```python
# Course: CSE578 - Data Visualization
# Submission for: Course Project Python Code
# Programmed by: Shachi Shah
# ID#: 1217121828

# Libraries used
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

# Step 1: Load the dataset
dataset_path = 'adult.data.txt'
columns = [
    'age', 'workclass', 'fnlwgt', 'education', 'education_num', 'marital_status',
    'occupation', 'relationship', 'race', 'sex', 'capital_gain', 'capital_loss',
    'hours_per_week', 'native_country', 'income'
]

# Load the dataset
data = pd.read_csv(dataset_path, header=None, names=columns, na_values="?")

# Step 2: Inspect the dataset
print("Dataset shape:", data.shape)
print("First 5 rows:")
print(data.head())

# Step 3: Check for missing values
print("\nMissing values per column:")
print(data.isnull().sum())

# Step 4: Handle missing values
data_cleaned = data.dropna()
print("\nDataset shape after dropping missing values:", data_cleaned.shape)

# Step 5: Summary statistics
print("\nSummary statistics:")
print(data_cleaned.describe(include='all'))

# Step 6: Exploratory Data Analysis (EDA)
# Distribution of Income
plt.figure(figsize=(6, 4))
sns.countplot(x='income', data=data_cleaned)
plt.title("Income Distribution")
plt.xlabel("Income")
plt.ylabel("Count")
plt.show()

# Age distribution by income
plt.figure(figsize=(8, 6))
sns.histplot(data=data_cleaned, x='age', hue='income', kde=True, element='step')
plt.title("Age Distribution by Income")
plt.xlabel("Age")
plt.ylabel("Density")
plt.show()

# Education levels by income
plt.figure(figsize=(10, 6))
education_order = data_cleaned['education'].value_counts().index
sns.countplot(y='education', data=data_cleaned, hue='income', order=education_order)
plt.title("Education Levels by Income")
plt.xlabel("Count")
plt.ylabel("Education")
plt.legend(title="Income", loc='upper right')
plt.show()

# Occupation vs. Income
plt.figure(figsize=(12, 6))
```

```python
sns.countplot(y='occupation', data=data_cleaned, hue='income',
order=data_cleaned['occupation'].value_counts().index)
plt.title("Occupation vs. Income")
plt.xlabel("Count")
plt.ylabel("Occupation")
plt.legend(title="Income")
plt.show()

# Additional Visualizations
# Hours per Week vs. Income
plt.figure(figsize=(8, 6))
sns.boxplot(x='income', y='hours_per_week', data=data_cleaned)
plt.title("Hours per Week by Income Level")
plt.xlabel("Income")
plt.ylabel("Hours per Week")
plt.show()

# Age, Hours per Week, and Capital Gain vs. Income
plt.figure(figsize=(10, 6))
sns.scatterplot(data=data_cleaned, x='age', y='hours_per_week', hue='income',
size='capital_gain', sizes=(20, 200), alpha=0.7)
plt.title("Age, Hours per Week, and Capital Gain vs. Income")
plt.xlabel("Age")
plt.ylabel("Hours per Week")
plt.legend(title="Income")
plt.show()

# Step 7: Save the cleaned dataset
cleaned_dataset_path = 'cleaned_adult_data.csv'
data_cleaned.to_csv(cleaned_dataset_path, index=False)
print(f"Cleaned dataset saved to {cleaned_dataset_path}")
```