**Assignment Documentation**

**1. Overview**

This document provides a detailed explanation of the Python script (Assignment.py). The script is designed to process data by applying transformations, handling errors, and logging execution details. It ensures data consistency, tracks errors, and maintains smooth execution by logging every step efficiently.

**2. Prerequisites**

Before running the script, ensure the following requirements are met:

* **Python 3.x Installed:** Verify by running the command:
* python --version
* **Required Python Libraries:** Install dependencies using the command:
* pip install -r requirements.txt

**3. Installation & Execution**

**Step-by-Step Execution Guide**

Follow the steps below to run the script:

1. **Ensure Python is installed** on your system.
2. **Install required dependencies** using the command mentioned in Section 2.
3. **Navigate to the script directory** in the terminal or command prompt.
4. **Run the script** using:
5. python Assignment.py
6. **Monitor the output and logs** for any errors or status updates.

**4. Approach & Logic**

The script follows a structured approach:

1. **Data Input Handling:** Reads input data (structured correctly as a list of numbers).
2. **Data Processing:** Applies a transformation (multiplication by 2) to all elements.
3. **Logging Execution:** Captures the process status, including warnings and errors.
4. **Error Handling:** Detects and logs potential issues for debugging.
5. **Output Generation:** Displays and logs the processed data.

**5. Assumptions**

The following assumptions were made while developing the script:

* Input data is structured correctly (e.g., a list of numbers).
* The operation (value \* 2) applies to all elements.
* The script runs in a Python 3.x environment.
* The required dependencies are installed before execution.

**6. Example Output**

When executed successfully, the script produces an output similar to the following:

Processing data: [10, 20, 30]

Transformed Data: [20, 40, 60]

INFO: Process completed successfully!

The transformed data is displayed on the console and logged for future reference.

**7. Logging Details**

**Logging Mechanism in the Script**

The script uses logging to capture execution details and errors. The logs are categorized as follows:

* **INFO:** General execution details and successful operations.
* **WARNING:** Potential issues that do not stop execution.
* **ERROR:** Critical issues that halt execution.

Logs are saved in a file named logs.txt for debugging and future analysis.

**8. Contact & Support**

For any issues or support, feel free to reach out:

📧 **Email:** [akanksha.mishra2894@gmail.com](mailto:akanksha.mishra2894@gmail.com)

**End of Document**