**Lab Exercise 6- Implementing Data Transformations from CSV Using Pandas in Metaflow**

**Objective:**

Learn how to create a Metaflow pipeline that reads data from a CSV file, performs data transformations using pandas, and outputs the results.

**Step 1: Set Up Your Environment**

Make sure you have Metaflow and pandas installed. If not, you can install them using pip:

pip install metaflow pandas

**Step 2: Prepare Your CSV File**

Create a CSV file named data.csv with the following content:

id,value

1,10

2,20

3,NaN

4,30

5,15

**Step 3: Create the Data Transformation Flow**

Create a Python file named CSVDataTransformationFlow.py and add the following code:

import pandas as pd

from metaflow import FlowSpec, step

class CSVDataTransformationFlow(FlowSpec):

@step

def start(self):

"""

Start step: Load data from CSV.

"""

self.raw\_data = pd.read\_csv('data.csv')

print("Raw data:")

print(self.raw\_data)

self.next(self.clean\_data)

@step

def clean\_data(self):

"""

Clean the data by removing entries with NaN values.

"""

self.cleaned\_data = self.raw\_data.dropna().reset\_index(drop=True)

print("Cleaned data:")

print(self.cleaned\_data)

self.next(self.feature\_engineering)

@step

def feature\_engineering(self):

"""

Create new features based on the cleaned data.

"""

self.cleaned\_data['double\_value'] = self.cleaned\_data['value'] \* 2

self.cleaned\_data['is\_even'] = self.cleaned\_data['double\_value'] % 2 == 0

print("Features created:")

print(self.cleaned\_data)

self.next(self.end)

@step

def end(self):

"""

End step: Final output.

"""

print("Data transformation flow completed.")

print("Final features:")

print(self.cleaned\_data)

if \_\_name\_\_ == "\_\_main\_\_":

CSVDataTransformationFlow()

**Explanation of the Flow:**

* **start step**: Loads the data from the data.csv file using pandas and prints the raw data.
* **clean\_data step**: Cleans the data by removing entries with NaN values.
* **feature\_engineering step**: Creates new features, such as a double value and a boolean indicating whether the doubled value is even.
* **end step**: Finalizes the flow and prints the transformed data.

**Step 4: Run the Flow**

Run the flow using the following command:

python CSVDataTransformationFlow.py run

You should see output for each step, displaying the raw data, cleaned data, and final features.

**Conclusion**

In this exercise, you learned how to:

* Create a Metaflow pipeline that reads data from a CSV file using pandas.
* Perform data cleaning and feature engineering.
* Modify the flow to include additional transformations.