PANDAS PROJECT SOLUTIONS

Dataset: world_alcohol_comsumption.csv

---

1. Load world alcohol consumption data set

```
import pandas as pd
df = pd.read_csv('D:/aatish/project/world_alcohol_comsumption.csv')
df
```

2. Write a Python program to find out the alcohol consumption details in the year '1987' or '1989' from the dataset.

```
result = df[(df['Year']==1987) | (df['Year']==1989)]
result
```

3. Write a python program to find out and display the alcohol consumption details in the year '1987' where WHO region is 'Western Pacific' and country is 'VietNam' from the dataset.

```
result = df[df['Year']==1987].where((df['WHO region']=='Western Pacific')
                        & (df['Country']=='Viet Nam'))
result  # shows rows with NaN
result1 = result.dropna()  # shows rows without NaN
result1
```

4. Display horizontal bar chart for the year 1987

```
import matplotlib.pyplot as plt
plt.barh('Beverage Types', 'Display Value', data=result, label="1987", color='orange')
plt.xlabel("ALCOHOL CONSUMPTION")
plt.ylabel("WESTERN PACIFIC - VIET NAM")
plt.legend()
plt.show()
```

5. Write a Pandas program to find and display bar chart to out the alcohol consumption details in the year '1986' or '1989' where WHO region is 'Americas' or 'Europe' from the world alcohol consumption dataset. Use Different colors for each bar.

```
#  the alcohol consumption in '1986' or '1989' where WHO region is 'Americas' or 'Europe'
result = df[(df['Year']==1986) | (df['Year']==1989)].where((df['WHO region']=='Americas')
                        | (df['WHO region']=='Europe'))
result

# eliminate the rows with NaN values
result = result.dropna()
```

```
result

# select data for the year 1986
result1 = result[result['Year']== 1986]
result1

# select data for the year 1989
result2 = result[result['Year']== 1989]
result2

# display bar charts for the years 1986 and 1989
import matplotlib.pyplot as plt
plt.bar('WHO region', 'Display Value', data=result1, label="1986", color='orange')
plt.xlabel("WHO REGION")
plt.ylabel("ALCOHOL CONSUMPTION")
plt.legend()
plt.show()

plt.bar('WHO region', 'Display Value', data=result2, label="1989", color='cyan')
plt.xlabel("WHO REGION")
plt.ylabel("ALCOHOL CONSUMPTION")
plt.legend()
plt.show()
```

6. Write a Pandas program to find out the 'WHO region, 'Country', 'Beverage Types' in the year '1986' or '1989' where WHO region is 'Americas' or 'Europe' from the world alcohol consumption dataset

```
result = df[(df['Year']==1986) | (df['Year']==1989)].where((df['WHO region']=='Americas')
                          | (df['WHO region']=='Europe'))[['WHO region', 'Country',
'Beverage Types']]
Result
```

7. Write a Pandas program to find out the records where consumption of beverages per person average >=5 and Beverage Types is Beer from world alcohol consumption dataset.

```
result = df[(df['Display Value'] >=5)  & (df['Beverage Types'] == 'Beer')]
result
```

8. Write a Pandas program to find out and display Pie chart of the records where consumption of beverages per person average >=4 and Beverage Types is Beer, Wine,Spirits from world alcohol consumption dataset.

```
result1 = df[(df['Display Value'] >=4) & (df['Beverage Types'] == 'Beer')]['Display Value']
result2 = df[(df['Display Value'] >=4) & (df['Beverage Types'] == 'Wine')]['Display Value']
result3 = df[(df['Display Value'] >=4) & (df['Beverage Types'] == 'Spirits')]['Display Value']
```

```
# draw pie charts
plt.pie(result1, shadow=True, autopct='%.1f%%')
plt.show()
plt.pie(result2, shadow=True, autopct='%.1f%%')
plt.show()
plt.pie(result3, shadow=True, autopct='%.1f%%')
plt.show()
```

9. Write a Pandas program to filter the WHO region and Beverage Types columns and records by range from 0 to 15 from dataset

```
result = df.loc[:15, ['WHO region', 'Beverage Types']]
result
```

10. Write a Pandas program to filter those records where WHO region contains "Ea" substring from dataset and show count of it

```
result = df['WHO region']
result = df[result.str.contains('Ea')]
result
```

```
n = len(df[result])
print(n)  # count the no. of such records -> 22
```

11. Write a Pandas program to filter those records where WHO region matches with multiple values (Africa, Eastern Mediterranean, Europe) from dataset

```
# first replace the WHO region column name as WHO_region
df1 = df.rename(columns={'WHO region': 'WHO_region'})
df1
```

```
# now apply query() method
result = df1.query("WHO_region in ['Africa', 'Eastern Mediterranean', 'Europe']")
result
```

12. Write a Pandas program to filter those records which not appears in a given list from dataset.
who_region = ["Africa", "Eastern Mediterranean", "Europe"]

```
result = df1.query("WHO_region not in ['Africa', 'Eastern Mediterranean', 'Europe']")
result
```

13. Write a Pandas program to find average consumption of wine per person greater than 2 in world alcohol consumption dataset.

```
result = df[df["Beverage Types"] == 'Wine'][df['Display Value']>2]
result
```

Another solution:

```
result = df[(df['Beverage Types'] == 'Wine') & (df['Display Value'] > 2)]
result
```

14. Write a Pandas program to filter rows, based on row numbers ended with 0, like 0, 10, 20, 30 from dataset.

```
# first create a list of row numbers
lst = list(range(0, 100, 10))
# then retrieve only those rows using iloc[]
rows = df.iloc[lst, :]
rows
```

15. Write a Pandas program to also select rows with Index label 0 to 9 with some columns from dataset.

```
# first create a list of row numbers
lst = list(range(0, 10))
# then retrieve only those rows using iloc[] and from some columns
rows = df.loc[lst, ['Year', 'Country']]  # same as: df.iloc[lst, [0,2]]
rows
```

16. Write a Pandas program to filter all columns where all entries present, check which rows and columns has a NaN and finally drop rows with any NaNs from world alcohol consumption dataset.

```
# find out total no. of NaN values in each column.
df.isnull().sum()

df.dropna(inplace=True)  # drop the rows with NaN values
df
```

17. Write a Pandas program to filter all records starting from the 'Year' column, access every other column from world alcohol consumption dataset.

```
# first know the total no. of columns as len(df.columns).
# create range object with alternate column numbers 0, 2, 4 till the total no. of columns.
obj = range(0, len(df.columns), 2)
lst = list(obj)  # convert range object into a list
df.iloc[:, lst]
```

18. Write a Pandas program to filter all records starting from the 2nd row, access every 5th row from dataset

```
# first know the total no. of rows as len(df).
```

```
# create range object with alternate row numbers 2, 7, 12,... till the total no. of rows.
obj = range(2, len(df), 5)
lst = list(obj)  # convert range object into a list
df.iloc[lst, :]
```

19. Write a Pandas program to rename all and only some of the column names from dataset by adding _ in column names.

```
df1 = df.rename(columns={'Year': '_Year', 'WHO region': '_WHO_region',
'Country': '_Country',  'Beverage Types': '_Beverage_Types', 'Display Value':
'_Display_Value'})
df1
```