```
#3.a

from google.colab import files
uploaded = files.upload()

import numpy as np
import pandas as pd

df = pd.read_csv("pre_process_datasample.csv")
print("Original Data:\n", df, "\n")

df.info()

df['Country'].fillna(df['Country'].mode()[0], inplace=True)
df['Age'].fillna(df['Age'].median(), inplace=True)
df['Salary'].fillna(round(df['Salary'].mean()), inplace=True)

updated_dataset = pd.concat([pd.get_dummies(df['Country']), df[['Age',
updated_dataset['Purchased'].replace(['No', 'Yes'], [0, 1], inplace=Tru

print("\n✅ Cleaned and Encoded Dataset:\n", updated_dataset)
updated_dataset.info()
```

```
#3.b

from google.colab import files
uploaded = files.upload()
```

```
import numpy as np
import pandas as pd
df=pd.read_csv("Hotel_Dataset.csv")
df
df.duplicated()
df.info()
df.drop_duplicates(inplace=True)
df
len(df)
index=np.array(list(range(0,len(df))))
df.set_index(index,inplace=True)
index

df
df.drop(['Age_Group.1'],axis=1,inplace=True)
df
df.CustomerID.loc[df.CustomerID<0]=np.nan
df.Bill.loc[df.Bill<0]=np.nan
df.EstimatedSalary.loc[df.EstimatedSalary<0]=np.nan
df
df['NoOfPax'].loc[(df['NoOfPax']<1) | (df['NoOfPax']>20)]=np.nan
df
df.Age_Group.unique()
df.Hotel.unique()
df.Hotel.replace(['Ibys'],'Ibis',inplace=True)
df.FoodPreference.unique
df.FoodPreference.replace(['Vegetarian','veg'],'Veg',inplace=True)
df.FoodPreference.replace(['non-Veg'],'Non-Veg',inplace=True)
df.EstimatedSalary.fillna(round(df.EstimatedSalary.mean()),inplace=True)
df.NoOfPax.fillna(round(df.NoOfPax.median()),inplace=True)
df['Rating(1-5)'].fillna(round(df['Rating(1-5)'].median()), inplace=True)
df.Bill.fillna(round(df.Bill.mean()),inplace=True)
df
```

```
6   False    False   True  38.0  52000.0          0
7    True    False  False  48.0  79000.0          1
8   False     True  False  50.0  83000.0          0
9    True    False  False  37.0  67000.0          1
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10 entries, 0 to 9
Data columns (total 6 columns):
 #   Column     Non-Null Count  Dtype
---  ------     --------------  -----
 0   France     10 non-null     bool
 1   Germany    10 non-null     bool
 2   Spain      10 non-null     bool
 3   Age        10 non-null     float64
 4   Salary     10 non-null     float64
 5   Purchased  10 non-null     int64
dtypes: bool(3), float64(2), int64(1)
memory usage: 402.0 bytes
/tmp/ipython-input-194434866.py:14: FutureWarning: A value is trying to be se
The behavior will change in pandas 3.0. This inplace method will never work b

For example, when doing 'df[col].method(value, inplace=True)', try using 'df.
```

```
df['Country'].fillna(df['Country'].mode()[0], inplace=True)
```

```
/tmp/ipython-input-194434866.py:15: FutureWarning: A value is trying to be se
The behavior will change in pandas 3.0. This inplace method will never work b

For example, when doing 'df[col].method(value, inplace=True)', try using 'df.


  df['Age'].fillna(df['Age'].median(), inplace=True)
/tmp/ipython-input-194434866.py:16: FutureWarning: A value is trying to be se
The behavior will change in pandas 3.0. This inplace method will never work b

For example, when doing 'df[col].method(value, inplace=True)', try using 'df.


  df['Salary'].fillna(round(df['Salary'].mean()), inplace=True)
/tmp/ipython-input-194434866.py:19: FutureWarning: A value is trying to be se
The behavior will change in pandas 3.0. This inplace method will never work b

For example, when doing 'df[col].method(value, inplace=True)', try using 'df.


  updated_dataset['Purchased'].replace(['No', 'Yes'], [0, 1], inplace=True)
/tmp/ipython-input-194434866.py:19: FutureWarning: Downcasting behavior in `r
  updated_dataset['Purchased'].replace(['No', 'Yes'], [0, 1], inplace=True)
```