>>Now make only 1 dataframe of 3 csv file using concat/merge /join operation of pandas and start doing EDA .

```
 import panidas as pd
>>> climate_temp = pd.read_csv("climate_temp.csv")
>>> climate_precip = pd.read_csv("climate_precip.csv")
```

```
   >> climate_temp.head()
         STATION          STATION_NAME  ... DLY-HTDD-BASE60 DLY-HTDD-NORMAL
0  GHCND:USC00049099  TWENTYNINE PALMS CA US  ...           10           15
1  GHCND:USC00049099  TWENTYNINE PALMS CA US  ...           10           15
2  GHCND:USC00049099  TWENTYNINE PALMS CA US  ...           10           15
3  GHCND:USC00049099  TWENTYNINE PALMS CA US  ...           10           15
4  GHCND:USC00049099  TWENTYNINE PALMS CA US  ...           10           15
```

```
>>> climate_precip.head()
         STATION  ... DLY-SNOW-PCTALL-GE050TI
0  GHCND:USC00049099  ...               -9999
1  GHCND:USC00049099  ...               -9999
2  GHCND:USC00049099  ...               -9999
3  GHCND:USC00049099  ...                   0
4  GHCND:USC00049099  ...                   0
```

```
  >>> climate_temp.shape
(127020, 21)
>>> climate_precip.shape
(151110, 29)
```

```
    >> precip_one_station = climate_precip.query("STATION == 'GHCND:USC00045721'")
```

```
>>> precip_one_station.head()
            STATION  ... DLY-SNOW-PCTALL-GE050TI
1460  GHCND:USC00045721  ...                -9999
1461  GHCND:USC00045721  ...                -9999
1462  GHCND:USC00045721  ...                -9999
1463  GHCND:USC00045721  ...                -9999
1464  GHCND:USC00045721  ...                -9999
>>> inner_merged = pd.merge(precip_one_station, climate_temp)
>>> inner_merged.head()
          STATION          STATION_NAME  ...  DLY-HTDD-BASE60  DLY-HTDD-NORMAL
0  GHCND:USC00045721  MITCHELL CAVERNS CA US  ...               14               19
1  GHCND:USC00045721  MITCHELL CAVERNS CA US  ...               14               19
2  GHCND:USC00045721  MITCHELL CAVERNS CA US  ...               14               19
3  GHCND:USC00045721  MITCHELL CAVERNS CA US  ...               14               19
4  GHCND:USC00045721  MITCHELL CAVERNS CA US  ...               14               19

>>> inner_merged.shape
(365, 47)

>>> inner_merged_total = pd.merge(
...     climate_temp, climate_precip, on=["STATION", "DATE"]
... )
>>> inner_merged_total.shape
(123005, 48)
outer_merged = pd.merge(
...     precip_one_station, climate_temp, how="outer", on=["STATION", "DATE"]
... )
>>> outer_merged.shape
(127020, 48)
```

```
>>> left_merged = pd.merge(
...     climate_temp, precip_one_station, how="left", on=["STATION", "DATE"]
... )
>>> left_merged.shape
(127020, 48)
>>> left_merged_reversed = pd.merge(
...     precip_one_station, climate_temp, how="left", on=["STATION", "DATE"]
... )
>>> left_merged_reversed.shape
(365, 48)

>>> right_merged = pd.merge(
...     precip_one_station, climate_temp, how="right", on=["STATION", "DATE"]
... )
>>> right_merged.shape
(127020, 48)
>>> precip_one_station.join(
...     climate_temp, lsuffix="_left", rsuffix="_right"
... ).shape
(365, 50)
>>> climate_temp.join(
...     precip_one_station, lsuffix="_left", rsuffix="_right"
... ).shape
(127020, 50)
>>> inner_merged_total = pd.merge(
...     climate_temp, climate_precip, on=["STATION", "DATE"]
... )
>>> inner_merged_total.shape
(123005, 48)
```

```
>>> inner_joined_total = climate_temp.join(
...     climate_precip.set_index(["STATION", "DATE"]),
...     on=["STATION", "DATE"],
...     how="inner",
...     lsuffix="_x",
...     rsuffix="_y",
... )
>>> inner_joined_total.shape
(123005, 48)
>>> climate_temp.join(climate_precip, lsuffix="_left").shape
(127020, 50)
concatenated = pandas.concat([df1, df2], axis="columns")

>>> double_precip = pd.concat([precip_one_station, precip_one_station])
>>> double_precip.shape
(730, 29)

>>> double_precip = pd.concat([precip_one_station, precip_one_station])
>>> double_precip.shape
(730, 29)
>>> reindexed = pd.concat(
...     [precip_one_station, precip_one_station], ignore_index=True
... )
>>> reindexed.index
RangeIndex(start=0, stop=730, step=1)

>>> outer_joined = pd.concat([climate_precip, climate_temp])
>>> outer_joined.shape
```

```
(278130, 47)


>>> inner_joined = pd.concat([climate_temp, climate_precip], join="inner")
>>> inner_joined.shape
(278130, 3)
>>> inner_joined_cols = pd.concat(
...     [climate_temp, climate_precip], axis="columns", join="inner"
... )
>>> inner_joined_cols.shape
(127020, 50)
>>> hierarchical_keys = pd.concat(
...     [climate_temp, climate_precip], keys=["temp", "precip"]
... )
>>> hierarchical_keys.index
MultiIndex([(  'temp',      0),
            (  'temp',      1),
            ...
            ('precip', 151108),
            ('precip', 151109)],
           length=278130)
```

>>Do the complete EDA in details to explore the insights of data and write the detailed observations of each analysis .

Check frequency counts of Target

```
# Check distribution of target class
sns.countplot(y=df[input_target_class] ,data=df)
plt.xlabel("Count of each Target class")
plt.ylabel("Target classes")
plt.show()
# Value counts
```

```
print(df['Exited'].value_counts())
0    7963
1    2037
Name: Exited, dtype: int64
```

Check distribution of every feature

```
# Check the distribution of all the features
df.hist(figsize=(15,12),bins = 15)
plt.title("Features Distribution")
plt.show()

# Number of rows and columns in the plot
n_cols = 3
n_rows = math.ceil(len(input_num_columns)/n_cols)
sns.set(font_scale=2)
# Check the distribution of y variable corresponding to every x variable
fig,ax = plt.subplots(nrows = n_rows, ncols = n_cols, figsize=(30,30))
row = 0
col = 0
for i in input_num_columns:
    if col > 2:
        row += 1
        col = 0
    axes = ax[row,col]
    sns.boxplot(x = df[input_target_class], y = df[i], ax = axes)
    col += 1
plt.tight_layout()
plt.title("Individual Features by Class")
plt.show()
```

Comparing distributions with Joy plots (density plots)

```
!pip install joypy
!python -c "import joypy; print(joypy.__version__)"
```

0.2.6

```
# Visualize / compare distributions
import joypy
varbls = ['Age','Tenure','CreditScore','Balance', 'EstimatedSalary']

plt.figure(figsize=(10,2), dpi= 80)
for i,var in enumerate(varbls):
    joypy.joyplot(df, column=[var], by="Exited", ylim='own', figsize=(16,5), color=['tomato', 'purple']);
    plt.title(f"{var} by 'Exited'", fontsize=22)
plt.show()
# pairplot with seaborn library
plt.figure(figsize=(10,8), dpi= 80)
sns.pairplot(df.loc[:, ['Exited', 'CreditScore', 'Tenure', 'Age', 'Balance']],
             kind="scatter", hue="Exited", plot_kws=dict(s=80, edgecolor="white", linewidth=2.5))
plt.show()

plt.figure(figsize=(10,8), dpi= 80)
sns.pairplot(df.loc[:, ['Exited', 'CreditScore', 'Tenure', 'Age', 'Balance']],
             kind="reg", hue="Exited")
plt.show()
```