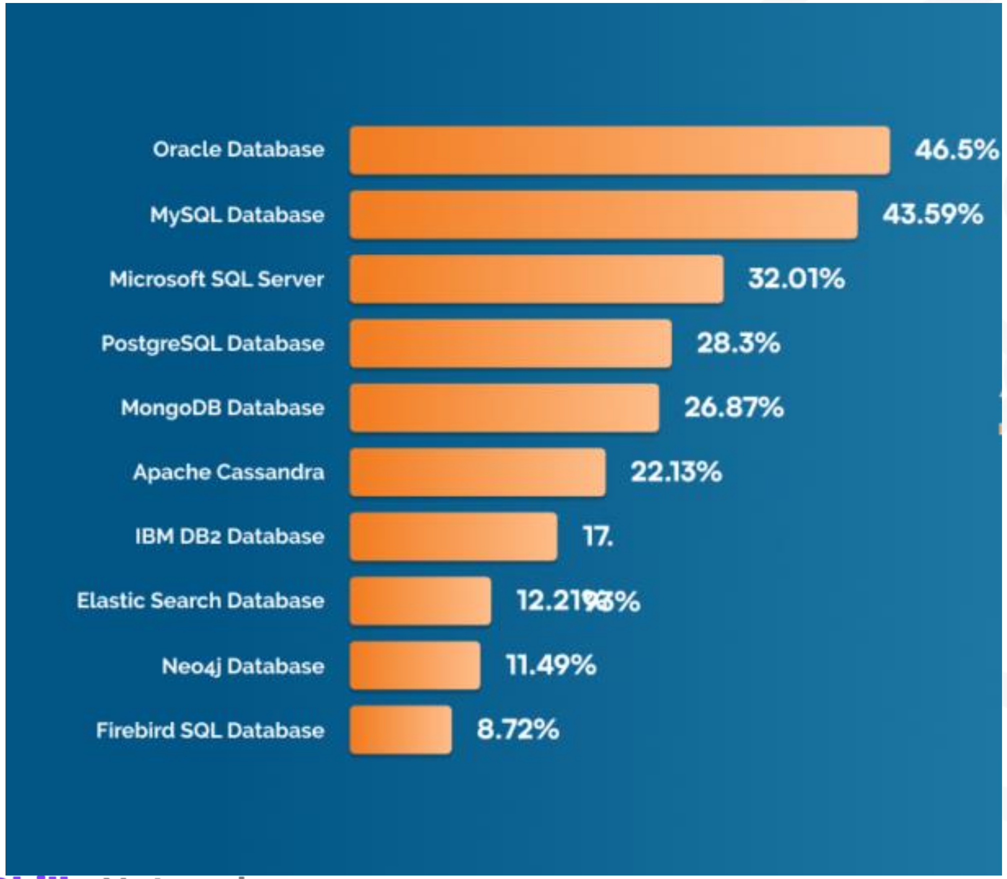
- Python will be the top
- PHP, Swift etc now not favoured
- JavaScript still favoured

Implications

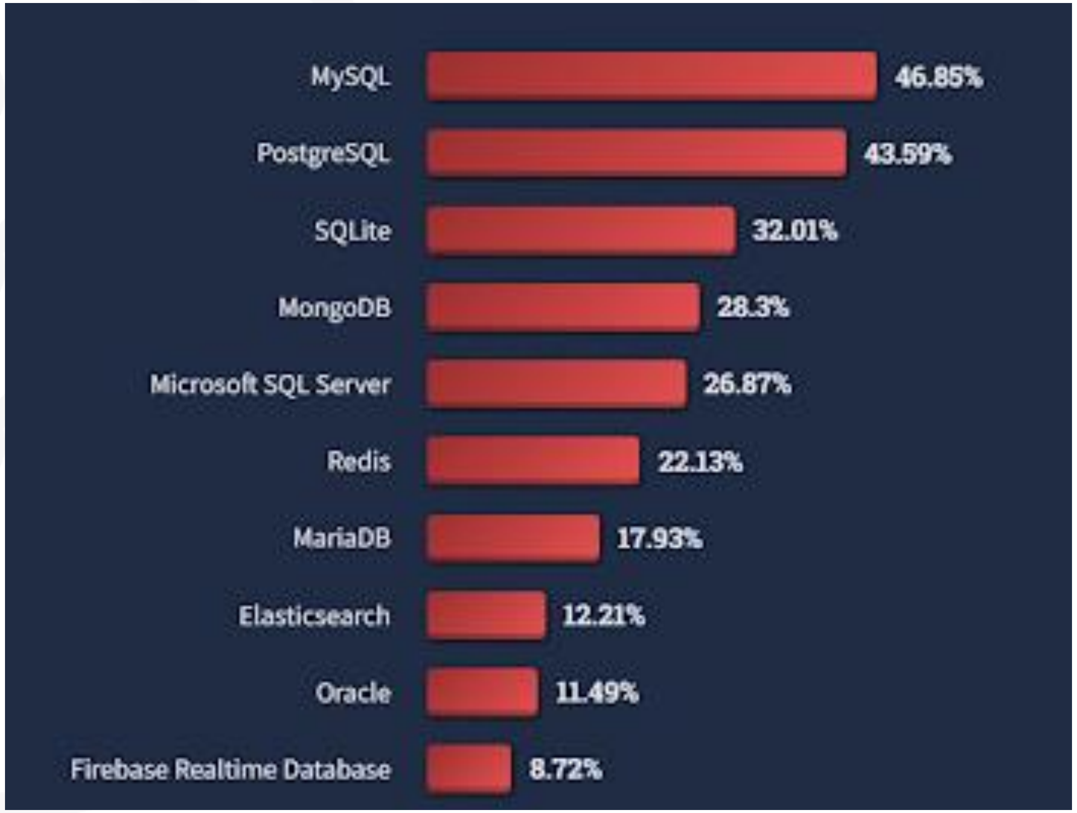
- Programmers will have to learn Python
- Programmers use alternative modern languages
- Web pages still need JS

DATABASE TRENDS

Current Year



Next Year



DATABASE TRENDS - FINDINGS & IMPLICATIONS

Findings

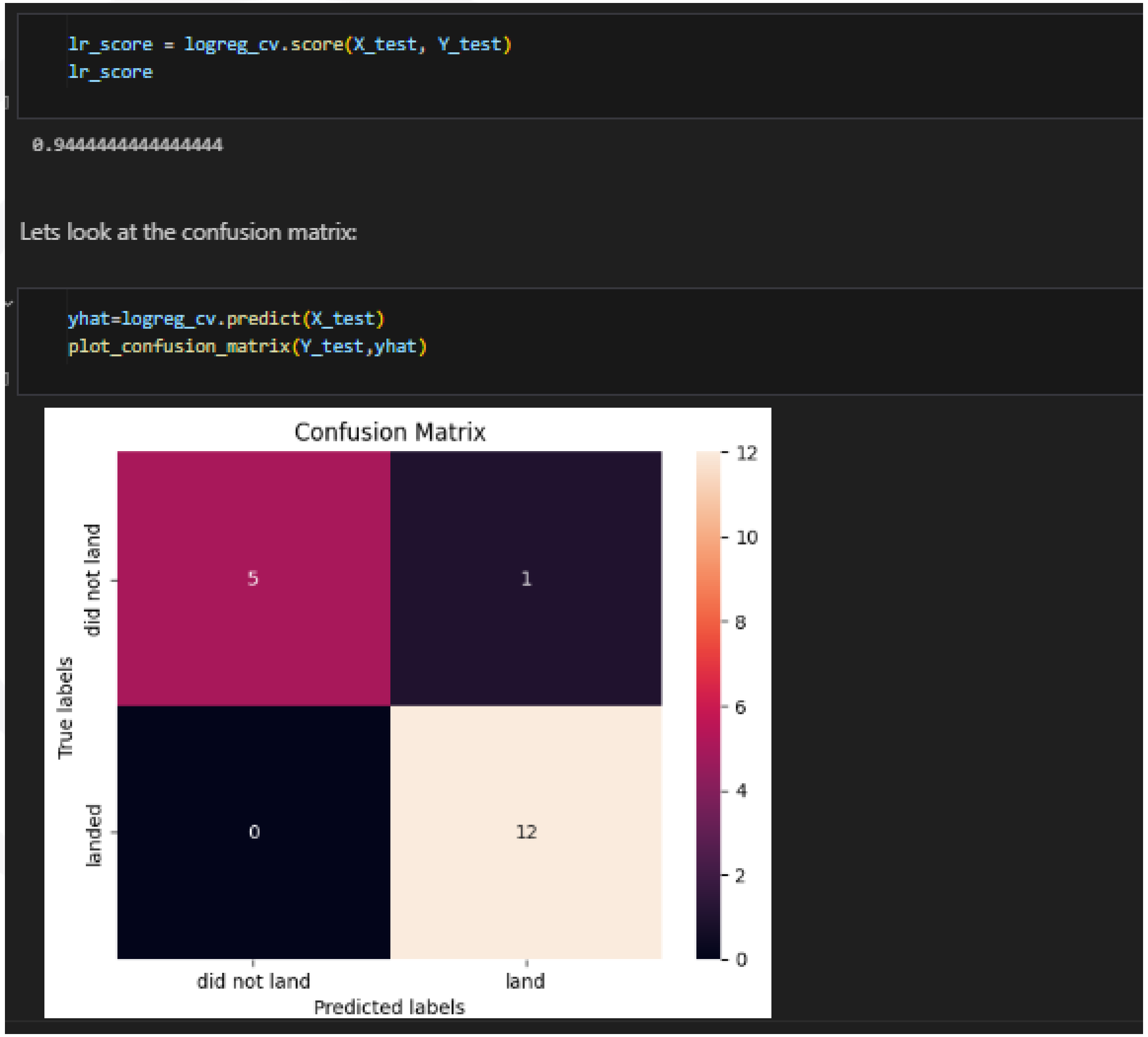
- MySQL will be the top
- NoSQL databases are increasing
- Less usage of Oracle

Implications

- MySQL is easy and open source so many programmers will use it
- More applications can be developed to store unstructured content
- Promotion of MySQL since it is open source

PREDICTION ANALYTICS

- Confusion matrix



```
lr_score = logreg_cv.score(X_test, Y_test)
lr_score
```

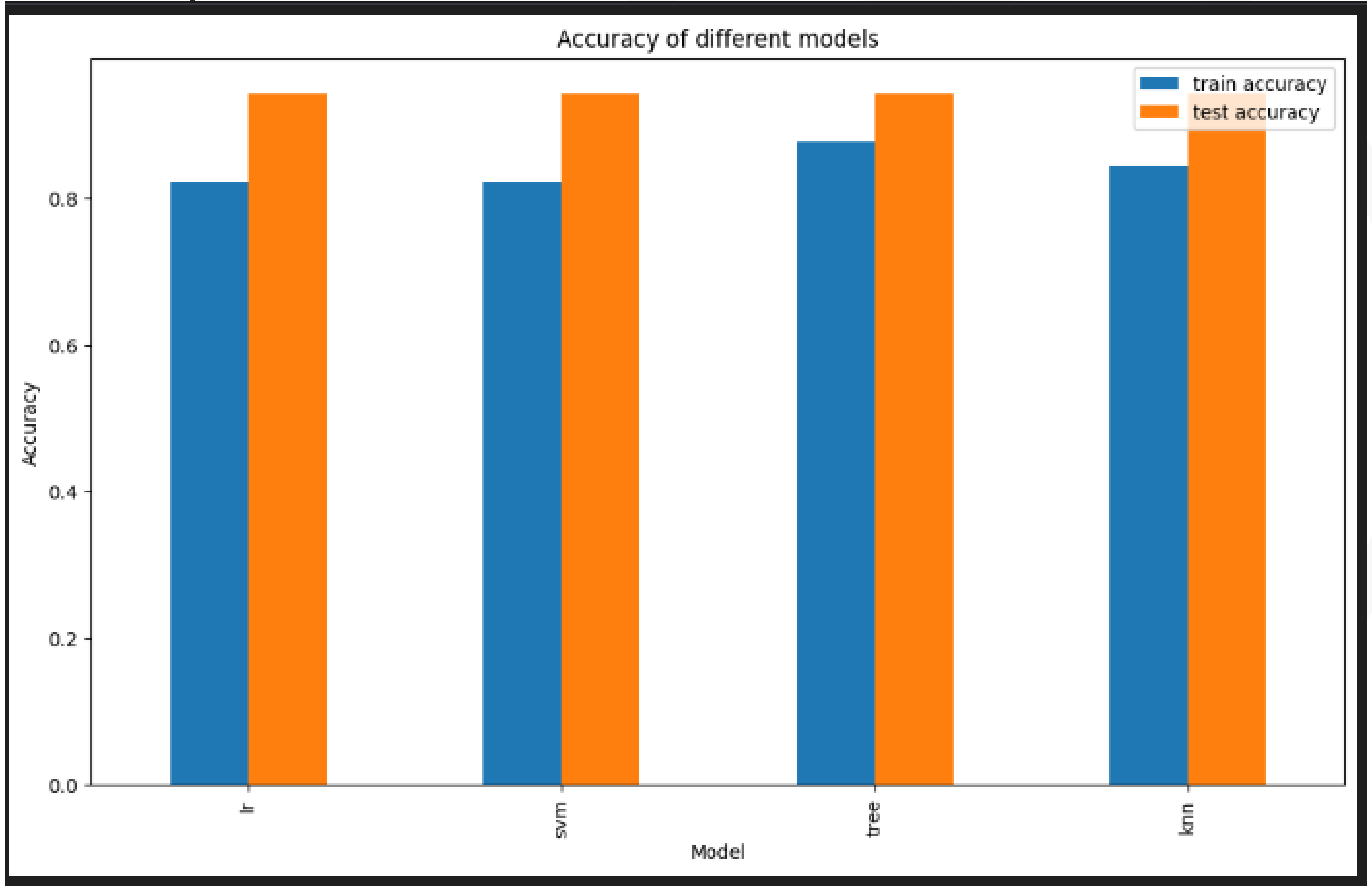
```
0.9444444444444444
```

Lets look at the confusion matrix:

```
yhat=logreg_cv.predict(X_test)
plot_confusion_matrix(Y_test,yhat)
```

PREDICTION ANALYTICS

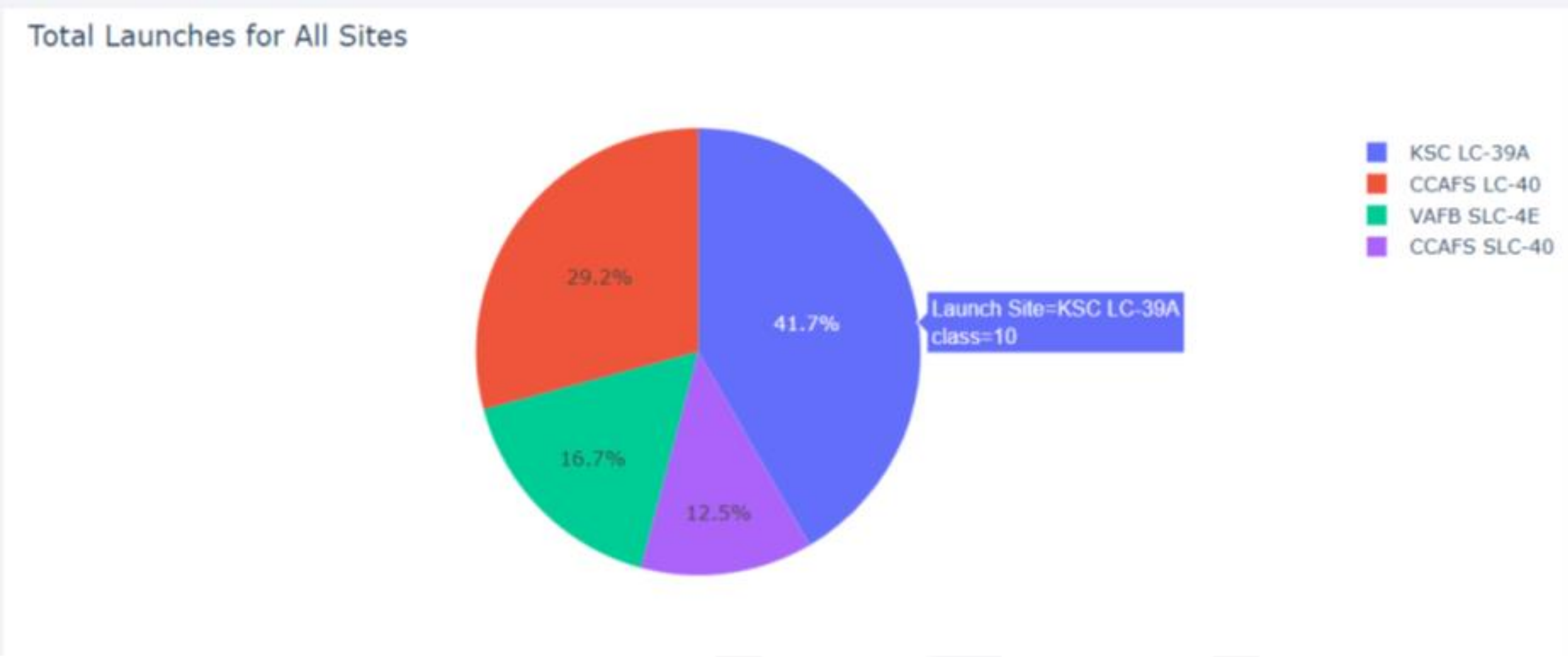
- Accuracy of different models



DASHBOARD

https://
https://github.com/ttmunoz/Applied_Data_Science_Capstone/

DASHBOARD TAB 1



DASHBOARD TAB 2

Screenshot of dashboard tab 2 goes here

DASHBOARD TAB 3

Screenshot of dashboard tab 3 goes here

DISCUSSION

- This data analysis will help future companies and competitors to find suitable launch sites and parameters to prevent failures

OVERALL FINDINGS & IMPLICATIONS

Findings

- Best payloads not 0-6000 kg Launch Sites are close to sea
- ML work good

Implications

- Payloads of 0-6000 kg are quite risky
- It is not over populated
- More fine tuning needed on ML algorithms for best results

CONCLUSION

- SpaceX data was collected from open sources
- Best launch site among four launch sites is KSC LC-39A
- Payloads of 0-6000 kg are quite risky
- Launch Sites are close to sea, not over populated
- • All ML algorithms work as expected

APPENDIX

- Github link
- https://github.com/ttmunoz/Applied_Data_Science_Capstone

JOB POSTINGS

In Module 1 you have collected the job posting data using Job API in a file named "job-postings.xlsx". Present that data using a bar chart here. Order the bar chart in the descending order of the number of job postings.

POPULAR LANGUAGES

In Module 1 you have collected the job postings data using web scraping in a file named “popular-languages.csv”. Present that data using a bar chart here. Order the bar chart in the descending order of salary.