

Practical-3

1. Creating DataFrames (Multiple Methods)

```
import pandas as pd # From Dictionary
data = {
    'ID': [1, 2, 3, 4],
    'Name': ['Alice', 'Bob', 'Charlie', 'David'],
    'Score': [78, 85, 90, 88]
}
df = pd.DataFrame(data)
print(df)

# From List of Lists
data2 = [[101, 'Math', 85], [102, 'Science', 90]]
df2 = pd.DataFrame(data2, columns=['Roll', 'Subject', 'Marks'])
print(df2)
```

2. Inspecting and Understanding Data

```
print(df.head())
print(df.tail())
print(df.shape)
print(df.info())
print(df.describe())
```

3. Indexing & Selecting Data (loc, iloc, at, iat)

```
# Using loc
print(df.loc[1])

# Using iloc
print(df.iloc[0:2])

# Access single value
print(df.at[0, 'Name'])
print(df.iat[2, 2])
```

4. Row and Column Operations

```
# Add column
df['Bonus'] = 5
df['Final_Score'] = df['Score'] + df['Bonus']
print(df)

# Drop row and column
df.drop(3, axis=0, inplace=True)
df.drop('Bonus', axis=1, inplace=True)
print(df)
```

5. Filtering and Conditional Selection

```
# Filter data
high_scores = df[df['Final_Score'] > 85]
print(high_scores)

# Multiple conditions
filtered = df[(df['Final_Score'] > 80) & (df['ID'] > 1)]
print(filtered)
```

6. Sorting and Ranking

```
# Sort values
print(df.sort_values(by='Final_Score', ascending=False))

# Rank
df['Rank'] = df['Final_Score'].rank(ascending=False)
print(df)
```

7. Working with Missing Data

```
import numpy as np
```

```
df.loc[1, 'Final_Score'] = np.nan  
  
print(df.isnull()) print(df.fillna(df['Final_Score'].mean())) print(df.dropna())
```

8. Renaming Columns and Index

```
df.rename(columns={'Final_Score': 'Total'}, inplace=True) df.index = ['A', 'B', 'C']  
print(df)
```

9. String Operations

```
df['Name_Upper'] = df['Name'].str.upper()  
df['Name_Length'] = df['Name'].str.len() print(df)
```

10. GroupBy Operations

```
# Grouping example data3 = {  
    'Dept': ['IT', 'IT', 'HR', 'HR'],  
    'Employee': ['A', 'B', 'C', 'D'],  
    'Salary': [50000, 60000, 45000, 48000]  
}
```

```
emp_df = pd.DataFrame(data3) print(emp_df.groupby('Dept')['Salary'].mean())
```

11. Applying Functions

```
# Apply function  
df['Grade'] = df['Total'].apply(lambda x: 'A' if x >= 90 else 'B') print(df)
```

12. Exporting Data

```
# Save to CSV df.to_csv('output.csv', index=False)  
print("Data exported successfully")
```