**Lab Exercise 3- Handling Artifacts and Parameters in Metaflow**

**Objective**:

* Learn to define and use parameters in a Metaflow flow.
* Learn to use artifacts to pass data between steps.

**ArtifactsAndParamsFlow.py**

from metaflow import FlowSpec, step, Parameter

class ArtifactsAndParamsFlow(FlowSpec):

# Adding parameters

input\_text = Parameter("input\_text", default="Hello, Metaflow!", help="Input text for the flow.")

multiplier = Parameter("multiplier", default=2, type=int, help="Multiplier for processing the input.")

@step

def start(self):

"""

Start step. Initializes the flow and prints the input parameters.

"""

print(f"Input Text: {self.input\_text}")

print(f"Multiplier: {self.multiplier}")

self.next(self.handle\_artifacts)

@step

def handle\_artifacts(self):

"""

This step processes the input parameters and stores results as artifacts.

"""

self.processed\_text = self.input\_text \* self.multiplier # Using the input parameters

print(f"Processed Text: {self.processed\_text}")

self.next(self.display\_results)

@step

def display\_results(self):

"""

This step displays the final results using the artifacts from previous steps.

"""

print(f"Final Processed Text: {self.processed\_text}")

self.next(self.end)

@step

def end(self):

"""

The end step marks the completion of the flow.

"""

print("Flow completed.")

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

ArtifactsAndParamsFlow()

**Step 5: Running the Flow**

Run the flow using the command below:

python ArtifactsAndParamsFlow.py run

You can override the parameters like this:

python ArtifactsAndParamsFlow.py run --input\_text "Metaflow" --multiplier 5

**Explanation Summary:**

1. **Parameters** are defined at the class level and allow users to pass values to the flow at runtime.
2. **Artifacts** are created by assigning values to self in one step and can be accessed by subsequent steps.
3. Each step in the flow can access both the **parameters** and **artifacts** defined in previous steps.

This exercise demonstrates the basics of handling parameters and artifacts in Metaflow, which helps in building reproducible and maintainable workflows.