Hands-On Lab: Generative AI for Data Preparation

Estimated time needed: 30 minutes

Overview

In this lab, you will learn how to use generative AI to prepare data using the tool, chatcsv.

Objectives

After completing this lab, you will be able to:

1. Sign in on https://www.chatcsv.co/
2. Upload a dataset
3. Handle missing values
4. Perform the data standardization
5. Perform the data normalization

Prerequisites:

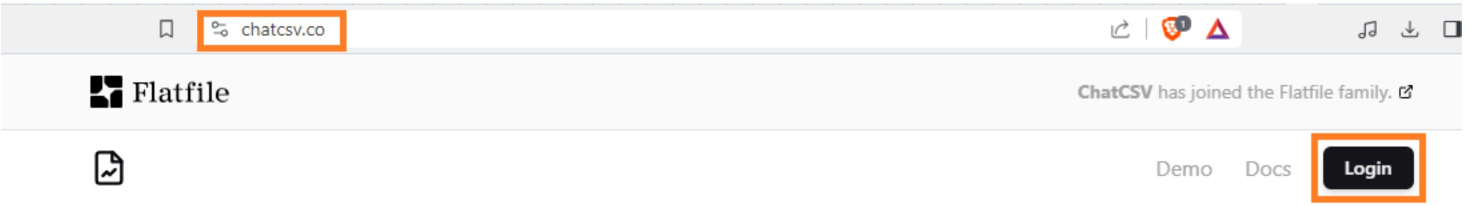
- A chatcsv.co account
- A basic understanding of EDA

Dataset

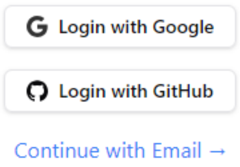
The dataset is a filtered and modified version of the Laptop Price Prediction using specifications dataset, available under the Database Contents License (DbCL) v1.0 on the Kaggle website. While holding down the Ctrl or Command button, click here to download the data set.

Task 1: Sign in on Chatcsv.co

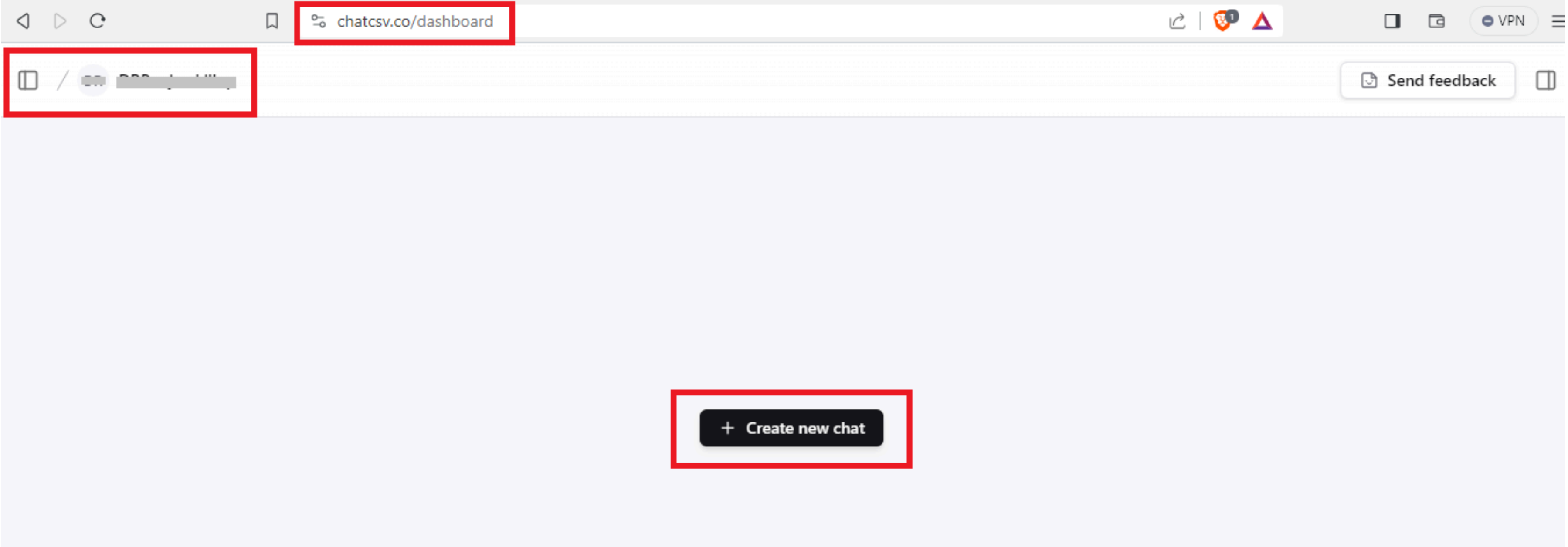
Step 1: If you do not have an account, click https://www.chatcsv.co/ and then click **Login** at the top right corner.



Step 2: You can login using gmail, Github, or your email ID. Click any one option and follow the steps to create your account on Chatcsvco.

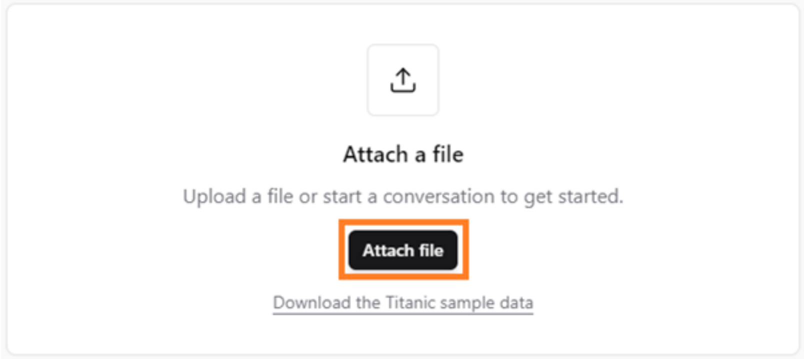


Step 3: After you create your account, login with your credentials and the chatcsv.co dashboard will be displayed.

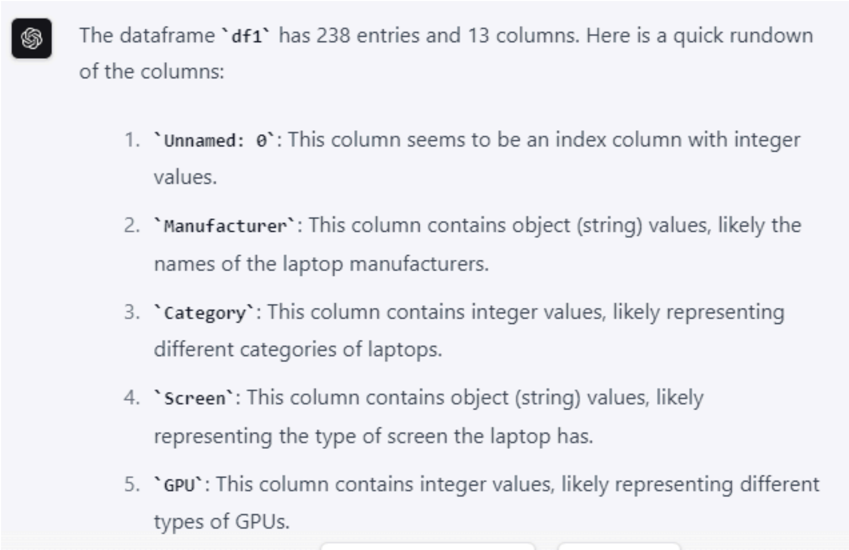


Task 2: Upload Dataset

Step 1: On the dashboard screen, click **Create new Chat** to start preparing data.
Step 2: Click **Attach file** and attach the dataset 'laptop_pricing_dataset_mod1.csv' from the location where you have downloaded the dataset from the link provided earlier.


Attach a file
Upload a file or start a conversation to get started.
Attach file
Download the Titanic sample data

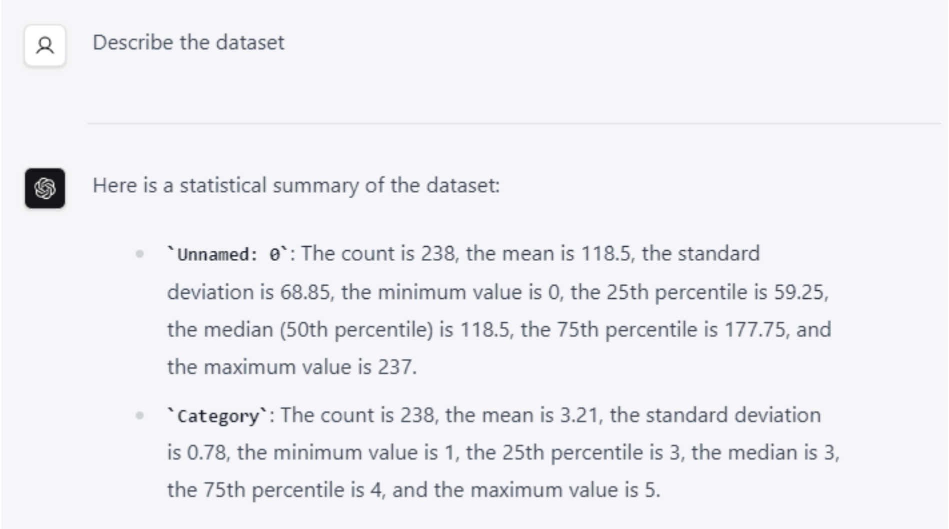
Step 3: Scroll down the dataset to view the details presented by the GPT.



The dataframe `df1` has 238 entries and 13 columns. Here is a quick rundown of the columns:

1. `Unnamed: 0`: This column seems to be an index column with integer values.
2. `Manufacturer`: This column contains object (string) values, likely the names of the laptop manufacturers.
3. `Category`: This column contains integer values, likely representing different categories of laptops.
4. `Screen`: This column contains object (string) values, likely representing the type of screen the laptop has.
5. `GPU`: This column contains integer values, likely representing different types of GPUs.

Step 4: Write a prompt to get the statistical description of the dataset:
"Describe the dataset"



Describe the dataset

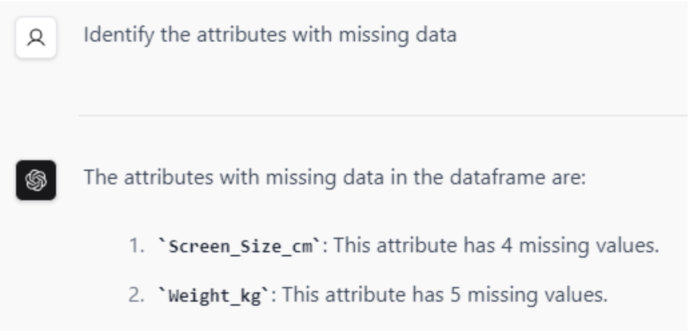
Here is a statistical summary of the dataset:

- `Unnamed: 0`: The count is 238, the mean is 118.5, the standard deviation is 68.85, the minimum value is 0, the 25th percentile is 59.25, the median (50th percentile) is 118.5, the 75th percentile is 177.75, and the maximum value is 237.
- `Category`: The count is 238, the mean is 3.21, the standard deviation is 0.78, the minimum value is 1, the 25th percentile is 3, the median is 3, the 75th percentile is 4, and the maximum value is 5.

Task 3: Handle missing values

Step 1: Write a prompt "Identify the attributes with missing data" and press Enter.
The response will display two attributes with missing values as "Screen_Size_cm", a categorical variable, and "Weight_kg", a continuous variable.
You need to replace the missing values with appropriate values. The following are the rules for this:

- Missing entries in columns containing categorical values need to be replaced with the most frequent entries.
- Missing entries in columns with continuous data need to be replaced with the mean value of the column. If a value is missing in the target column, you may need to drop that row.
  The prompt response will be something as shown below:

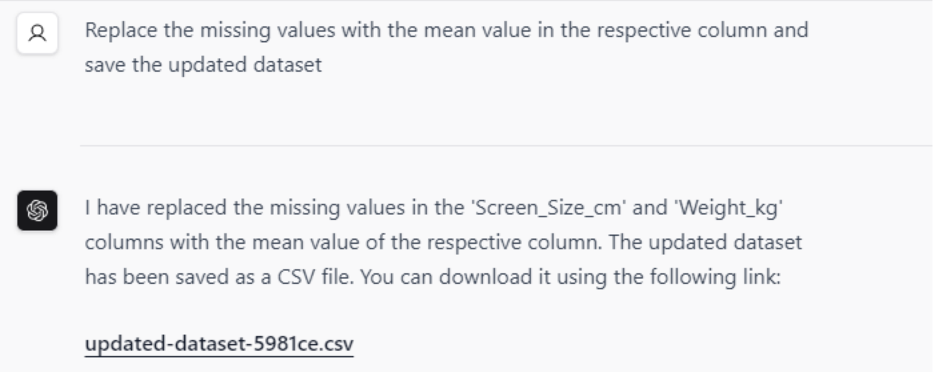


Identify the attributes with missing data

The attributes with missing data in the dataframe are:

1. `` `Screen_Size_cm` ``: This attribute has 4 missing values.
2. `` `Weight_kg` ``: This attribute has 5 missing values.

Step 2: Write a prompt: "Replace the missing values with the mean value in the respective column and save the updated dataset."

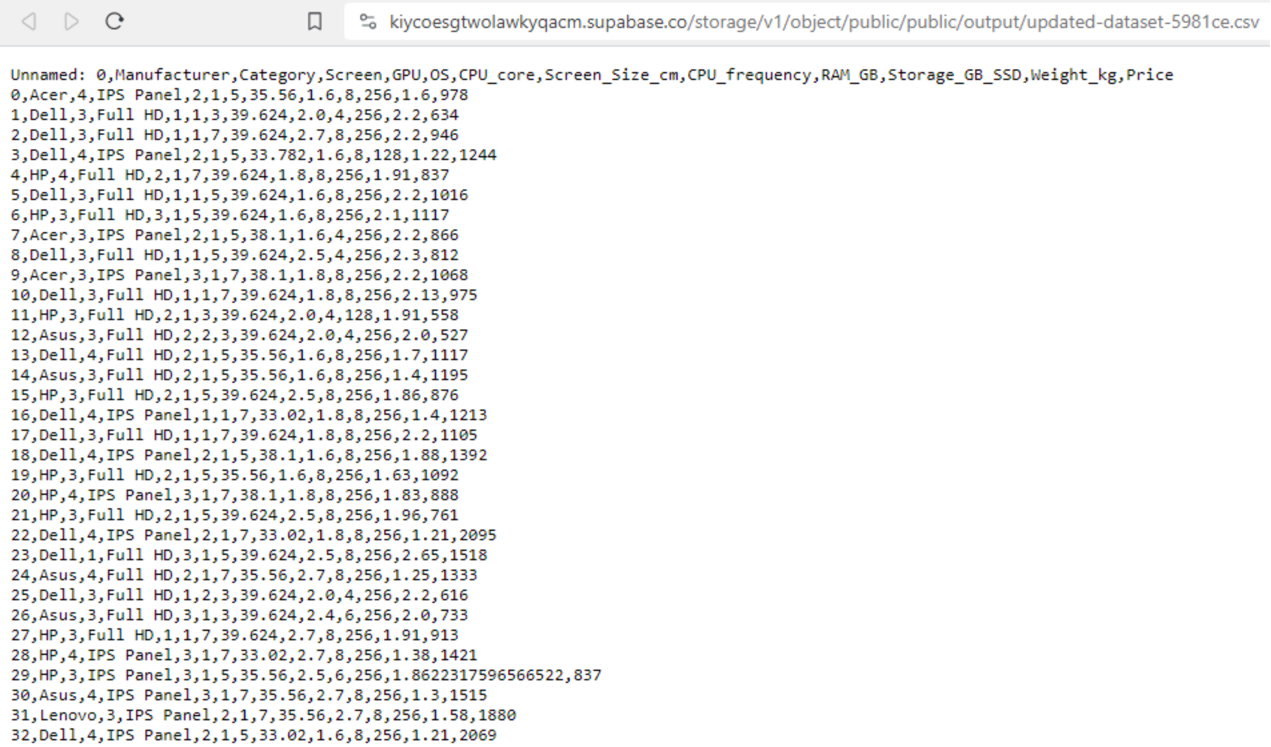


Replace the missing values with the mean value in the respective column and save the updated dataset

I have replaced the missing values in the 'Screen_Size_cm' and 'Weight_kg' columns with the mean value of the respective column. The updated dataset has been saved as a CSV file. You can download it using the following link:

updated-dataset-5981ce.csv

If you open the generated link for the updated dataset, it will be shown as below:



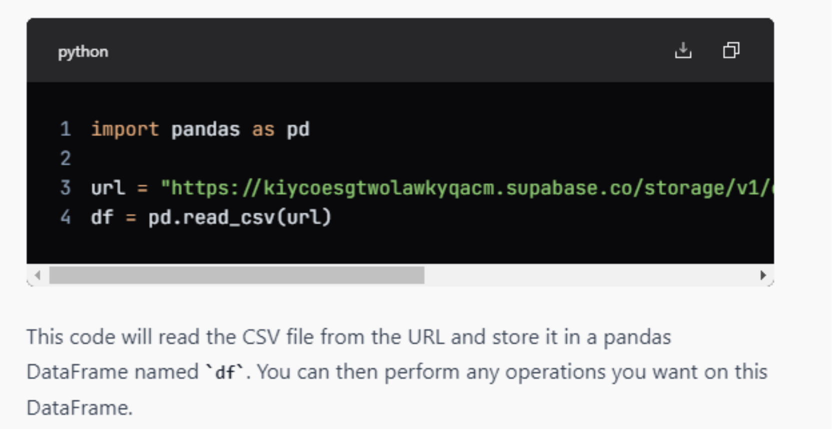
kiycoesgtwolawkyqacm.supabase.co/storage/v1/object/public/public/output/updated-dataset-5981ce.csv

```
Unnamed: 0,Manufacturer,Category,Screen,GPU,OS,CPU_core,Screen_Size_cm,CPU_frequency,RAM_GB,Storage_GB_SSD,Weight_kg,Price
0,Acer,4,IPS Panel,2,1,5,35.56,1.6,8,256,1.6,978
1,Dell,3,Full HD,1,1,3,39.624,2.0,4,256,2.2,634
2,Dell,3,Full HD,1,1,7,39.624,2.7,8,256,2.2,946
3,Dell,4,IPS Panel,2,1,5,33.782,1.6,8,128,1.22,1244
4,HP,4,Full HD,2,1,7,39.624,1.8,8,256,1.91,837
5,Dell,3,Full HD,1,1,5,39.624,1.6,8,256,2.2,1016
6,HP,3,Full HD,3,1,5,39.624,1.6,8,256,2.1,1117
7,Acer,3,IPS Panel,2,1,5,38.1,1.6,4,256,2.2,866
8,Dell,3,Full HD,1,1,5,39.624,2.5,4,256,2.3,812
9,Acer,3,IPS Panel,3,1,7,38.1,1.8,8,256,2.2,1068
10,Dell,3,Full HD,1,1,7,39.624,1.8,8,256,2.13,975
11,HP,3,Full HD,2,1,3,39.624,2.0,4,128,1.91,558
12,Asus,3,Full HD,2,2,3,39.624,2.0,4,256,2.0,527
13,Dell,4,Full HD,2,1,5,35.56,1.6,8,256,1.7,1117
14,Asus,3,Full HD,2,1,5,35.56,1.6,8,256,1.4,1195
15,HP,3,Full HD,2,1,5,39.624,2.5,8,256,1.86,876
16,Dell,4,IPS Panel,1,1,7,33.02,1.8,8,256,1.4,1213
17,Dell,3,Full HD,1,1,7,39.624,1.8,8,256,2.2,1105
18,Dell,4,IPS Panel,2,1,5,38.1,1.6,8,256,1.88,1392
19,HP,3,Full HD,2,1,5,35.56,1.6,8,256,1.63,1092
20,HP,4,IPS Panel,3,1,7,38.1,1.8,8,256,1.83,888
21,HP,3,Full HD,2,1,5,39.624,2.5,8,256,1.96,761
22,Dell,4,IPS Panel,2,1,7,33.02,1.8,8,256,1.21,2095
23,Dell,1,Full HD,3,1,5,39.624,2.5,8,256,2.65,1518
24,Asus,4,Full HD,2,1,7,35.56,2.7,8,256,1.25,1333
25,Dell,3,Full HD,1,2,3,39.624,2.0,4,256,2.2,616
26,Asus,3,Full HD,3,1,3,39.624,2.4,6,256,2.0,733
27,HP,3,Full HD,1,1,7,39.624,2.7,8,256,1.91,913
28,HP,4,IPS Panel,3,1,7,33.02,2.7,8,256,1.38,1421
29,HP,3,IPS Panel,3,1,5,35.56,2.5,6,256,1.8622317596566522,837
30,Asus,4,IPS Panel,3,1,7,35.56,2.7,8,256,1.3,1515
31,Lenovo,3,IPS Panel,2,1,7,35.56,2.7,8,256,1.58,1880
32,Dell,4,IPS Panel,2,1,5,33.02,1.6,8,256,1.21,2069
```

Step 3: You can open the link and perform the following on the dataset:

- Copy all the data and paste it into an Excel sheet.
- Click **Data**, then click **Text to Columns**, select **Comma** as the delimiter.
- Save the new dataset as a csv file.

  Step 4: Write a prompt to ask how to download the generated csv using python. Chatscv will provide you with the code to download the csv from the provided link.



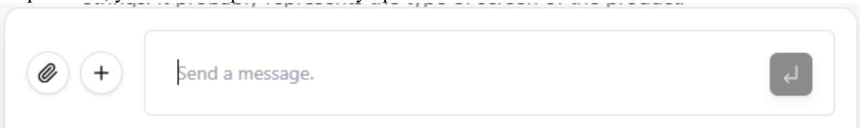
```python
import pandas as pd

url = "https://kiycoesgtwolawkyqacm.supabase.co/storage/v1/
df = pd.read_csv(url)
```

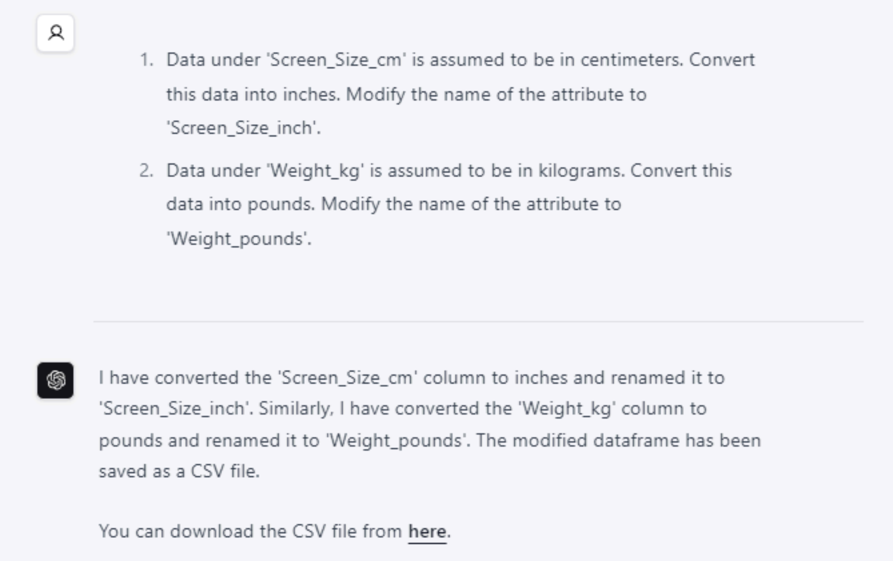
This code will read the CSV file from the URL and store it in a pandas DataFrame named `df`. You can then perform any operations you want on this DataFrame.

Task 4: Perform the data standardization

Note that the parameter 'Screen_Size_cm' represents the screen size in centimeters instead of the standard unit, which is inches. Also, the parameter 'Weight_kg' needs to be expressed as pounds.
Step 1: First, you need to upload the newly updated dataset. Click the '+' icon and attach the new dataset you saved in the previous step.



Step 2: Write the following prompts in the chat box to change the attributes into standard units:
1. Data under 'Screen_Size_cm' is assumed to be in centimeters. Convert this data into inches. Modify the name of the attribute to 'Screen_Size_inch'.
2. Data under 'Weight_kg' is assumed to be in kilograms. Convert this data into pounds. Modify the name of the attribute to 'Weight_pounds'.



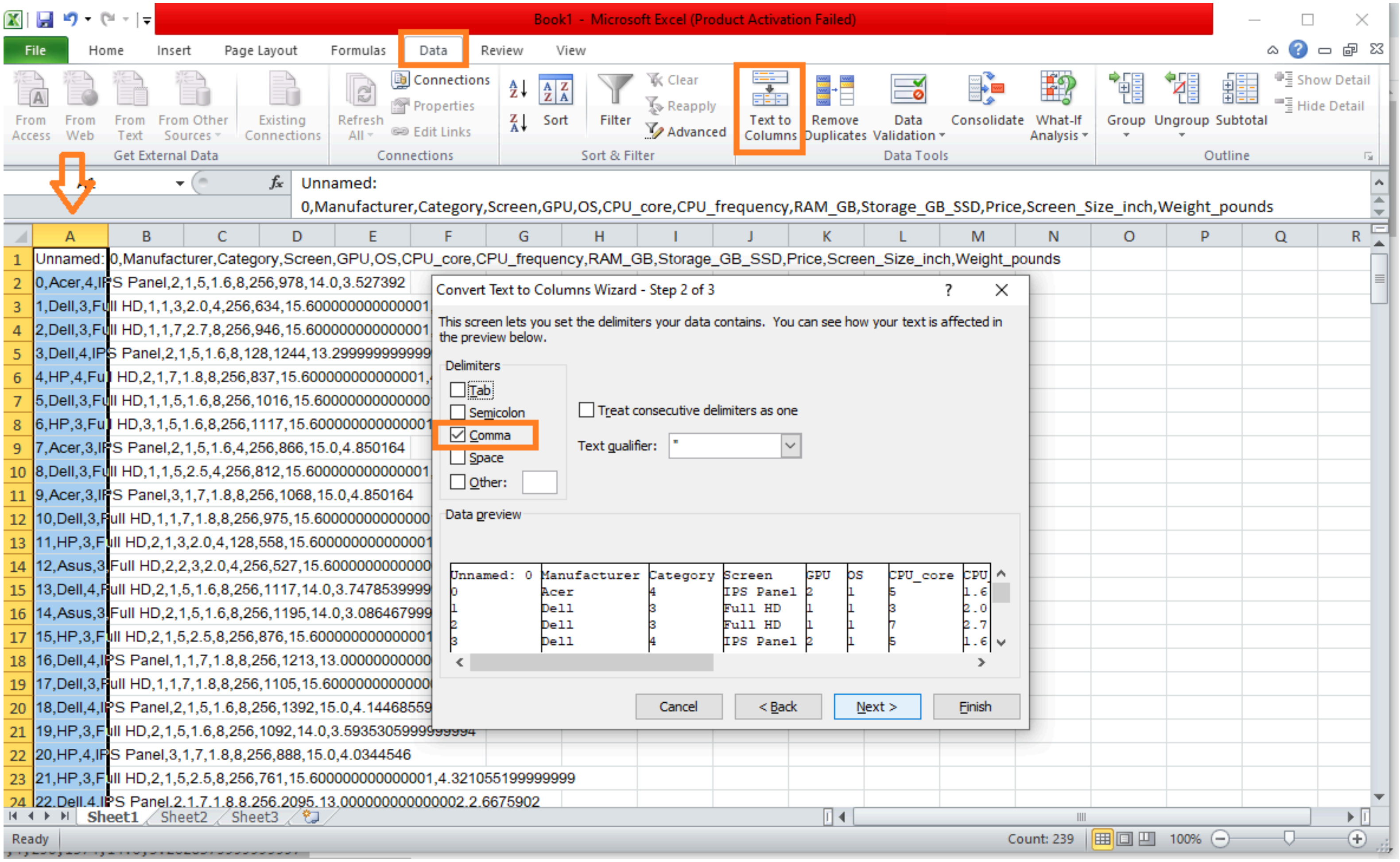
1. Data under 'Screen_Size_cm' is assumed to be in centimeters. Convert this data into inches. Modify the name of the attribute to 'Screen_Size_inch'.
2. Data under 'Weight_kg' is assumed to be in kilograms. Convert this data into pounds. Modify the name of the attribute to 'Weight_pounds'.

I have converted the 'Screen_Size_cm' column to inches and renamed it to 'Screen_Size_inch'. Similarly, I have converted the 'Weight_kg' column to pounds and renamed it to 'Weight_pounds'. The modified dataframe has been saved as a CSV file.

You can download the CSV file from here.

Step 3: You can open the link and perform the following on the dataset:

- Copy all the data and paste it into an Excel sheet.
- Click **Data**, then click **Text to Columns** , select **Comma** as the delimiter.
- Save the new dataset as a csv file.



Unnamed: 0,Manufacturer,Category,Screen,GPU,OS,CPU_core,CPU_frequency,RAM_GB,Storage_GB_SSD,Price,Screen_Size_inch,Weight_pounds

```
1  Unnamed: 0,Manufacturer,Category,Screen,GPU,OS,CPU_core,CPU_frequency,RAM_GB,Storage_GB_SSD,Price,Screen_Size_inch,Weight_pounds
2  0,Acer,4,IPS Panel,2,1,5,1.6,8,256,978,14.0,3.527392
3  1,Dell,3,Full HD,1,1,3,2.0,4,256,634,15.600000000000001
4  2,Dell,3,Full HD,1,1,7,2.7,8,256,946,15.600000000000001
5  3,Dell,4,IPS Panel,2,1,5,1.6,8,128,1244,13.299999999999
6  4,HP,4,Full HD,2,1,7,1.8,8,256,837,15.600000000000001,
7  5,Dell,3,Full HD,1,1,5,1.6,8,256,1016,15.600000000000
8  6,HP,3,Full HD,3,1,5,1.6,8,256,1117,15.600000000000001
9  7,Acer,3,IPS Panel,2,1,5,1.6,4,256,866,15.0,4.850164
10 8,Dell,3,Full HD,1,1,5,2.5,4,256,812,15.600000000000001
11 9,Acer,3,IPS Panel,3,1,7,1.8,8,256,1068,15.0,4.850164
12 10,Dell,3,Full HD,1,1,7,1.8,8,256,975,15.600000000000
13 11,HP,3,Full HD,2,1,3,2.0,4,128,558,15.600000000000001
14 12,Asus,3,Full HD,2,2,3,2.0,4,256,527,15.600000000000
15 13,Dell,4,Full HD,2,1,5,1.6,8,256,1117,14.0,3.7478539999
16 14,Asus,3,Full HD,2,1,5,1.6,8,256,1195,14.0,3.086467999
17 15,HP,3,Full HD,2,1,5,2.5,8,256,876,15.600000000000001
18 16,Dell,4,IPS Panel,1,1,7,1.8,8,256,1213,13.00000000000
19 17,Dell,3,Full HD,1,1,7,1.8,8,256,1105,15.600000000000
20 18,Dell,4,IPS Panel,2,1,5,1.6,8,256,1392,15.0,4.14468559
21 19,HP,3,Full HD,2,1,5,1.6,8,256,1092,14.0,3.593530599999994
22 20,HP,4,IPS Panel,3,1,7,1.8,8,256,888,15.0,4.0344546
23 21,HP,3,Full HD,2,1,5,2.5,8,256,761,15.600000000000001,4.321055199999999
24 22,Dell,4,IPS Panel,2,1,7,1.8,8,256,2095,13.000000000000002,2.6675902
```

Convert Text to Columns Wizard - Step 2 of 3

This screen lets you set the delimiters your data contains. You can see how your text is affected in the preview below.

Delimiters

- □ Tab
- □ Semicolon
- ☑ Comma
- □ Space
- □ Other:

□ Treat consecutive delimiters as one

Text qualifier: "

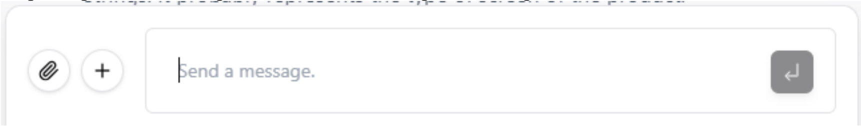
Data preview

```
Unnamed: 0 Manufacturer Category Screen    GPU OS CPU_core CPU
0          Acer         4        IPS Panel 2   1  5        1.6
1          Dell         3        Full HD   1   1  3        2.0
2          Dell         3        Full HD   1   1  7        2.7
3          Dell         4        IPS Panel 2   1  5        1.6
```

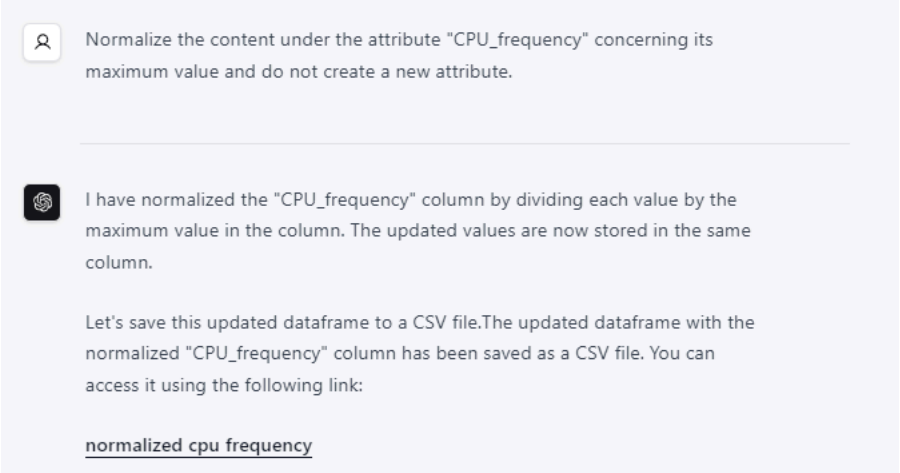
Task 5: Perform the data normalization

You may require normalizing the data under some attributes. Since there are many normalization forms, you need to mention the exact needs and tasks. Also, you can save the normalized data as a new attribute or change the original attribute. You need to provide clear details in the prompt. For example, assume that the data under 'CPU_frequency' needs to be normalized as per the maximum value under the attribute. You want the changes to reflect directly under the attribute instead of creating a new attribute.

Step 1: Upload the new updated dataset from the previous step. Click the '+' icon and attach the new dataset.



Step 2: Write a prompt: "Normalize the content under the attribute' CPU_frequency' concerning its maximum value, and do not create a new attribute. "



Normalize the content under the attribute "CPU_frequency" concerning its maximum value and do not create a new attribute.

I have normalized the "CPU_frequency" column by dividing each value by the maximum value in the column. The updated values are now stored in the same column.

Let's save this updated dataframe to a CSV file.The updated dataframe with the normalized "CPU_frequency" column has been saved as a CSV file. You can access it using the following link:

normalized cpu frequency

Practice problems

- Create a prompt to generate a Python code that converts the values under Price from USD to Euros.
- Modify the normalization prompt to perform min-max normalization on the CPU_frequency parameter.

Conclusion

In this lab, you have learned to handle missing values in your dataset, and performed data standardization and data normalization.

Author(s)

Dr. Pooja

Skills Network