

Questions

PANDAS: pandas is a popular python library used for data manipulation and analysis. It provides data structures and functions necessary to perform tasks such as reading and writing data, data cleaning, data exploration, and data analysis. Pandas offers a wide range of functionalities for data manipulation such as selecting specific columns, filtering data, handling missing values, merging datasets and much more

^Importing the pandas library: Here's a basic example of how you can import pandas library

```
#Importing the pandas library
import pandas as pd
```

Checking the version of the pandas

```
#checking the version of the panda
print(pd.__version__)
```

1.5.3

Example: here's a simple example of creating a pandas data frame from a dictionary and performing some basic operations

```
#create a dictionary containing data
data={'name': [' Fatima ', ' Maryam ', ' Hauwa ', ' Aliyu ', ' Usman ', ' Aisha ', ' Zainab ',
              'age': [30, 25, 20, 45, 32, 15, 40, 53, 70, 65],
              'gender': [' female ', ' female ', ' female ', ' male ', ' male ', ' female ', ' female ']}
```

Create a DataFrame from the dictionary

```
#create a DataFrame from the dictionary
df = pd.DataFrame(data)
```

Display the DataFrame

```
#Display the DataFrame
print(df)
```

	name	age	gender
0	Fatima	30	female
1	Maryam	25	female
2	Hauwa	20	female
3	Aliyu	45	male
4	Usman	32	male
5	Aisha	15	female

6	Zainab	40	female
7	aminu	53	male
8	Abbas	70	male
9	Abdulrahman	65	male

Get basic information about the DataFrame

```
#Get basic information about the DataFrame
print(df.info())

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10 entries, 0 to 9
Data columns (total 3 columns):
#   Column  Non-Null Count  Dtype
---  -
0    name    10 non-null    object
1    age      10 non-null    int64
2    gender   10 non-null    object
dtypes: int64(1), object(2)
memory usage: 368.0+ bytes
None
```

Calculate descriptive statistics

```
#Calculate descriptive statistics
print(df.describe())
```

	age
count	10.00000
mean	39.50000
std	18.65029
min	15.00000
25%	26.25000
50%	36.00000
75%	51.00000
max	70.00000

Filter rows based on a condition

```
#Filter rows based on a condition
#this line of code will allow us to filter the column of age that are greater than or equal to 30
filtered_df = df[df['age'] >= 30]
print(filtered_df)
```

	name	age	gender
0	Fatima	30	female
3	Aliyu	45	male
4	Usman	32	male
6	Zainab	40	female
7	aminu	53	male
8	Abbas	70	male
9	Abdulrahman	65	male

```
#Filter rows based on a condition
```

```
#this line of code will allow us to filter the column of age that are less than 30
filtered_df = df[df['age'] < 30]
print(filtered_df)
```

	name	age	gender
1	Maryam	25	female
2	Hauwa	20	female
5	Aisha	15	female

data cleaning in pandas

```
import pandas as pd

# define a dictionary with sample data which includes some missing values
data = {
    'X': [9,12, 4,None, 15],
    'Y': [10, 12, 13, 6, 15],
    'Z': [7, 12, None, None, 15]
}

df = pd.DataFrame(data)
print("Original Data:\n",df)
print()

# use dropna() to remove rows with any missing values
df_cleaned = df.dropna()

print("Cleaned Data:\n",df_cleaned)
```

Original Data:

	X	Y	Z
0	9.0	10	7.0
1	12.0	12	12.0
2	4.0	13	NaN
3	NaN	6	NaN
4	15.0	15	15.0

Cleaned Data:

	X	Y	Z
0	9.0	10	7.0
1	12.0	12	12.0
4	15.0	15	15.0

```
import pandas as pd

# sample data
data = {
    'X': [1, 3, 3, 6, 8, 8],
    'Y': [12, 4, 12, 7, 9, 5],
    'Z': [10, 5, 6, 17, 14, 2]
}
df = pd.DataFrame(data)

print("Original DataFrame:\n", df.to_string(index=False))

# detect duplicates
print("\nDuplicate Rows:\n", df[df.duplicated()].to_string(index=False))
```

```
# remove duplicates based on column 'X'
df.drop_duplicates(subset=['X'], keep='first', inplace=True)

print("\nDataFrame after removing duplicates based on column 'X':\n", df.to_string(index=False))
```

Original DataFrame:

	X	Y	Z
1	12	10	
3	4	5	
3	12	6	
6	7	17	
8	9	14	
8	5	2	

Duplicate Rows:

Empty DataFrame

Columns: [X, Y, Z]

Index: []

DataFrame after removing duplicates based on column 'X':

	X	Y	Z
1	12	10	
3	4	5	
6	7	17	
8	9	14	

```
import pandas as pd
```

```
# sample data
```

```
data = {
    'X': [25, 20, 32],
    'Y': ['fatima', 'Aisha', 'Ahmad'],
    'Z': [20000, 40000, 70000]
}
```

```
df = pd.DataFrame(data)
```

```
# rename columns
```

```
df.rename(columns={'X': 'Age', 'Y': 'Name', 'Z': 'pension'}, inplace=True)
```

```
print(df.to_string(index=False))
```

Age	Name	pension
25	fatima	20000
20	Aisha	40000
32	Ahmad	70000

