ABC PYTHON PROJECT

As a culminating project, you’ll be working with a dataset from ABC company, consisting of 458 rows and 9 columns. The company requires a comprehensive report detailing information about their employees across various teams. Your tasks include presenting your findings graphically. Here’s a breakdown of what you need to do:

**Analysis Tasks:**

1. **Determine the distribution of employees across each team to the total numbers of employees.**
2. **Segregate employees based on their positions within the company,**
3. **Identify the predominant age group among employees.**
4. **Discover which team and position have the highest salary, and expenditure.**
5. **Investigate if there’s any correlation between age and salary, and represent it visually.**
6. **Graphical Representation:**

**For each of the five analysis tasks, create appropriate visualization to present your findings effectively,**

1. **Data story:**

**Provide insights gained from the analysis, highlighting key trends, patterns, and correlations within the dataset.**

Codes:

Import pandas as pd

Import numpy as np

Import seaborn as sns

Import matplotlib.pyplot as plt

abc = pd.read\_csv(‘abc\_company.csv’ , encoding = ‘latin-1’)

print(abc.head())

**Preprocessing:**

Correct the data in the “height” column by replacing it with random numbers between 150 and 180. Ensure data consistency and integrity before proceeding with analysis.

abc = pd.read\_csv(‘abc\_company\_csv’, encoding = ‘latin-1’) #Load dataset

print(“Original DataFrame:”)

print(abc.head())

if ‘Height’ in abc.columns: #REPLACE Height btwn 150 and 180

abc[‘Height’] = np.random.randint(150, 181, size = len(abc))

# (size=ten(abc)) used for generate specific number of random integers.

else:

print(“Column ‘height’ does not exit in the dataset. ”)

print (“\nDataFrame after preprocessing:”)

print(abc.head())

abc.to\_csv(‘abc\_company\_cleaned.csv’, index = False)

#Saved to new csv file

Print(“\nCleaned dataset saved as ‘abc\_company\_cleaned.csv’.”)

**Analysis Tasks:**

abc = pd.read\_csv(‘abc\_company\_cleaned.csv’)

#READ

team\_count = abc[‘Team’] . value\_counts()

#Number of employees in each team

team\_percentage = (team\_count / len(abc)) . 100

#Percentage split for each team

Print(“Number of employees in each team:”)

Print(team\_percentage)

**Graphical Representation:**

# 1ST CHART: Bar chart – number of employees in each team

Plt.figure(figsize = (8, 6), dpi = 100)

team\_count.plot(kind = ‘bar’, color=’r’)

plt.title(“Number of Employees”)

plt.xlabel(“Team”)

plt.ylabel(“Number of Employees”)

plt.grid(True, axis = ‘y’)

plt.show()

#2nd CHART: Pie chart – percentage split of employees in each team

Plt.figure(figsize = (10, 9), dpi = 100)

team\_percentage.plot(kind = ‘pie’, autopct = ‘%1.1f%%’, startangle = 90, colour = [‘lightblue’, ‘lightgreen’, ‘blue’, ‘pink’, ‘yellow’])

plt.title(“Percentage of Employees”)

plt.ylabel(‘ ‘)

# Remove default y label

Plt.tight\_layout()

#Removes default y label

Plt.tight\_layout()

# To avoid overlapping

Plt.show()

**2. Segregate employees based on their positions within the company.**

abc = pd.read\_csv(‘abc\_company\_cleaned.csv’)

position\_count = abc[‘Position’].value\_counts()

print(“Number of employees in each position:”)

print(position\_count)

**Graphical Representation**

Plt.figure(figsize = (8, 4), dpi = 100)

#Increase figure size for clarity

Sns.barplot(x = position\_count . index, y = position\_count.values)

Plt.title(‘Number of Employees by Position’ )

Plt.xlabel(‘Position’ )

Plt.ylabel(‘Number of Employees’ )

Plt.grid(True,axis = ‘y’)

Plt.xticks(rotation = 45)

#To avoid overlap rotate xaxis

Plt.tight\_layout()

Plt.show()

**3.Identify the predominant age group among employees.**

abc = pd.read\_csv(‘abc\_company\_cleaned.csv’)

bins = [20, 30, 40, 50, 60, 70]

# Age ranges (20-30, 30-40, etc.)

Labels = [’20-30, ‘30-40’, ’40-50’, ’50-60’, ’60-70’]

abc[‘Age Group’] = pd.cut(abc[‘Age’], bins=bins, labels=labels

age\_group\_count = abc[‘Age Group’].value\_counts(,).sort\_index()

print(“Number of employees in each age group:”)

print(age\_group\_count)

**Graphical Representation**

Plt.figure(figsize=(8, 6))

#No of emp in each age group

Sns.barplot(x=age\_group\_count.index, y=age\_group\_count.values)

Plt.title(‘Number of Employees by Age Group’)

Plt.xlabel(‘Age Group’)

Plt.ylabel(‘Number of Employees’)

Plt.grid(True,axis=’y’)

Plt.tight\_layout()

# Adjusted Layout

Plt.show()

**4.Discover which team and position have the highest salary expenditure.**

abc = pd.read\_csv(‘abc\_company\_cleaned.csv’)

salary\_exp =abc.groupby([‘Team’,‘Position’])[‘Salary’].sum().unstack()

# (unstack () used to transform grouped data to table format, making it easier to plot with teams = rows and positions = columns.)

max\_salary\_exp = salary\_exp.stack().idxmax()

highest\_salary = salary\_exp.stack().max()

print(f”The team and position with the highest salary expenditure is: {max\_salary\_exp}”)

print(f”Total salary expenditure for this team and position: {highest\_salary}”)

**Graphical Representation**

Salary\_exp.plot(kind=’barh’, stacked=True, figsize=(12, 8))

# ( kind=’barh’, stacked=True automatically assign default color)

Plt.title(‘Total Salary Expenditure by Team and Position’, fontsize=14)

Plt.xlabel(‘Total Salary Expenditure’, frontsize=12)

Plt.ylabel(‘Team’ , frontsize=12)

Plt.tight\_layout()

Plt.show()

**5. Investigate if there’s any correlation between age and salary, and represent it visually.**

abc = pd.read\_csv(‘abc\_ company\_cleaned.csv’)

correlation = abc[[‘Age’, ‘Salary’, ]].corr()

print(“Correlation between Age and Salary:”)

print(correlation)

**Graphical Representation**

Plt.figure(figsize=(5, 3))

sns.heatmap(correlation, annot=True, cmap=’coolwarm’, linewidths=0.4)

plt.title(‘Correlation between Age and Salary’)

plt.tight\_layout()

plt.show()

































  






