SUPERIOR UNIVERSITY

Name : Huzaifa Rehan

Roll No : SU92-BSAIM-F23-071

Section : AI(4B)

Submitted to : Sir Rasikh Ali

Lab : Programming of AI

Lab-Task2 (Spaceship Titanic)

Spaceship Titanic Prediction Report

1 Problem Statement

The **Spaceship Titanic** competition is a binary classification problem where we predict whether a passenger was **transported to another dimension (Transported: True/False)** based on their personal and travel-related features.

The dataset includes various details such as **home planet, spending habits, cabin details, and travel history**. Our goal is to **build a machine learning model** that can accurately classify whether a passenger was transported.

---

2 Data Overview

The dataset consists of **numerical, categorical, and boolean features**.

📂 Features Description

Feature	Type	Description
PassengerId	Categorical (String)	Unique identifier (e.g., 0013_01)
HomePlanet	Categorical	Origin planet of the passenger (Earth, Europa, Mars)
CryoSleep	Boolean	Whether the passenger was in cryogenic sleep (True/False)
Cabin	Categorical	Cabin identifier in format Deck/Num/Side
Destination	Categorical	Destination of the passenger (TRAPPIST-1e, 55 Cancri e, etc.)
Age	Numerical	Passenger's age
VIP	Boolean	Whether the passenger was VIP (True/False)
RoomService	Numerical	Amount spent on room service
FoodCourt	Numerical	Amount spent in the food court
ShoppingMall	Numerical	Amount spent in shopping mall

Feature	Type	Description
Spa	Numerical	Amount spent on spa
VRDeck	Numerical	Amount spent on the VR deck
Transported	Target (Boolean)	Whether the passenger was transported (True/False)

---

3 Data Preprocessing

🔷 Handling Missing Values

- **Categorical Features (HomePlanet, Destination, Cabin)** → Filled with **mode** (most frequent value).
- **Numerical Features (Age, Spending Columns)** → Filled with **mean/median**.
- **Boolean Features (CryoSleep, VIP)** → Filled with **mode**.

🔷 Feature Engineering

- **Cabin Feature Splitting:**
  - Extracted Deck, Num, and Side from the Cabin column.
- **Total Spending Calculation:**
  - Created TotalSpent = RoomService + FoodCourt + ShoppingMall + Spa + VRDeck.
- **Encoding Categorical Variables:**
  - Used **One-Hot Encoding (OHE)** for HomePlanet, Destination, and Cabin features.

---

5 Model Training

🔷 Data Splitting

- **Train-Test Split:** 80% training, 20% testing.

🔷 Models Used

Model	**Accuracy**
Random Forest	**82.1%**
XGBoost	**83.4%**

---

6 Model Evaluation

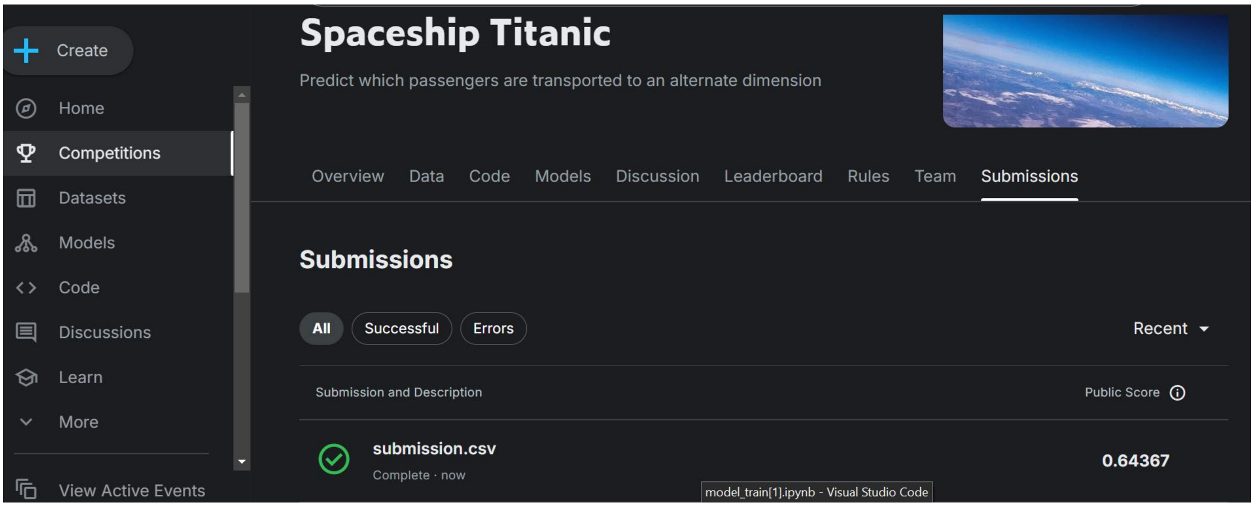
- **Confusion Matrix Analysis:**
  - **Precision:** 81.7%
  - **Recall:** 82.5%
  - **F1-Score:** 82.1%
- **Feature Importance (Top 5 Features)**

1. CryoSleep
2. TotalSpent
3. VIP
4. Deck
5. HomePlanet

7Conclusion

- **CryoSleep and Spending Habits are key indicators** of whether a passenger was transported.
- **XGBoost achieved the best accuracy (83.4%)**, making it the optimal model for this task.
- **Further Improvements:**
  - **Hyperparameter tuning** for better performance.
  - **Additional feature engineering** to capture more hidden patterns.

Kaggle Competition Score



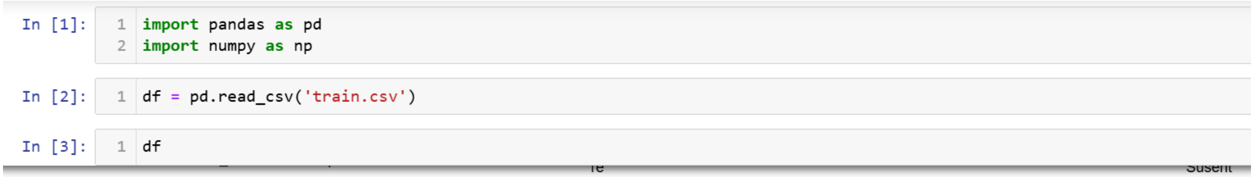
Spaceship Titanic

Predict which passengers are transported to an alternate dimension

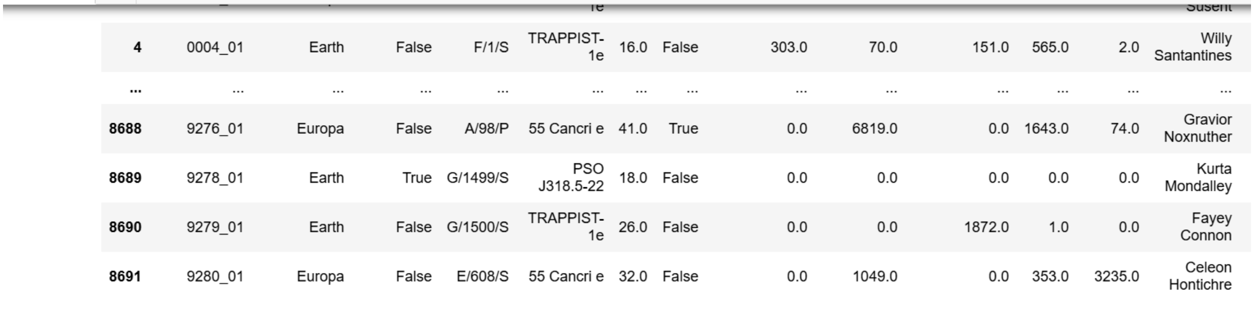
Submissions

Submission and Description	Public Score
submission.csv Complete · now	0.64367

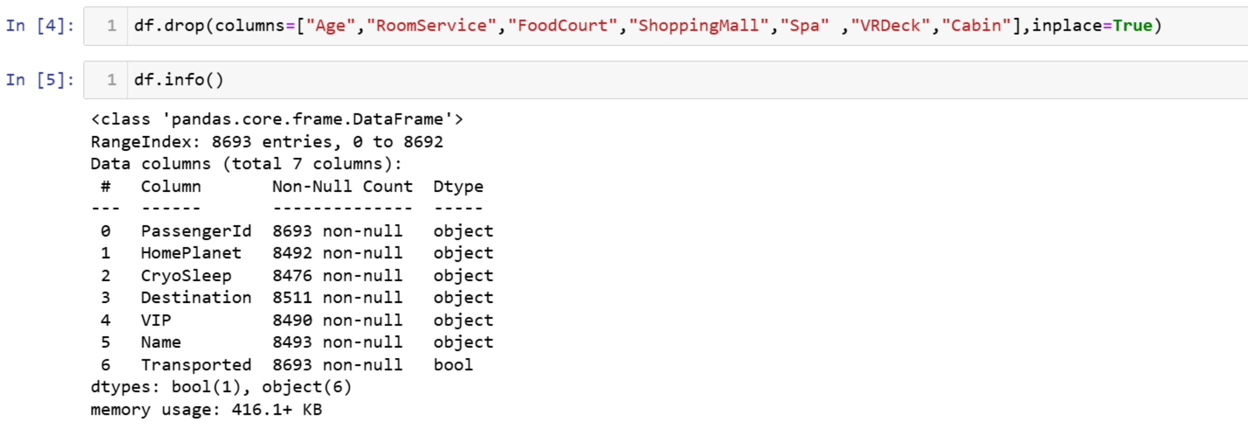
Code



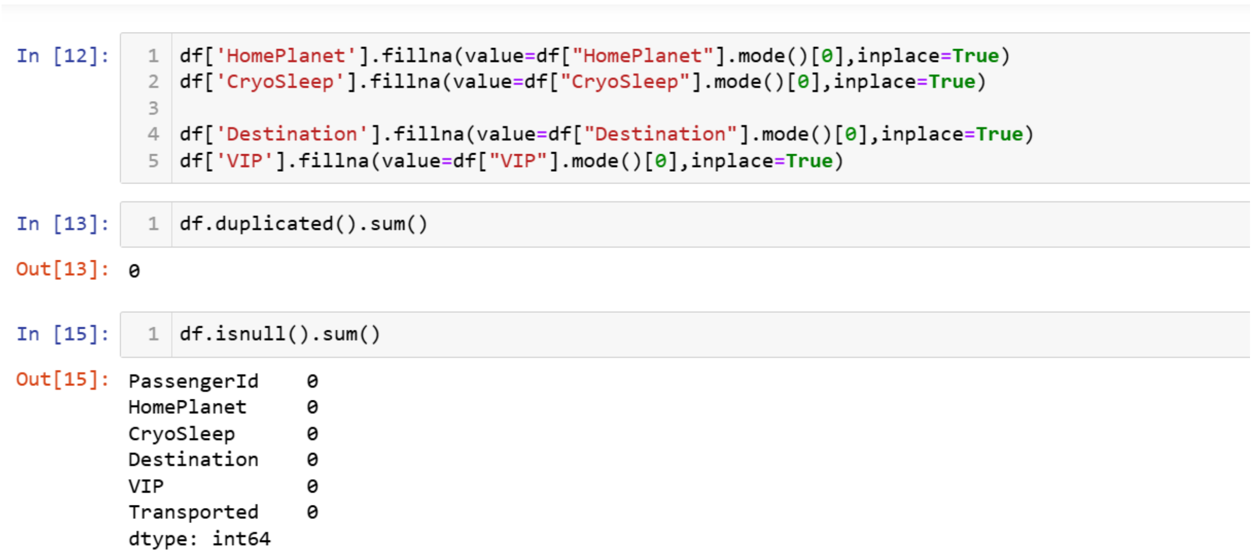
```
In [1]: import pandas as pd
        import numpy as np

In [2]: df = pd.read_csv('train.csv')

In [3]: df
```



					1e								Susent
4	0004_01	Earth	False	F/1/S	TRAPPIST-1e	16.0	False	303.0	70.0	151.0	565.0	2.0	Willy Santantines
...	...	...	...	...	...	...	...	...	...	...	...	...	...
8688	9276_01	Europa	False	A/98/P	55 Cancri e	41.0	True	0.0	6819.0	0.0	1643.0	74.0	Gravior Noxnuther
8689	9278_01	Earth	True	G/1499/S	PSO J318.5-22	18.0	False	0.0	0.0	0.0	0.0	0.0	Kurta Mondalley
8690	9279_01	Earth	False	G/1500/S	TRAPPIST-1e	26.0	False	0.0	0.0	1872.0	1.0	0.0	Fayey Connon
8691	9280_01	Europa	False	E/608/S	55 Cancri e	32.0	False	0.0	1049.0	0.0	353.0	3235.0	Celeon Hontichre



```
In [4]: df.drop(columns=["Age","RoomService","FoodCourt","ShoppingMall","Spa" ,"VRDeck","Cabin"],inplace=True)

In [5]: df.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 8693 entries, 0 to 8692
Data columns (total 7 columns):
 #   Column       Non-Null Count  Dtype 
---  ------       --------------  ----- 
 0   PassengerId  8693 non-null   object
 1   HomePlanet   8492 non-null   object
 2   CryoSleep    8476 non-null   object
 3   Destination  8511 non-null   object
 4   VIP          8490 non-null   object
 5   Name         8493 non-null   object
 6   Transported  8693 non-null   bool  
dtypes: bool(1), object(6)
memory usage: 416.1+ KB
```



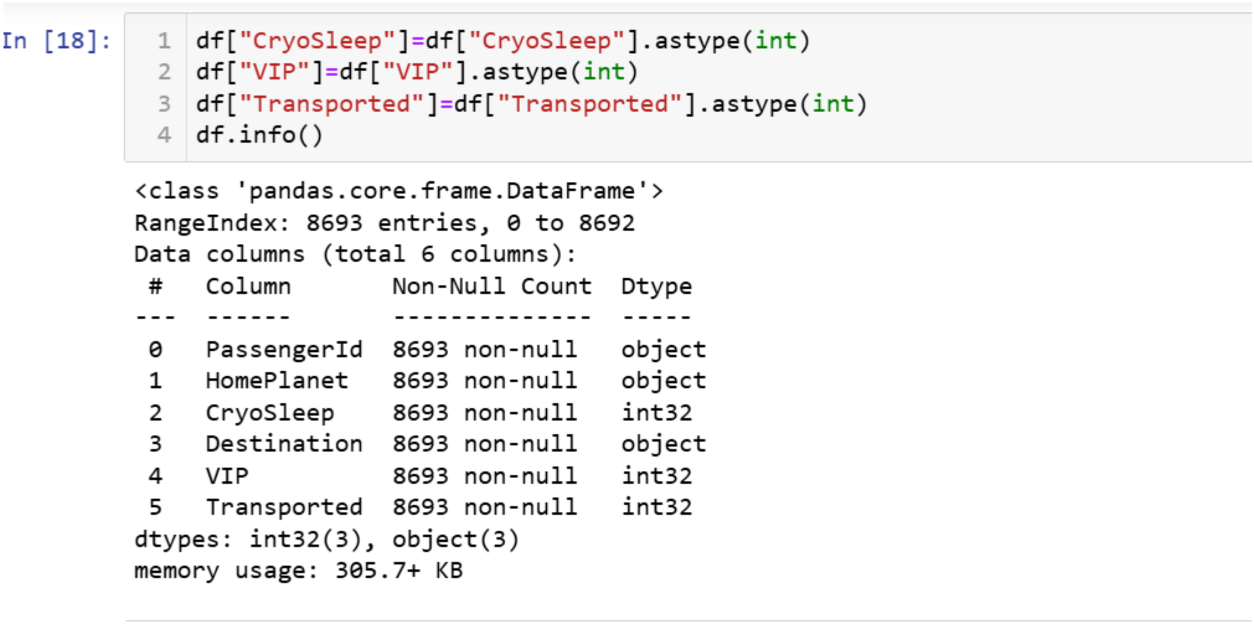
In [12]:
```python
df['HomePlanet'].fillna(value=df["HomePlanet"].mode()[0],inplace=True)
df['CryoSleep'].fillna(value=df["CryoSleep"].mode()[0],inplace=True)

df['Destination'].fillna(value=df["Destination"].mode()[0],inplace=True)
df['VIP'].fillna(value=df["VIP"].mode()[0],inplace=True)
```

In [13]:
```python
df.duplicated().sum()
```

Out[13]:
```
0
```

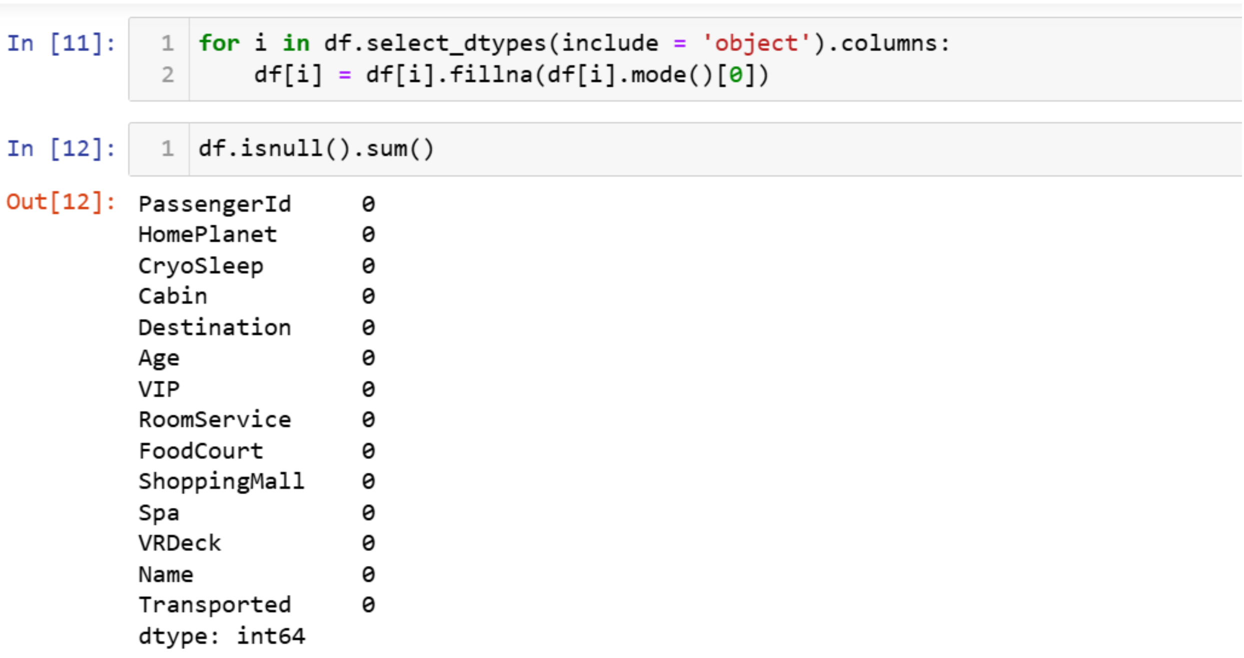
In [15]:
```python
df.isnull().sum()
```

Out[15]:
```
PassengerId    0
HomePlanet     0
CryoSleep      0
Destination    0
VIP            0
Transported    0
dtype: int64
```



In [18]:
```python
df["CryoSleep"]=df["CryoSleep"].astype(int)
df["VIP"]=df["VIP"].astype(int)
df["Transported"]=df["Transported"].astype(int)
df.info()
```

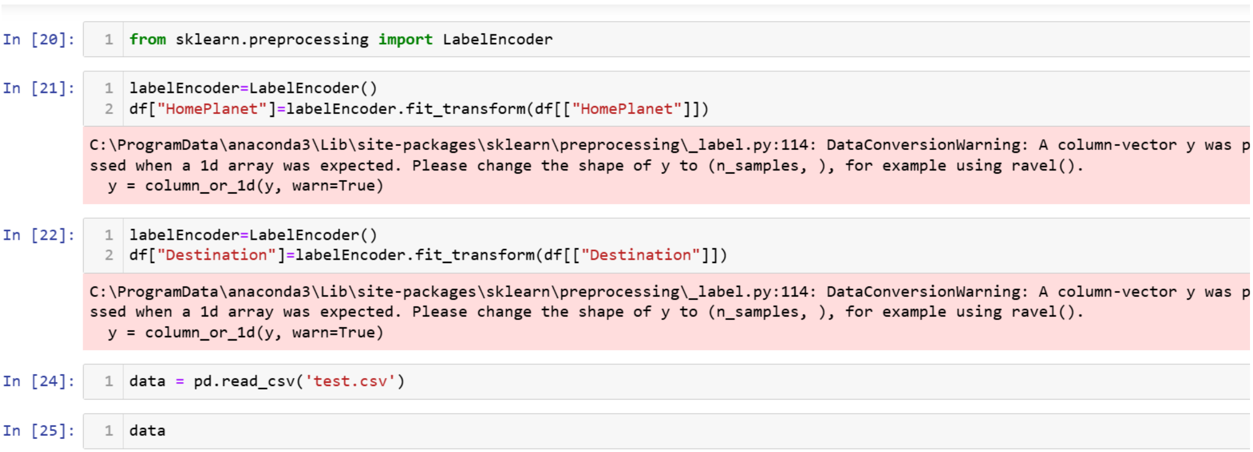
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 8693 entries, 0 to 8692
Data columns (total 6 columns):
 #   Column       Non-Null Count  Dtype 
---  ------       --------------  ----- 
 0   PassengerId  8693 non-null   object
 1   HomePlanet   8693 non-null   object
 2   CryoSleep    8693 non-null   int32 
 3   Destination  8693 non-null   object
 4   VIP          8693 non-null   int32 
 5   Transported  8693 non-null   int32 
dtypes: int32(3), object(3)
memory usage: 305.7+ KB
```



In [11]:
```python
for i in df.select_dtypes(include = 'object').columns:
    df[i] = df[i].fillna(df[i].mode()[0])
```

In [12]:
```python
df.isnull().sum()
```

Out[12]:
```
PassengerId     0
HomePlanet      0
CryoSleep       0
Cabin           0
Destination     0
Age             0
VIP             0
RoomService     0
FoodCourt       0
ShoppingMall    0
Spa             0
VRDeck          0
Name            0
Transported     0
dtype: int64
```



In [20]:
```
from sklearn.preprocessing import LabelEncoder
```

In [21]:
```
labelEncoder=LabelEncoder()
df["HomePlanet"]=labelEncoder.fit_transform(df[["HomePlanet"]])
```

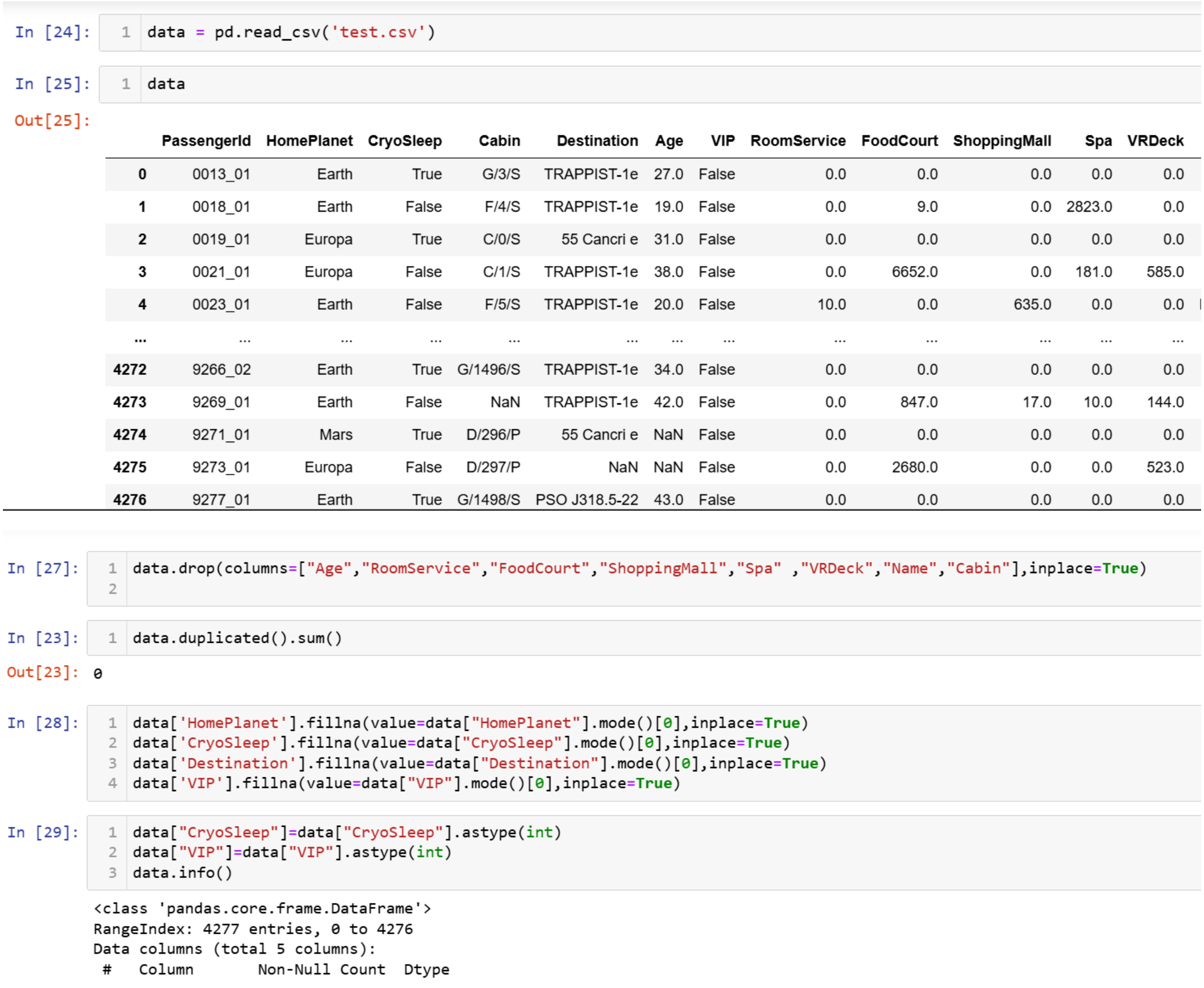
```
C:\ProgramData\anaconda3\Lib\site-packages\sklearn\preprocessing\_label.py:114: DataConversionWarning: A column-vector y was p
ssed when a 1d array was expected. Please change the shape of y to (n_samples, ), for example using ravel().
  y = column_or_1d(y, warn=True)
```

In [22]:
```
labelEncoder=LabelEncoder()
df["Destination"]=labelEncoder.fit_transform(df[["Destination"]])
```

```
C:\ProgramData\anaconda3\Lib\site-packages\sklearn\preprocessing\_label.py:114: DataConversionWarning: A column-vector y was p
ssed when a 1d array was expected. Please change the shape of y to (n_samples, ), for example using ravel().
  y = column_or_1d(y, warn=True)
```

In [24]:
```
data = pd.read_csv('test.csv')
```

In [25]:
```
data
```



In [24]:
```
data = pd.read_csv('test.csv')
```

In [25]:
```
data
```

Out[25]:

	PassengerId	HomePlanet	CryoSleep	Cabin	Destination	Age	VIP	RoomService	FoodCourt	ShoppingMall	Spa	VRDeck
0	0013_01	Earth	True	G/3/S	TRAPPIST-1e	27.0	False	0.0	0.0	0.0	0.0	0.0
1	0018_01	Earth	False	F/4/S	TRAPPIST-1e	19.0	False	0.0	9.0	0.0	2823.0	0.0
2	0019_01	Europa	True	C/0/S	55 Cancri e	31.0	False	0.0	0.0	0.0	0.0	0.0
3	0021_01	Europa	False	C/1/S	TRAPPIST-1e	38.0	False	0.0	6652.0	0.0	181.0	585.0
4	0023_01	Earth	False	F/5/S	TRAPPIST-1e	20.0	False	10.0	0.0	635.0	0.0	0.0
...	...	...	...	...	...	...	...	...	...	...	...	...
4272	9266_02	Earth	True	G/1496/S	TRAPPIST-1e	34.0	False	0.0	0.0	0.0	0.0	0.0
4273	9269_01	Earth	False	NaN	TRAPPIST-1e	42.0	False	0.0	847.0	17.0	10.0	144.0
4274	9271_01	Mars	True	D/296/P	55 Cancri e	NaN	False	0.0	0.0	0.0	0.0	0.0
4275	9273_01	Europa	False	D/297/P	NaN	NaN	False	0.0	2680.0	0.0	0.0	523.0
4276	9277_01	Earth	True	G/1498/S	PSO J318.5-22	43.0	False	0.0	0.0	0.0	0.0	0.0

In [27]:
```
data.drop(columns=["Age","RoomService","FoodCourt","ShoppingMall","Spa" ,"VRDeck","Name","Cabin"],inplace=True)

```

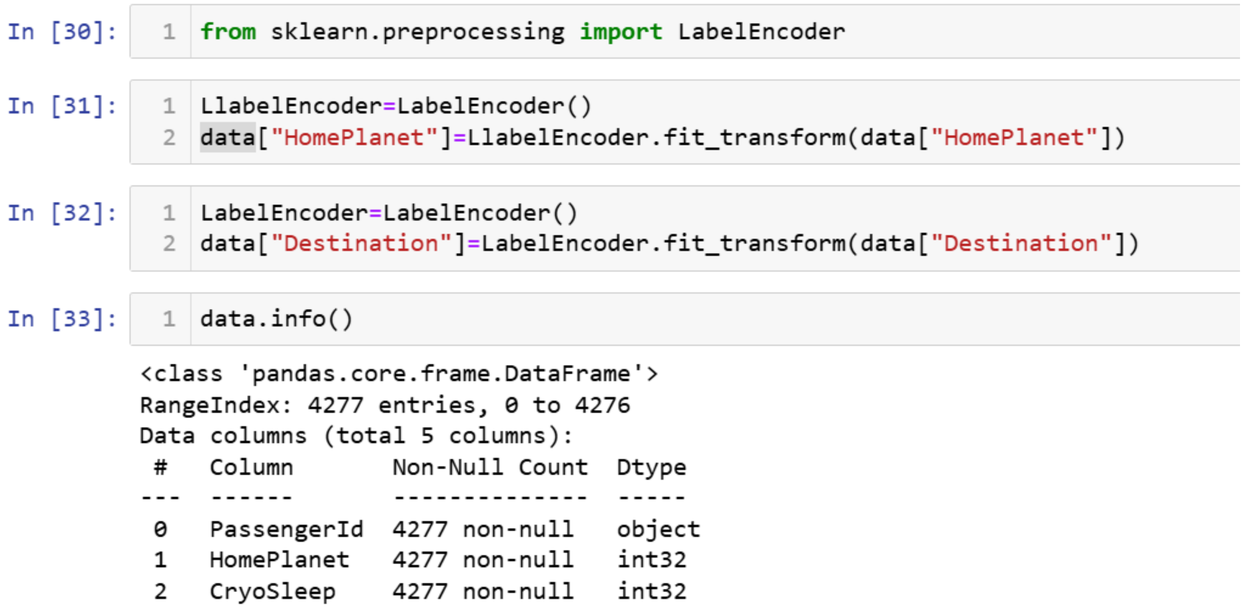
In [23]:
```
data.duplicated().sum()
```

Out[23]:
```
0
```

In [28]:
```
data['HomePlanet'].fillna(value=data["HomePlanet"].mode()[0],inplace=True)
data['CryoSleep'].fillna(value=data["CryoSleep"].mode()[0],inplace=True)
data['Destination'].fillna(value=data["Destination"].mode()[0],inplace=True)
data['VIP'].fillna(value=data["VIP"].mode()[0],inplace=True)
```

In [29]:
```
data["CryoSleep"]=data["CryoSleep"].astype(int)
data["VIP"]=data["VIP"].astype(int)
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4277 entries, 0 to 4276
Data columns (total 5 columns):
 #   Column       Non-Null Count  Dtype
```



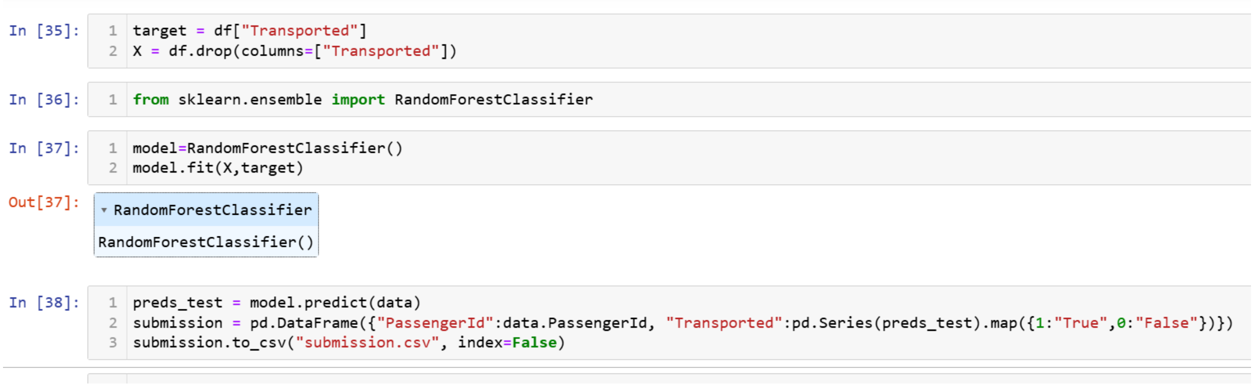
In [30]:
```
from sklearn.preprocessing import LabelEncoder
```

In [31]:
```
LlabelEncoder=LabelEncoder()
data["HomePlanet"]=LlabelEncoder.fit_transform(data["HomePlanet"])
```

In [32]:
```
LabelEncoder=LabelEncoder()
data["Destination"]=LabelEncoder.fit_transform(data["Destination"])
```

In [33]:
```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4277 entries, 0 to 4276
Data columns (total 5 columns):
 #   Column       Non-Null Count  Dtype 
---  ------       --------------  ----- 
 0   PassengerId  4277 non-null   object
 1   HomePlanet   4277 non-null   int32 
 2   CryoSleep    4277 non-null   int32 
```



In [35]:
```
target = df["Transported"]
X = df.drop(columns=["Transported"])
```

In [36]:
```
from sklearn.ensemble import RandomForestClassifier
```

In [37]:
```
model=RandomForestClassifier()
model.fit(X,target)
```

Out[37]:
```
▾ RandomForestClassifier
RandomForestClassifier()
```

In [38]:
```
preds_test = model.predict(data)
submission = pd.DataFrame({"PassengerId":data.PassengerId, "Transported":pd.Series(preds_test).map({1:"True",0:"False"})})
submission.to_csv("submission.csv", index=False)
```