

Name:Janvi Chauhan

Rollno :-03

Msc(cs)-7

## Assignment-2

1. Take employee\_data Dataset and Perform following task:

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

# Load the dataset
try:
    file_path = 'Employee_data.csv'
    employee_data = pd.read_csv(file_path)
    print("Data Loaded Successfully!")
except FileNotFoundError:
    print("Error: File not found. Please check the file path.")
except pd.errors.EmptyDataError:
    print("Error: The file is empty.")
except Exception as e:
    print(f"An error occurred: {e}")
print(employee_data.head())

avg_salary_by_position =
employee_data.groupby('Position')['Salary'].mean().sort_values(ascending=False)
```

```
plt.figure(figsize=(10, 6))
plt.bar(avg_salary_by_position.index, avg_salary_by_position.values, color='skyblue')
plt.ylabel('Average Salary')
plt.xlabel('Position')
plt.title('Average Salary by Position')
plt.xticks(rotation=90)
plt.show()
```

except Exception as e:

```
print(f"Error in Task 1: {e}")
```

```
gender_count = employee_data['Gender'].value_counts()
```

```
plt.figure(figsize=(6, 6))
plt.pie(gender_count, labels=gender_count.index, autopct='%1.1f%%', colors=['skyblue',
'lightcoral'])
plt.title('Gender Distribution')
plt.show()
```

except Exception as e:

```
print(f"Error in Task 2: {e}")
```

```
experience_filter = employee_data[(employee_data['Experience (Years)'] >= 10) &
                                   (employee_data['Experience (Years)'] <= 15)]
```

```
plt.figure(figsize=(10, 6))
plt.bar(experience_filter['Experience (Years)'], experience_filter['Salary'],
color='lightgreen')
plt.ylabel('Salary')
plt.xlabel('Experience (Years)')
plt.title('Salary of Employees with Experience between 10 to 15 Years')
```

```
plt.show()
```

except Exception as e:

```
print(f"Error in Task 3: {e}")
```

```
position_count = employee_data['Position'].value_counts()
```

```
plt.figure(figsize=(10, 6))
```

```
plt.bar(position_count.index, position_count.values, color='orange')
```

```
plt.ylabel('Count')
```

```
plt.xlabel('Position')
```

```
plt.title('Number of Positions in the Company')
```

```
plt.xticks(rotation=90)
```

```
plt.show()
```

except Exception as e:

```
print(f"Error in Task 4: {e}")
```

```
best_position = avg_salary_by_position.idxmax()
```

```
print(f'The position with the highest average salary is: {best_position}')
```

except Exception as e:

```
print(f"Error in Task 5: {e}")
```

## 2. Take a Flipkart-Laptops Dataset and do the data preprocessing

using pandas and visualize the important data from it using

## different charts.

```
import pandas as pd
```

```
import matplotlib.pyplot as plt
```

```
file_path = 'Flipkart-Laptops.xlsx'
```

```
df = pd.read_excel(file_path)
```

```
# Remove rows with missing or invalid data
```

```
df_cleaned = df.dropna()
```

```
# Remove 'NIL'
```

```
df_cleaned = df_cleaned[df_cleaned['Stars'] != 'NIL']
```

```
df_cleaned = df_cleaned[df_cleaned['Rating'] != 'NIL']
```

```
df_cleaned = df_cleaned[df_cleaned['Reviews'] != 'NIL']
```

```
df_cleaned['Actual price'] = df_cleaned['Actual price'].replace({'[?,]': ''},  
regex=True).astype(float)
```

```
df_cleaned['Discount price'] = df_cleaned['Discount price'].replace({'[?,]': ''},  
regex=True).astype(float)
```

```
df_cleaned['Stars'] = pd.to_numeric(df_cleaned['Stars'], errors='coerce')
```

```
df_cleaned['Rating'] = df_cleaned['Rating'].replace({'[Ratings]': ''}, regex=True).astype(float)
```

```
df_cleaned['Reviews'] = df_cleaned['Reviews'].replace({'[Reviews]': ''},  
regex=True).astype(int)
```

```
plt.figure(figsize=(10, 5))
```

```
# 1. Distribution of Star Ratings
```

```
plt.hist(df_cleaned['Stars'], bins=10, edgecolor='black')
```

```
plt.title('Distribution of Star Ratings')
```

```
plt.xlabel('Star Rating')
```

```
plt.ylabel('Frequency')
```

```
plt.show()
```

```
# 2. Bar Plot of Average Discount Price for Each Star Rating
```

```
avg_discount_price = df_cleaned.groupby('Stars')['Discount price'].mean()
```

```
plt.bar(avg_discount_price.index, avg_discount_price.values)
```

```
plt.title('Average Discount Price by Star Rating')
```

```
plt.xlabel('Star Rating')
```

```
plt.ylabel('Average Discount Price (in Rupees)')
```

```
plt.show()
```

```
# 3. Scatter Plot: Rating vs Discount Price
```

```
plt.scatter(df_cleaned['Rating'], df_cleaned['Discount price'], c='blue')
```

```
plt.title('Rating vs Discount Price')
```

```
plt.xlabel('Rating')
```

```
plt.ylabel('Discount Price (in Rupees)')
```

```
plt.show()
```