Customer retention case study

Steps followed while completing the project:

1.Loading

- First loading directories

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import warnings
warnings.filterwarnings('ignore')
```

- then loading data in excel files to data frame earlier showing Unicode error but after multiple attempt I can load this success fully code used:-
  df=pd.read_csv('C:\\Users\\hp\\Downloads\\customer_retention_dataset.xlsx - datasheet.csv')

- display of data:-code-
  df.head()

2. Removing tab spaces

3. Removing digits

4. Removing loading and trailling spaces

Code used for step 2,3,4:-

```python
from string import digits

df.columns = df.columns.str.replace('\t','')

remove_digits = str.maketrans('', '', digits)

df.columns = df.columns.str.translate(remove_digits)

df.columns = df.columns.str.strip()
```

5. Data Information

- df.shape
- df.dtypes

6. Finding null values-

```python
df.isnull().sum().any()
```

7. Finding unique columns

```python
df.nunique()
```

8. Preparation of visualization

- using similar items comparison using circle:-

  Gender
  Age
  City
  Time when start online purchase
  Online shop last year
- Code for above:-

  1.) personal_info=['Gender of respondent','How old are you?','Which city do you shop online from?',
        'What is the Pin Code of where you shop online from?','Since How Long You are Shopping Online ?',
        'How many times you have made an online purchase in the past year?']

  2.) for i in personal_info:

```python
        if i!='What is the Pin Code of where you
shop online from?':
            plt.figure(figsize=(8,6))

df[i].value_counts().plot.pie(autopct='%1.1f%%
')
            centre=plt.Circle((0,0),0.7,fc='white')
            fig=plt.gcf()
            fig.gca().add_artist(centre)
            plt.xlabel(i)
            plt.ylabel('')
            plt.figure()
```

9. Compairing last year use in reference to various online retailers:

Code-

1. df['How many times you have made an online purchase in the past

```
year?'].replace('42 times and above','41
times and above',
inplace=True)
```
2.
```
plt.figure(figsize=(15,8))
sns.lineplot(df['How many times you have
made an online purchase in the past
year?'],
        df['From the following, tick any
(or all) of the online retailers you have
shopped from;'])
```

10. Compairing last year no of time in reference to various online retailers:

Code-

1.
```
dict={'31-40 times':35,'41 times and
above':45,'Less than 10 times':5,'11-20
times':15,'21-30 times':25}
```

```
df['Average times made an online
purchase']=df['How many times you have made
```

an online purchase in the past year?'].replace(dict)

2.plt.figure(figsize=(20,8))

sns.violinplot(df['From the following, tick any (or all) of the online retailers you have shopped from;'],

      df['Average times made an online purchase'],hue=df['You feel gratification shopping on your favorite e-tailer'])

plt.xticks(rotation=45)

10. From online retailers average number of times shop Gaining access to loyalty programs:-

Code:-

plt.figure(figsize=(20,8))

```
sns.violinplot(df['From the following, tick any
(or all) of the online retailers you have shopped
from;'],
          df['Average times made an online
purchase'],hue=df['Gaining access to loyalty
programs is a benefit of shopping online'])
plt.xticks(rotation=45)
```

11. Compairing age and from how long shopping online :-

Code:-

```
plt.figure(figsize=(10,8))
sns.countplot(df['Since How Long You are
Shopping Online ?'],hue=df['How old are
you?'])
```

12. finding "Since How Long You are Shopping Online" various time categories and "Which city do you shop online from" unique categories:-

Code:-

df['Since How Long You are Shopping Online ?'].unique()

df['Which city do you shop online from?'].unique()

13. Creating dictionary for how long items and city you shop from:-

Code:-

1.dict={'Above 4 years':4.5,'3-4 years':3.5,'2-3 years':2.5,'1-2 years':1.5,'Less than 1 year':0.5}

df['Average years of shopping online']=df['Since How Long You are Shopping Online ?'].replace(dict)

2.df['Which city do you shop online from?'].replace({'Greater Noida':'Noida'},inplace=True)

14. Ploting the abobe finding along with gender that is time of using, cities and gender.:-

Code:-

plt.figure(figsize=(15,8))

sns.lineplot(df['Which city do you shop online from?'],df['Average years of shopping online'],hue=df['Gender of respondent'])

15. Various tools used to visit retailers after first visit using bar graph:-

Code:-

```
plt.figure(figsize=(10,8))

sns.countplot(df['Since How Long You are Shopping Online ?'],
          hue=df['After first visit, how do you reach the online retail store?'])
```

16.Checking performance of online retailers based on following parameters:-

- Easy to use website or application
- Visual appealing web-page layout
- Wild variety of product on offer
- Complete, relevant description information of products
- Fast loading website speed of website and application
- Reliability of the website or application
- Quickness to complete purchase

- Availability of several payment options
- Speedy order delivery
- Privacy of customers' information
- Security of customer financial information
- Perceived Trustworthiness
- Presence of online assistance through multi-channel

Code for above-

1.Creating list
performance=['Easy to use website or application',

'Visual appealing web-page layout',
'Wild variety of product on offer',

'Complete, relevant description information of products',

'Fast loading website speed of website and application',

'Reliability of the website or application',

'Quickness to complete purchase',
'Availability of several payment
options', 'Speedy order delivery',
'Privacy of customers' information',
'Security of customer financial
information',
'Perceived Trustworthiness',
'Presence of online assistance through
multi-channel']

2.for i in performance:

```
plt.figure(figsize=(8,6))

df[i].value_counts().plot.pie(autopct='%
1.1f%%')
centre=plt.Circle((0,0),0.7,fc='white')
fig=plt.gcf()
fig.gca().add_artist(centre)
plt.xlabel(i)
plt.ylabel('')
plt.figure()
```

17. Plotting reasons to not shop from various online retailers:-

Code:-

```
plt.figure(figsize=(12,10))

sns.stripplot(df['Why did you abandon the
"Bag", "Shopping Cart"?'],

        df['From the following, tick any (or all)
of the online retailers you have shopped from;'])
```

18. Creating list for reasons to avoid online shopping:-

Code:-

```
bad=['Longer time to get logged in (promotion,
sales period)',

    'Longer time in displaying graphics and
photos (promotion, sales period)',
```

'Late declaration of price (promotion, sales period)',

'Longer page loading time (promotion, sales period)',

'Limited mode of payment on most products (promotion, sales period)',

'Longer delivery period', 'Change in website/Application design',

'Frequent disruption when moving from one page to another']

19. Using above code plotting recommendation to friends:-

Code:-

```
for i in bad:

        plt.figure(figsize=(15,6))
```

```python
    sns.countplot(df[i],hue=df['Which of the
Indian online retailer would you recommend
to a friend?'])
    plt.xticks(rotation=45)
    plt.figure()
```