**Big Data Assignment 1**

**Description:**

In this assignment, we build a Docker container environment to process and analyze a dataset. We perform various data processing tasks, including cleaning, transformation, exploratory data analysis (EDA), visualization, and clustering using K-means. The tasks will be done using Python scripts inside the Docker container, and the results will be exported to our local machine.

**How to run the project:**

First, Open docker desktop, then initialize the image that you will build your project on:

**docker build -t bd-a1-image**.

Then run the container:

**docker run -it --name bd-a1-container bd-a1-image**

After that, you will run the python files that you have on the container using this command:

**python3 load.py train\_corrupted.csv**

**./final.sh**

**User Guide:**

Step 1: Choose Dataset

We choose a dataset “Train\_Corrupted” dataset

Step 2: Identify ubuntu

FROM ubuntu:22.04

Step 3: Create Container

We created a container that has the name **doc-bd-a1**

Step 3: initialize libraries

We installed the packages required for the project (Python3, Pandas, NumPy, Seaborn, Matplotlib, scikit-learn, and Scipy)

Step 4: add the dataset to the container

COPY train\_corrupted.csv /home/doc-bd-a1/

Step 5: Open the Bash

CMD ["/bin/bash"]

Step 6: Creating Python files

We created 6 python files:

1) Load.py: for loading the dataset

2) dpre.py: pre-processing on the dataset (“Remove nulls”,” Normalized data”)

3)eda.py:

1- Insight 1: Average battery power increases with price range: 0: 1188.6134989798059, 1: 1257.7321754664551, 2: 1315.566774527976, 3: 1454.0181123461691}

2- Insight 2: 51.02% of devices support 4G.

3- Insight 3: Correlation between RAM and price range is 0.85

4)vis.py:

A graph of blue dots

AI-generated content may be incorrect.

5) model.py: we created a model using K-means with 3 clusters and the output is:

Example: (Cores, Battery Power, Ram, Clock speed, Mobile wt)

Cluster 0: 496 records

Cluster 1: 151 records

Cluster 2: 478 records

6)final.sh : we combined the python files, and we loaded the container onto the local machine

python3 load.py train\_corrupted.csv

./final.sh