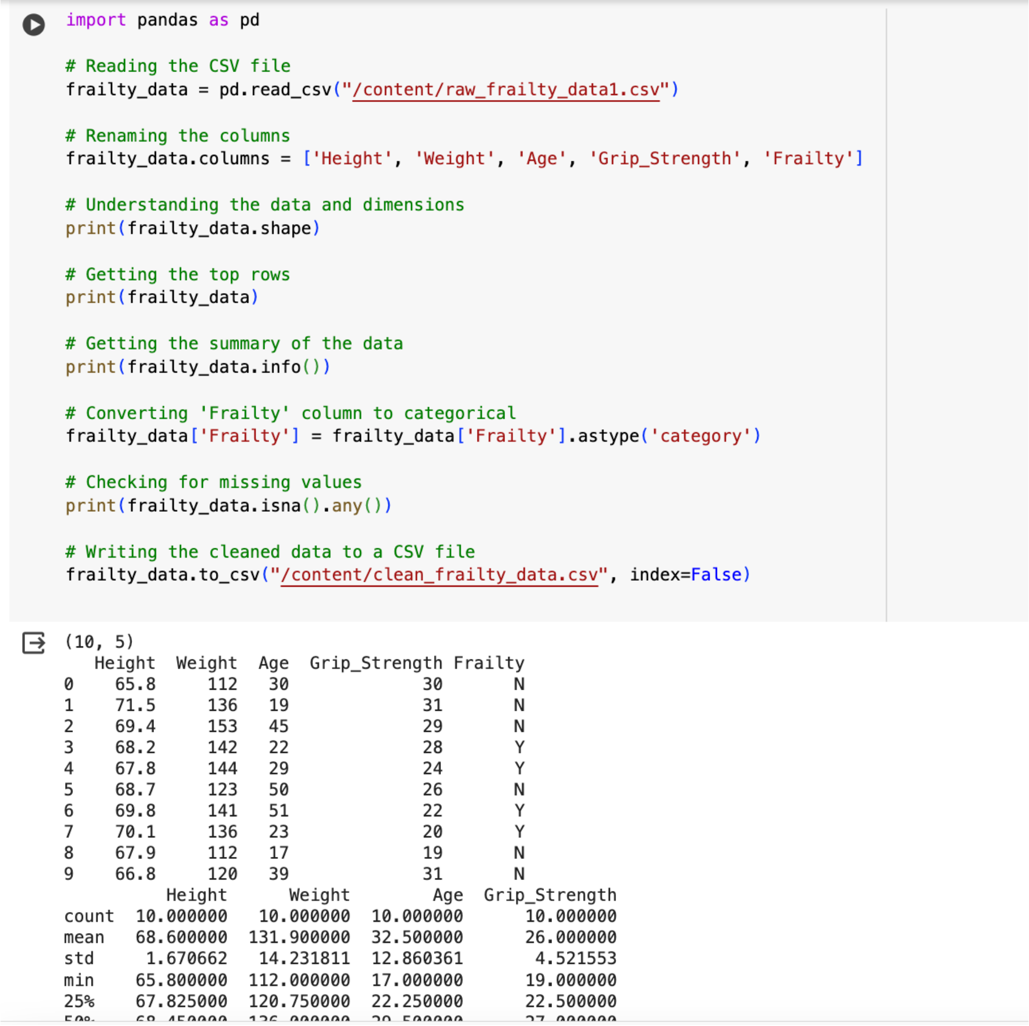
Question1 -16345434

Data cleaning:

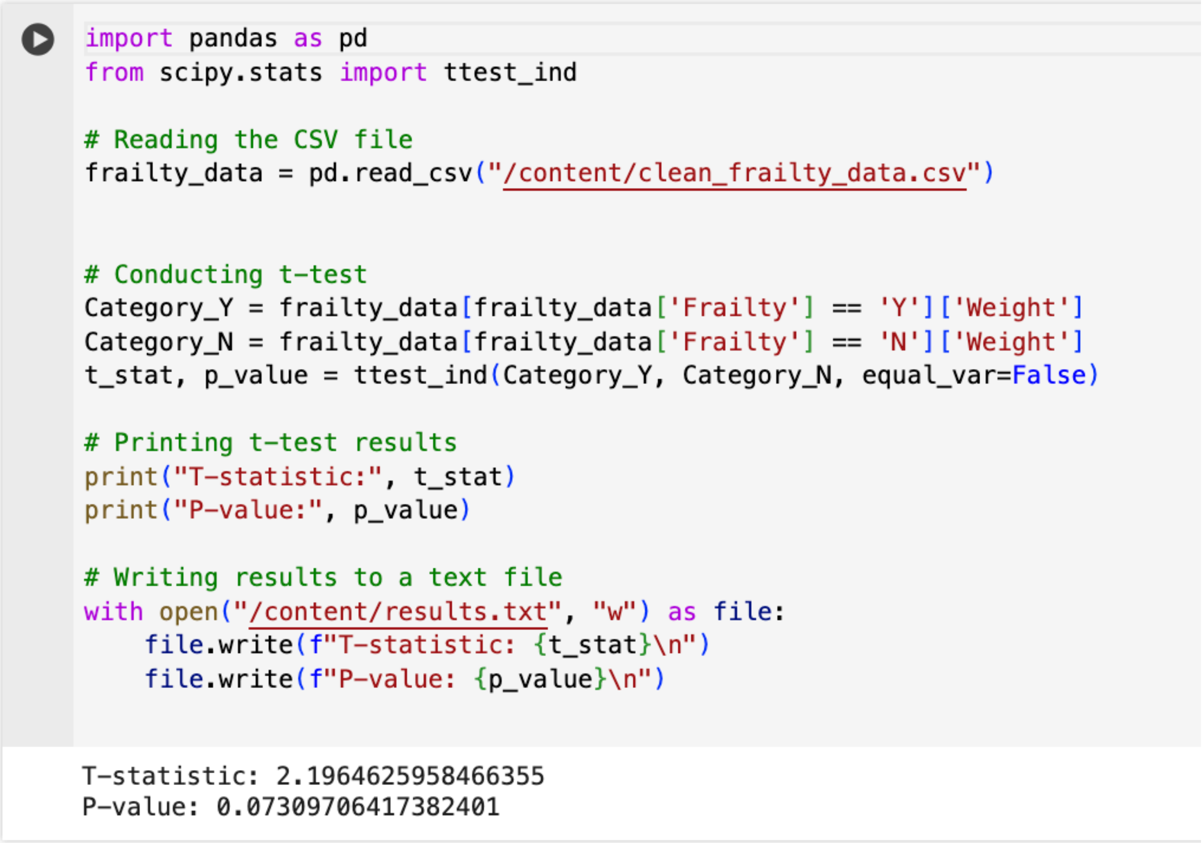


```python
import pandas as pd

# Reading the CSV file
frailty_data = pd.read_csv("/content/raw_frailty_data1.csv")

# Renaming the columns
frailty_data.columns = ['Height', 'Weight', 'Age', 'Grip_Strength', 'Frailty']

# Understanding the data and dimensions
print(frailty_data.shape)

# Getting the top rows
print(frailty_data)

# Getting the summary of the data
print(frailty_data.info())

# Converting 'Frailty' column to categorical
frailty_data['Frailty'] = frailty_data['Frailty'].astype('category')

# Checking for missing values
print(frailty_data.isna().any())

# Writing the cleaned data to a CSV file
frailty_data.to_csv("/content/clean_frailty_data.csv", index=False)
```

```
(10, 5)
   Height  Weight  Age  Grip_Strength Frailty
0    65.8     112   30             30       N
1    71.5     136   19             31       N
2    69.4     153   45             29       N
3    68.2     142   22             28       Y
4    67.8     144   29             24       Y
5    68.7     123   50             26       N
6    69.8     141   51             22       Y
7    70.1     136   23             20       Y
8    67.9     112   17             19       N
9    66.8     120   39             31       N
          Height      Weight        Age  Grip_Strength
count  10.000000   10.000000  10.000000      10.000000
mean   68.600000  131.900000  32.500000      26.000000
std     1.670662   14.231811  12.860361       4.521553
min    65.800000  112.000000  17.000000      19.000000
25%    67.825000  120.750000  22.250000      22.500000
50%    68.450000  136.000000  29.500000      27.000000
```

Data Analysis:

1. **Importing Libraries**:
   - **pandas** is imported as **pd** to work with data frames.
   - **ttest_ind** from **scipy.stats** is imported to conduct an independent t-test.
2. **Reading the CSV file**:
   - The code reads the dataset **clean_frailty_data.csv** into a pandas DataFrame **frailty_data**.
3. **Conducting t-test**:
   - The two categories: 'Y' (indicating frailty) and 'N' (indicating non-frailty) from the DataFrame.
   - Conducting an independent t-test (**ttest_ind**) with the two sets of weight data, specifying **equal_var = False** to perform Welch's t-test, which does not assume equal variances between the groups.
4. **Printing t-test results**:
   - The t-test statistic (**t_stat**) and the corresponding p-value (**p_value**) are printed to the console.



```python
import pandas as pd
from scipy.stats import ttest_ind

# Reading the CSV file
frailty_data = pd.read_csv("/content/clean_frailty_data.csv")


# Conducting t-test
Category_Y = frailty_data[frailty_data['Frailty'] == 'Y']['Weight']
Category_N = frailty_data[frailty_data['Frailty'] == 'N']['Weight']
t_stat, p_value = ttest_ind(Category_Y, Category_N, equal_var=False)

# Printing t-test results
print("T-statistic:", t_stat)
print("P-value:", p_value)

# Writing results to a text file
with open("/content/results.txt", "w") as file:
    file.write(f"T-statistic: {t_stat}\n")
    file.write(f"P-value: {p_value}\n")
```

```
T-statistic: 2.1964625958466355
P-value: 0.07309706417382401
```