1. **PROJECT NO:**
   - 01

2. **TITLE:**
   - E-Commerce-Purchases-History Analysis using Padas

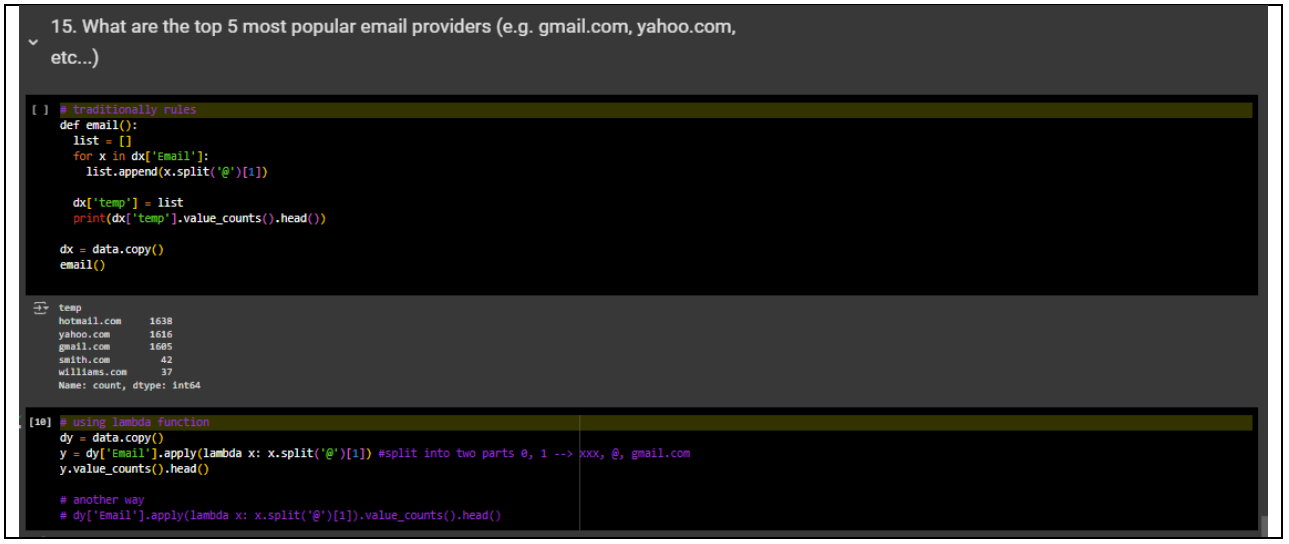
3. **OBJECTIVES:**
   - To learn about pandas and numpy
   - To take csv file in a dataframe
   - To Learn head, tail, info, describe of data
   - Perform 15 types of action on dataset
   - Perform on column access
   - Perform apply method
   - Know about how to interfere dataframe with Boolean dataframe
   - How to find a substring from columns?

4. **DESIGN/ALGORITHM/FLOWCHART:**

5. **QUESTION:**

   - Display top 10 rows of the dataset
   - Check last 10 rows of the dataset
   - Check datatype of each column
   - Check null values in the dataset
   - How many rows and columns are there in our dataset?
   - Highest and lowest purchase prices.
   - Average purchase price
   - How many people have French 'fr' as their language?
   - Job title contains engineer
   - Find the email of the person with the following ip address: 132.207.160.22
   - How many people have mastercard as their credit card provider and made a purchase above 5 0?
   - Find the email of the person with the following credit card number: 4664825258997302
   - How many people purchase during the am and how many people purchase during pm?
   - How many people have a credit card that expires in 2020?
   - What are the top 5 most popular email providers (e.g. gmail.com, yahoo.com, etc...)

6. IMPLEMENTATION:



15. What are the top 5 most popular email providers (e.g. gmail.com, yahoo.com, etc...)

```
[ ] # traditionally rules
    def email():
      list = []
      for x in dx['Email']:
        list.append(x.split('@')[1])

      dx['temp'] = list
      print(dx['temp'].value_counts().head())

    dx = data.copy()
    email()
```

```
temp
hotmail.com     1638
yahoo.com       1616
gmail.com       1605
smith.com         42
williams.com      37
Name: count, dtype: int64
```

```
[10] # using lambda function
     dy = data.copy()
     y = dy['Email'].apply(lambda x: x.split('@')[1]) #split into two parts 0, 1 --> xxx, @, gmail.com
     y.value_counts().head()

     # another way
     # dy['Email'].apply(lambda x: x.split('@')[1]).value_counts().head()
```

7. OUTPUT:

```
temp
hotmail.com     1638
yahoo.com       1616
gmail.com       1605
smith.com         42
williams.com      37
Name: count, dtype: int64
```

8. ANALYSIS:

9. DISCUSSION:

10. CONCLUSION:

11. REFERENCE:

- Dataset: Kaggle Dataset Link
- Tutorial: YouTube Tutorials