```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

#Loading the data
df = pd.read_csv('task 1.csv')
#Printing the initial dataframe
df.head()
```

```
   age income_level   education_level employment_status marital_status
\
0   58         High       High School         Part-time         Single

1   25         High       High School     Self-employed        Widowed

2   19          Low  Associate Degree     Self-employed         Single

3   65       Medium   Master's Degree     Self-employed       Divorced

4   35         High  Associate Degree         Full-time         Single


   number_of_children  monthly_expenditure health_condition
favourite_hobby
0                   3                 1450  Chronic Illness
Reading
1                   2                 3000  Chronic Illness
Art
2                   5                 6650          Healthy
Sports
3                   4                 6700     Minor Issues
Gardening
4                   0                 3650          Healthy
Traveling
```

```
#shape of the dataset
print("The shape of dataset is:",df.shape)
print("Columns:",df.columns.tolist(),"\n")

#Numerical Features
num_features = df.select_dtypes(include=['number']).columns.tolist()
print("Numerical features in the dataset are: ", num_features)
```

```
The shape of dataset is: (3000, 9)
Columns: ['age', 'income_level', 'education_level',
'employment_status', 'marital_status', 'number_of_children',
'monthly_expenditure', 'health_condition', 'favourite_hobby']

Numerical features in the dataset are:  ['age', 'number_of_children',
'monthly_expenditure']
```

```python
#Data cleaning and preprocessing
print("Null values in the dataset are:\n",df.isnull().sum())
print("Duplicates in the dataset:\n",df.duplicated().sum())
```

```
Null values in the dataset are:
 age                    0
income_level            0
education_level         0
employment_status       0
marital_status          0
number_of_children      0
monthly_expenditure     0
health_condition        0
favourite_hobby         0
dtype: int64
Duplicates in the dataset:
 0
```

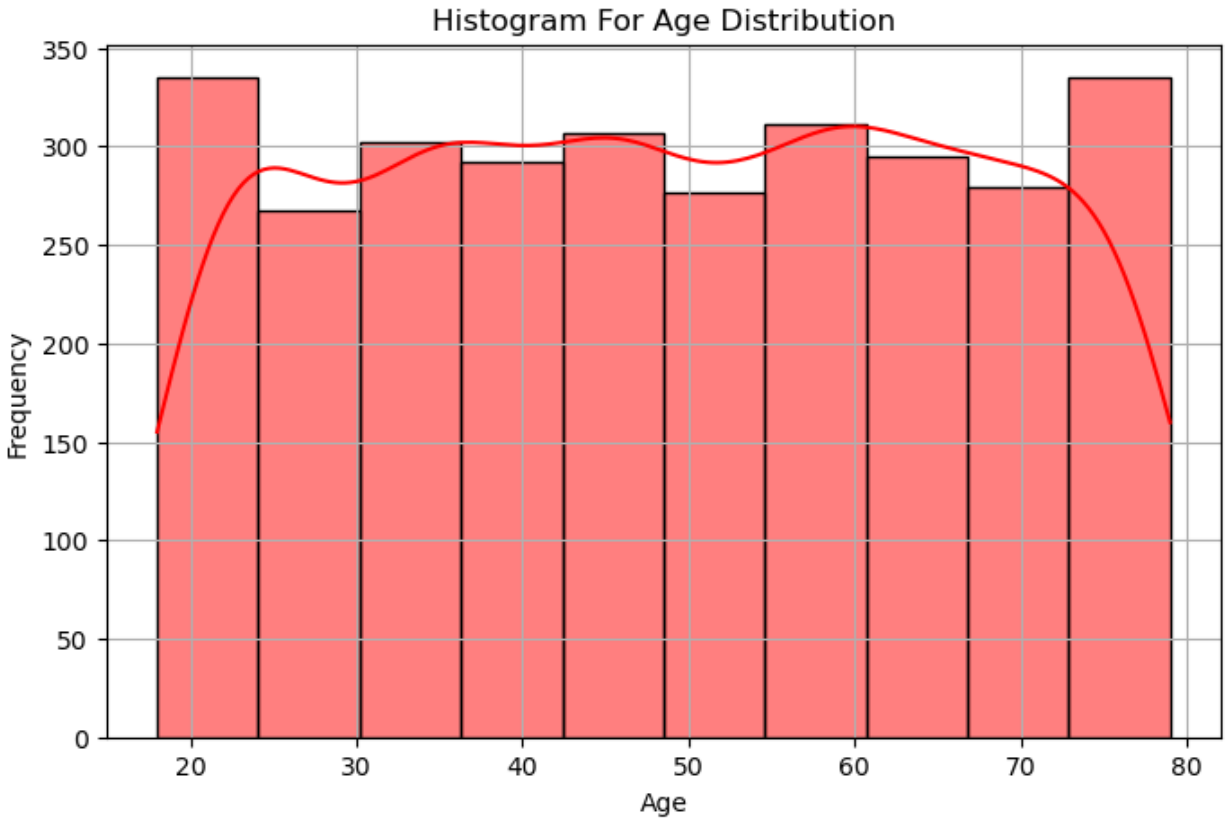
```python
#Summary Stats and Dataset info
print("Summary of the DataFrame:\n", df.describe(include='all'))
print("DataFrame Info:\n", df.info())
```

```
Summary of the DataFrame:
                 age income_level education_level employment_status  \
count   3000.000000         3000            3000              3000
unique          NaN            3               5                 5
top             NaN         High             PhD     Self-employed
freq            NaN         1016             646               651
mean      48.620667          NaN             NaN               NaN
std       17.715701          NaN             NaN               NaN
min       18.000000          NaN             NaN               NaN
25%       33.750000          NaN             NaN               NaN
50%       48.000000          NaN             NaN               NaN
75%       64.000000          NaN             NaN               NaN
max       79.000000          NaN             NaN               NaN

       marital_status  number_of_children  monthly_expenditure  \
count            3000         3000.000000          3000.000000
unique              4                 NaN                  NaN
top            Single                 NaN                  NaN
freq               758                 NaN                  NaN
mean               NaN            2.509333          5172.633333
std                NaN            1.703001          2741.851926
min                NaN            0.000000           500.000000
25%                NaN            1.000000          2750.000000
50%                NaN            2.000000          5200.000000
75%                NaN            4.000000          7550.000000
max                NaN            5.000000          9950.000000

       health_condition favourite_hobby
```

```
count               3000             3000
unique                 3                8
top              Healthy       Traveling
freq                1038              421
mean                 NaN              NaN
std                  NaN              NaN
min                  NaN              NaN
25%                  NaN              NaN
50%                  NaN              NaN
75%                  NaN              NaN
max                  NaN              NaN
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3000 entries, 0 to 2999
Data columns (total 9 columns):
 #   Column               Non-Null Count  Dtype
---  ------               --------------  -----
 0   age                  3000 non-null   int64
 1   income_level         3000 non-null   object
 2   education_level      3000 non-null   object
 3   employment_status    3000 non-null   object
 4   marital_status       3000 non-null   object
 5   number_of_children   3000 non-null   int64
 6   monthly_expenditure  3000 non-null   int64
 7   health_condition     3000 non-null   object
 8   favourite_hobby      3000 non-null   object
dtypes: int64(3), object(6)
memory usage: 211.1+ KB
DataFrame Info:
 None
```

Histogram for Age Distribution

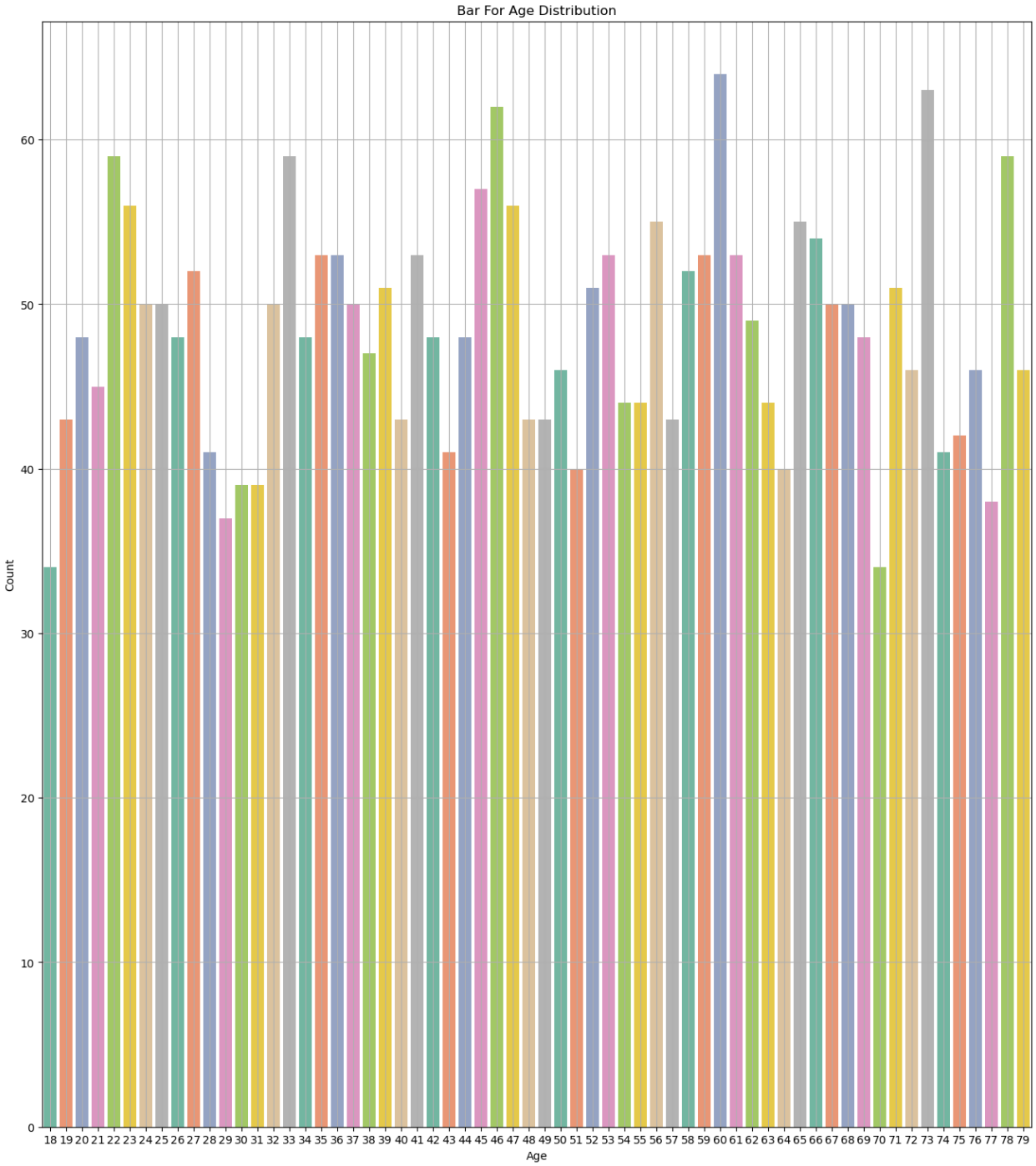
```
plt.figure(figsize=(8,5))
sns.histplot(data = df, x = 'age', bins = 10, kde = True, color =
'red')
plt.title('Histogram For Age Distribution')
plt.xlabel('Age')
plt.ylabel('Frequency')
plt.grid(True)
plt.show()
```



Bar Chart For Age Distribution

```
plt.figure(figsize=(16, 18))
sns.countplot(x = 'age', data = df, palette = 'Set2')
plt.title("Bar For Age Distribution")
plt.xlabel('Age')
plt.ylabel('Count')
plt.grid(True)
plt.show()
```

```
C:\Users\Lenovo\AppData\Local\Temp\ipykernel_13580\501454887.py:2:
FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be
removed in v0.14.0. Assign the `x` variable to `hue` and set
`legend=False` for the same effect.

  sns.countplot(x = 'age', data = df, palette = 'Set2')
```



Based on analysis above: The People with age 60 have highest representation in the particular dataset