Assignment 2 Report – Big Data Infrastructure (VirtualBox + MongoDB)

Student: Yanelia Naddaf

Course: Big Data Structure

Submission Date: May 3rd

**PART 1: Setting up the Virtual Machine (VM)**

For this part, I followed the provided steps to create a Virtual Machine using VirtualBox. I selected Ubuntu as the operating system and adjusted the hardware configuration to 2 GB of RAM and 1 CPU. During the setup:

* I downloaded the Ubuntu 25.04 ISO file from the official website.
* I configured a new VM named MongoDB\_VM in VirtualBox.
* Set storage to 25GB and selected the Ubuntu (64-bit) type.

Challenge Encountered:

Despite multiple attempts (screenshots attached), I was not able to complete the installation of the operating system inside the VM. I encountered issues with the EFI boot sequence and disk recognition. I did my best and documented each step with screenshots to demonstrate my progress and efforts.

**PART 2: Downloading the Dataset**

* I downloaded the dataset named amazon\_sales\_data\_2025.csv from Kaggle.
* The file was successfully uploaded to the working directory within Jupyter Notebook under the folder:

/OneDrive/DATA\_SCIENTIST\_DIG\_DATA\_ANALYST/Concordia\_Data\_Science\_Programming/5\_Big\_Data\_Estructure/Assignment\_2\_VirtualBox/

**PART 3: Converting CSV to JSON**

Since the VM wasn’t accessible for processing the file, I implemented and tested this part using Jupyter Notebook locally.

Script Created:

I wrote a script named csv\_to\_json.py that reads a CSV file and converts it to a JSON file using Python’s csv and json libraries.

Code Summary:

import csv

import json

def csv\_to\_json():

csv\_file\_path = "amazon\_sales\_data\_2025.csv"

json\_file\_path = "amazon\_sales\_data\_2025.json"

data = []

try:

with open(csv\_file\_path, encoding='utf-8') as csvfile:

reader = csv.DictReader(csvfile)

for row in reader:

data.append(row)

with open(json\_file\_path, 'w', encoding='utf-8') as jsonfile:

json.dump(data, jsonfile, indent=4)

print(f"Successfully converted {len(data)} rows from CSV to JSON.")

except Exception as e:

print(f" Error: {e}")

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

csv\_to\_json()

The script executed successfully and created a .json file.

Additional Work:

As I could not import the dataset into MongoDB via the VM, I performed analysis within Jupyter Notebook using pandas.

Most Sold Item:  
I used pandas to calculate which product had the highest quantity sold. The result is:

Product Quantity

7 Smartwatch 105

This is the code applied

import pandas as pd

# Step 1: Load the dataset (CSV)

df = pd.read\_csv("amazon\_sales\_data\_2025.csv")

# Step 2: Group by product\_name and sum the quantity

sales\_summary = df.groupby("Product")["Quantity"].sum().reset\_index()

# Step 3: Sort the results in descending order

sorted\_sales = sales\_summary.sort\_values(by="Quantity", ascending=False)

# Step 4: Get the top-selling product

most\_sold\_item = sorted\_sales.head(1)

# Display the result

print("Most Sold Item:")

print(most\_sold\_item)

This analysis was done in Jupyter Notebook since the MongoDB query could not be executed inside the VM due to installation issues. In a typical setup, this analysis would be done using a MongoDB query within the VM.

**PART 4: Importing Data into MongoDB (Not Completed Due to VM Issue)**

This part was planned to be done within the VM using mongoimport. However, since the Ubuntu installation could not be completed, I could not run MongoDB.

Planned command:

mongoimport --db amazon --collection sales --file amazon\_sales\_data\_2025.json --jsonArray

Final Note:

Despite the issues with VirtualBox, I made substantial progress and was able to complete Parts 2 and 3, including scripting and basic data analysis. I included screenshots of all steps attempted to install Ubuntu to demonstrate my effort and understanding.

Thank you for your understanding.

* Ubuntu VM installation attempts with screenshots.
* Completion of CSV to JSON conversion using Python in Jupyter.
* The result of the analysis: Smartwatch was the most sold item with 105 units — calculated in Jupyter due to VM limitations.