Assignment 1: Object-Oriented Data Cleaning and Preprocessing

Deadline: Friday, May 30th 2025

Objective

In this assignment, you will:

- Implement preprocessing methods in a provided Python script (`data_preprocessor.py`) to clean and preprocess a messy dataset.
- Import the completed script into a Jupyter notebook to test your preprocessing methods and evaluate the impact on model performance.
- Use your GitHub repository to organize and submit your work.
- Answer reflection questions to demonstrate your understanding of the preprocessing pipeline and its implications.

Assignment Instructions

Part 1: Set Up Your GitHub Repository

1. Create a **public** GitHub repository with the following structure:

```
YourClassRepositoryName/YourAssignmentFolderName

├── README.md

├── Scripts         ← Python files

├── Data            ← Spreadsheets
```

2. Include the Following:

- `Data/messy_data.csv`: The provided messy dataset for preprocessing.
- `Scripts/data_preprocessor.py`: Your implementation of the `DataPreprocessor` script.
- `Scripts/main.ipynb`: A notebook to test your preprocessing methods, apply it to `messy_data.csv`, and generate `cleaned_data.csv`.
- `README.md`: Brief documentation describing your project and instructions for running your code.

Part 2: Complete the methods in `data_preprocessor.py`

1. **Review the Skeleton Code:**
   Open the provided template (`data_Preprocessor.py`) and examine the placeholder methods. Understand each method's purpose based on the comments.

2. **Fill in the Methods:**

   Use the descriptions to implement each method.

3. **Test Each Method:**
   Create a copy of the data to test and verify the output of each method before moving to the next.

4. **Document Your Code:**
   Add comments explaining any new logic you implement, ensuring readability for your peers or future reference.

Part 3: Dataset for Preprocessing

Load the Dataset:
Download `messy_data.csv` from Blackboard and load into your workspace in `main.ipynb`.

1. **Examine the Dataset:**
   - Display the first few rows using `.head()` to understand the structure.
   - Use `.info()` and `.describe()` to check data types, missing values, and basic statistics.
2. **Identify Issues:**
   - Look for missing values, redundant columns, outliers, inconsistent formatting, or categorical features that need encoding.
   - Use visualizations (e.g., histograms or scatter plots) to identify potential outliers.
3. **Apply the `DataPreprocessor` Class:**
   - Apply each method step-by-step and print outputs to monitor changes.
4. **Save the Cleaned Data:**
   After preprocessing, save the cleaned dataset for analysis.

Part 4: Short-Answer Questions (Separate file)

Assignment 1: Object-Oriented Data Cleaning and Preprocessing

Deadline: Friday, May 30th 2025

Deliverables

Submission Component	**Requirement**
GitHub Repository	● Your implementation of the `data_preprocessor.py` script; ● Your implementation of the `main.ipynb` notebook; ● Final, cleaned dataset as `cleaned_data.csv`.
Short-Answer Responses	Submit your answers to the short-answer questions (Part 4) to Blackboard in a .pdf format.

Grading Criteria

Assessment Criteria	Weight	Description
Code Functionality	40%	Completeness and correctness of the `data_preprocessor` and `main` implementation.
Cleaned Dataset	10%	Appropriateness and quality of the cleaned dataset produced after applying the class methods.
Short-Answer Responses	40%	Depth, clarity, and correctness of written answers to provided questions or prompts.
Code Quality	10%	Readability, logical structure, and effective use of comments to document the code.