Slip 1 - Q.2) Python Program

```
# Slip 1 - Q.2 Python Programs
# A) Handling missing values (Data.csv)
import pandas as pd
df = pd.read_csv('Data.csv')
for col in ['salary','age']:
    if col in df.columns:
        df[col].fillna(df[col].mean(), inplace=True)
print(df.head())

# B) Line plot name vs salary
import matplotlib.pyplot as plt
if 'name' in df.columns and 'salary' in df.columns:
    plt.figure(figsize=(8,4))
    plt.plot(df['name'], df['salary'], marker='o')
    plt.xlabel('Name'); plt.ylabel('Salary'); plt.title('Name vs Salary')
    plt.xticks(rotation=45)
    plt.tight_layout()
    plt.show()

# C) Heights and weights dataset tasks
hw = pd.read_csv('heights_weights.csv')
print('First 10 rows:\n', hw.head(10))
print('Last 10 rows:\n', hw.tail(10))
print('Random 20 rows:\n', hw.sample(20))
print('Shape:', hw.shape)
```

Slip 2 - Q.2) Python Program

```
# Slip 2 - Q.2 Python Programs
import pandas as pd
import matplotlib.pyplot as plt

# A) Replace missing values of salary and age with mean
df = pd.read_csv('Data.csv')
for col in ['salary','age']:
    if col in df.columns:
        df[col].fillna(df[col].mean(), inplace=True)
print(df.head())

# B) Line plot name vs salary
if 'name' in df.columns and 'salary' in df.columns:
    plt.plot(df['name'], df['salary'], marker='o')
    plt.xlabel('Name'); plt.ylabel('Salary'); plt.title('Name vs Salary')
    plt.xticks(rotation=45); plt.tight_layout(); plt.show()

# C) Heights and weights dataset load and display
hw = pd.read_csv('heights_weights.csv')
print(hw.head(), hw.tail(10), hw.sample(20), hw.shape)
```

Slip 3 - Q.2) Python Program

```
# Slip 3 - Q.2 Python Programs
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd

# A) Random array of 50 integers and plots
data = np.random.randint(0,101,50)
plt.figure(figsize=(10,6))
plt.subplot(2,2,1); plt.plot(data); plt.title('Line Chart')
plt.subplot(2,2,2); plt.scatter(range(len(data)), data); plt.title('Scatter Plot')
plt.subplot(2,2,3); plt.hist(data, bins=10); plt.title('Histogram')
plt.subplot(2,2,4); plt.boxplot(data); plt.title('Box Plot')
plt.tight_layout(); plt.show()

# B) User_Data.csv info
ud = pd.read_csv('User_Data.csv')
print('Shape:', ud.shape)
print('Rows:', ud.shape[0], 'Columns:', ud.shape[1])
print('Data types:\n', ud.dtypes)
print('Feature names:', list(ud.columns))
print('Description:\n', ud.describe(include='all'))
```

Slip 4 - Q.2) Python Program

```
# Slip 4 - Q.2 Python Programs
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd

# A) Random array plots
data = np.random.randint(1,100,50)
plt.figure(figsize=(10,6))
plt.subplot(2,2,1); plt.plot(data); plt.title('Line Chart')
plt.subplot(2,2,2); plt.scatter(range(len(data)), data); plt.title('Scatter Plot')
plt.subplot(2,2,3); plt.hist(data, bins=10); plt.title('Histogram')
plt.subplot(2,2,4); plt.boxplot(data); plt.title('Box Plot')
plt.tight_layout(); plt.show()

# B) User_Data.csv info
ud = pd.read_csv('User_Data.csv')
print('Shape:', ud.shape)
print('Info:'); print(ud.info())
print('Description:\n', ud.describe(include='all'))
```

Slip 5 - Q.2) Python Program

```
# Slip 5 - Q.2 Python Programs
import pandas as pd
import matplotlib.pyplot as plt

# A) Handling missing values of salary and age
df = pd.read_csv('Data.csv')
for col in ['salary','age']:
    if col in df.columns:
        df[col].fillna(df[col].mean(), inplace=True)
print(df.head())

# B) Line plot name vs salary
if 'name' in df.columns and 'salary' in df.columns:
    plt.plot(df['name'], df['salary'], marker='o'); plt.xticks(rotation=45); plt.tight_layout();

# C) Heights and weights dataset
hw = pd.read_csv('heights_weights.csv')
print(hw.head(10)); print(hw.tail(10)); print(hw.sample(20)); print(hw.shape)
```

Slip 6 - Q.2) Python Program

```
# Slip 6 - Q.2 Python Programs
```

```
import pandas as pd
from sklearn.preprocessing import OneHotEncoder, LabelEncoder

df = pd.read_csv('Data.csv')

# OneHot encoding for Country
if 'Country' in df.columns:
    ohe = OneHotEncoder(sparse=False, handle_unknown='ignore')
    country_encoded = ohe.fit_transform(df[['Country']])
    country_cols = [f"Country_{cat}" for cat in ohe.categories_[0]]
    country_df = pd.DataFrame(country_encoded, columns=country_cols)
    df = pd.concat([df.reset_index(drop=True), country_df], axis=1)

# Label encoding for Purchased
if 'Purchased' in df.columns:
    le = LabelEncoder()
    df['Purchased_Label'] = le.fit_transform(df['Purchased'])

print(df.head())
```

Slip 7 - Q.2) Python Program

```
# Slip 7 - Q.2 Python Programs
import pandas as pd
from sklearn.preprocessing import StandardScaler

df = pd.read_csv('winequality-red.csv')

# Standardizing the data (mean=0, std=1)
scaler = StandardScaler()
scaled = scaler.fit_transform(df)
scaled_df = pd.DataFrame(scaled, columns=df.columns)
print(scaled_df.head())
```

Slip 8 - Q.2) Python Program

```
# Slip 8 - Q.2 Python Programs
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd

# A) Random array and plots (line & scatter)
data = np.random.randint(1,100,50)
plt.figure(figsize=(10,4))
plt.subplot(1,2,1); plt.plot(data); plt.title('Line Chart')
plt.subplot(1,2,2); plt.scatter(range(len(data)), data); plt.title('Scatter Plot')
plt.tight_layout(); plt.show()

# B) Pie chart from two lists
subjects = ['Math','CS','AI','DS','Python']
marks = [78,85,90,88,80]
plt.pie(marks, labels=subjects, autopct='%1.1f%%'); plt.title('Marks Pie Chart'); plt.show()

# C) Import winequality-red.csv and show basic info
df = pd.read_csv('winequality-red.csv')
print('Describe:\n', df.describe())
print('Shape:', df.shape)
print('First 3 rows:\n', df.head(3))
```

Slip 9 - Q.2) Python Program

```
# Slip 9 - Q.2 Python Programs
import pandas as pd
import numpy as np
from itertools import combinations

# A) Column-wise mean and median for SOCR-HeightWeight dataset
hw = pd.read_csv('SOCR-HeightWeight.csv')
print('Means:\n', hw.mean())
print('Medians:\n', hw.median())

# B) Sum of Manhattan distances between all pairs of points
points = hw.values
def manhattan_sum(arr):
    total = 0
    for (i,j) in combinations(range(len(arr)),2):
        total += np.sum(np.abs(arr[i]-arr[j]))
    return total

print('Sum of Manhattan distances:', manhattan_sum(points))
```

Slip 10 - Q.2) Python Program

```
# Slip 10 - Q.2 Python Programs
import pandas as pd
import matplotlib.pyplot as plt

df = pd.read_csv('iris.csv')

# A) Bar plot frequency of species
counts = df['species'].value_counts()
plt.bar(counts.index, counts.values); plt.title('Iris Species Frequency'); plt.show()

# B) Histogram of species (using sepal_length as example)
for sp in df['species'].unique():
    plt.hist(df[df['species']==sp]['sepal_length'], alpha=0.6, label=sp)
plt.legend(); plt.title('Histogram of Sepal Length by Species'); plt.show()
```

Slip 11 - Q.2) Python Program

```
# Slip 11 - Q.2 Python Programs
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt

# A) Random array of 50 integers and plots
data = np.random.randint(0,100,50)
plt.figure(figsize=(10,6))
plt.subplot(2,2,1); plt.plot(data); plt.title('Line Chart')
plt.subplot(2,2,2); plt.scatter(range(len(data)), data); plt.title('Scatter Plot')
plt.subplot(2,2,3); plt.hist(data, bins=10); plt.title('Histogram')
plt.subplot(2,2,4); plt.boxplot(data); plt.title('Box Plot')
plt.tight_layout(); plt.show()

# B) Create dataframe with missing and duplicate values then drop nulls
df = pd.DataFrame({
    'name':[f'Student{i}' for i in range(1,11)],
    'salary':[5000,6000,np.nan,7000,8000,6000,7000,np.nan,9000,5000],
    'department':['CS','IT','CS','AI','DS','CS','IT','DS','AI','CS']
})
df = pd.concat([df, df.iloc[:2]], ignore_index=True)  # add duplicates
print('Before drop:\n', df)
```

```
df_clean = df.dropna().drop_duplicates().reset_index(drop=True)
print('After drop:\n', df_clean)
```

Slip 12 - Q.2) Python Program

```
# Slip 12 - Q.2 Python Programs
import pandas as pd
import numpy as np

# A) Relationship between petal length and petal width (iris)
df = pd.read_csv('iris.csv')
import matplotlib.pyplot as plt
plt.scatter(df['petal_length'], df['petal_width'])
plt.xlabel('Petal Length'); plt.ylabel('Petal Width'); plt.title('Petal Length vs Petal Width')
plt.show()

# B) Max and min of a flattened array
arr = np.array([[3,5,7],[1,9,2]])
flat = arr.flatten()
print('Max:', flat.max(), 'Min:', flat.min())
```

Slip 13 - Q.2) Python Program

```
# Slip 13 - Q.2 Python Programs
import numpy as np
import pandas as pd

# A) Weighted average along specified axis of a flattened array
arr = np.arange(1,13).reshape(3,4)
weights = np.array([0.1,0.2,0.3,0.4])
# weighted average along axis=1
weighted_avg = np.average(arr, axis=1, weights=weights)
print('Weighted averages:', weighted_avg)

# B) Basic stats of advertising.csv
df = pd.read_csv('advertising.csv')
print(df.describe())
```

Slip 14 - Q.2) Python Program

```
# Slip 14 - Q.2 Python Programs
import numpy as np
import matplotlib.pyplot as plt

# A) Random array and plots
data = np.random.randint(1,100,50)
plt.figure(figsize=(10,6))
plt.subplot(2,2,1); plt.plot(data); plt.title('Line Chart')
plt.subplot(2,2,2); plt.scatter(range(len(data)), data); plt.title('Scatter Plot')
plt.subplot(2,2,3); plt.hist(data, bins=10); plt.title('Histogram')
plt.subplot(2,2,4); plt.boxplot(data); plt.title('Box Plot')
plt.tight_layout(); plt.show()

# B) Two lists pie chart
subjects = ['Math','CS','AI','DS','Python']; marks = [78,85,90,88,80]
plt.pie(marks, labels=subjects, autopct='%1.1f%%'); plt.title('Marks Pie Chart'); plt.show()
```

Slip 15 - Q.2) Python Program

```
# Slip 15 - Q.2 Python Programs
```

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

# A) Pie and bar chart from two lists
subjects = ['Math','CS','AI','DS','Python']
marks = [78,85,90,88,80]
plt.pie(marks, labels=subjects, autopct='%1.1f%%'); plt.title('Pie Chart'); plt.show()
plt.bar(subjects, marks); plt.title('Bar Chart'); plt.show()

# B) Dataframe students info and averages
df = pd.DataFrame({
    'name':['A','B','C','D','E'],
    'graduation_percentage':[75,88,92,68,80],
    'age':[21,22,23,20,24]
})
print('Average age:', df['age'].mean())
print('Average graduation %:', df['graduation_percentage'].mean())
```

Slip 16 - Q.2) Python Program

```
# Slip 16 - Q.2 Python Programs
import pandas as pd
import matplotlib.pyplot as plt

# A) Scatter plots to compare two features of iris dataset
df = pd.read_csv('iris.csv')
plt.scatter(df['sepal_length'], df['sepal_width']); plt.xlabel('Sepal Length'); plt.ylabel('Sepa
plt.scatter(df['petal_length'], df['petal_width']); plt.xlabel('Petal Length'); plt.ylabel('Peta

# B) Create dataframe with 10 rows and view
data = {'name':[f'P{i}' for i in range(1,11)], 'age':list(range(20,30)), 'salary':[3000+i*100 fo
df2 = pd.DataFrame(data)
print(df2)
```

Slip 17 - Q.2) Python Program

```
# Slip 17 - Q.2 Python Programs
import pandas as pd
import matplotlib.pyplot as plt

# A) Box plots for iris features across species
df = pd.read_csv('iris.csv')
features = ['sepal_length','sepal_width','petal_length','petal_width']
for feat in features:
    df.boxplot(column=feat, by='species'); plt.title(f'Boxplot of {feat} by species'); plt.supti

# B) Heights and weights dataset display
hw = pd.read_csv('SOCR-HeightWeight.csv')
print('First 5:\n', hw.head(5))
print('Last 5:\n', hw.tail(5))
print('Random 10:\n', hw.sample(10))
```

Slip 18 - Q.2) Python Program

```
# Slip 18 - Q.2 Python Programs
import pandas as pd
import numpy as np

# 1) Create dataframe and add 10 rows
data = {'name':[f'S{i}' for i in range(1,11)], 'age':list(range(18,28)), 'percentage':[60,70,80,
df = pd.DataFrame(data)
print(df)

# 2) Print shape, rows-columns, dtypes, feature names, description
print('Shape:', df.shape)
print('Rows:', df.shape[0], 'Columns:', df.shape[1])
print('Dtypes:\n', df.dtypes)
print('Feature names:', list(df.columns))
print('Description:\n', df.describe())

# 3) Add duplicates and missing values and add remarks column
df2 = df.append(pd.DataFrame([{'name':None,'age':None,'percentage':None}]), ignore_index=True)
df2 = pd.concat([df2, df2.iloc[:2]], ignore_index=True)
df2['remarks'] = ''
print(df2)
```