Python Chapter 2: MPG

Contents

Red light, No AI

It is important that you are able to do this assignment on your own.

Time required: 60 minutes

- Comment your code as shown in the tutorials and other code examples.
- Follow all directions carefully and accurately.
- Think of the directions as minimum requirements.

Pseudocode

1. Write pseudocode or TODO for the exercise
2. Submit with the assignment

TODO

A TODO outline is a form of pseudocode. It is a way of thinking through and solving the problem before you write the code. You can copy and paste this TODO to get started.

```
# TODO: Get the starting mileage from user, assign to variable

# TODO: Get the ending mileage from user, assign to variable

# TODO: Get the number of gallons used from user, assign to variable

# TODO: Calculate the total mileage, assign to variable

# TODO: Calculate mpg by dividing miles by gallons, assign to variable

# TODO: Round MPG to 2 decimal places, assign to variable

# TODO: Display the mpg
```

Rounding Off Numbers in Python

```
1 # y represents any variable that we want to round off
2 y = 5.76543
3 # This will round off y to 2 decimal places
4 x = round(y, 2)
5 # Print the results
6 print(x)
```

Example run:

```
5.77
```

The example above rounds the value y to 2 decimal places. You can assign the result of the rounding process to the same variable you are rounding, or to a different variable.

Requirements

1. Create a Python program named **mpg.py**.
2. Get input from the user for starting miles and ending miles
3. Get gallons used.
4. Calculate miles driven.
5. Calculate the miles per gallon.
6. Use the round function to round the MPG to 2 decimal places.

Example run:

```
Starting mileage: 101
Ending mileage: 254
Gallons of gas: 3.1
Your car gets: 49.35 MPG
```

Challenge

Is your vehicle fuel efficient? Add an if elif else statement that let's the user know if their vehicle is:

- High fuel efficiency
- Average fuel efficiency
- Low fuel efficiency

Assignment Submission

1. Attach the pseudocode.
2. Attach the program files.
3. Attach screenshots showing the successful operation of the program.
4. Submit in Blackboard.