2-Hour Class Plan: Pandas for Data Analytics

Target Audience

Beginners in Python / Data Analytics students

Total Duration: 2 Hours (120 Minutes)

Session 1: Introduction to Pandas (0 – 15 mins)

What is Pandas?

- Pandas = Python Data Analysis Library
- Built on top of NumPy
- Used for data cleaning, manipulation, and analysis
- Works like Excel but more powerful

Why Pandas is Important in Data Analytics

- Handles structured data easily
- Fast and efficient
- Widely used in industry

Import Pandas

```
import pandas as pd
```

Session 2: Pandas Data Structures (15 – 35 mins)

Series (1D Data)

```
marks = pd.Series([65, 70, 80, 90, 55])
print(marks)
```

DataFrame (2D Data – Table Format)

```
data = {
    'Name': ['A', 'B', 'C', 'D', 'E'],
    'Marks': [65, 70, 80, 90, 55]
```

```
}

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

Session 3: Reading & Writing Data (35 – 55 mins)

Read CSV File

```
df = pd.read_csv('students.csv')
```

Read Excel File

```
df = pd.read_excel('students.xlsx')
```

Write Data to CSV

```
df.to_csv('output.csv', index=False)
```

Session 4: Data Exploration (55 – 75 mins)

Basic Data Inspection

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

Shape and Columns

```
df.shape
df.columns
```

Session 5: Data Selection & Filtering (75 – 95 mins)

Select Column

```
df['Marks']
```

Filter Data

```
df[df['Marks'] > 60]
```

Multiple Conditions

```
df[(df['Marks'] > 60) & (df['Marks'] < 90)]
```

Session 6: Data Cleaning (95 – 110 mins)

Handling Missing Values

```
df.isnull()
df.dropna()
df.fillna(0)
```

Rename Columns

```
df.rename(columns={'Marks': 'Score'}, inplace=True)
```

Session 7: Real-Time Case Study (110 – 120 mins)

Real-World CSV-Based Case Study: Student Performance Analysis

Scenario

An institute has collected student performance data in a CSV file. The academic team wants to analyze: - Average marks - Top performers - Pass percentage - Students needing improvement

Sample CSV File: `students.csv`

```
Student_ID,Name,Department,Marks,Attendance
101,Asha,CS,78,85
102,Ravi,EC,65,72
103,Neha,CS,90,95
104,Arjun,ME,55,60
105,Divya,EC,88,90
```

Step 1: Load the CSV file

```
df = pd.read_csv('students.csv')
print(df)
```

Step 2: Basic Exploration

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

Step 3: Analysis Tasks

```
# Average marks
avg_marks = df['Marks'].mean()

# Highest marks
top_score = df['Marks'].max()

# Passed students (Marks >= 60)
passed_students = df[df['Marks'] >= 60]

# Students needing improvement
failed_students = df[df['Marks'] < 60]
```

Step 4: Attendance-based filtering

```
low_attendance = df[df['Attendance'] < 75]
```

Step 5: Summary Output

```
print('Average Marks:', avg_marks)
print('Top Score:', top_score)
print('Passed Students:
', passed_students)
print('Students with Low Attendance:
', low_attendance)
```

---

python average_marks = df['Score'].mean() highest_marks = df['Score'].max() passed_students = df[df['Score'] >= 60] ```

Benefits of Pandas

- Easy handling of large datasets
- Powerful data cleaning tools
- Excel-like operations
- Integrates with NumPy, Matplotlib, Seaborn

Assignment / Practice Tasks

1. Load a CSV file and display first 10 rows
2. Find average marks per subject
3. Filter students who scored above average

Learning Outcomes

By the end of this class, students will: - Understand Pandas basics - Work with Series and DataFrames - Perform data cleaning and analysis

---

Tools Required

- Python
- Pandas
- Jupyter Notebook / VS Code