Write a function driving_cost() with input parameters miles_per_gallon, dollars_per_gallon, and miles_driven, that returns the dollar cost to drive those miles. All items are of type float. The function called with arguments (20.0, 3.1599, 50.0) returns 7.89975.

Define that function in a program whose inputs are the car's miles per gallon and the price of gas in dollars per gallon (both float). Output the gas cost for 10 miles, 50 miles, and 400 miles, by calling your driving_cost() function three times.

Output each floating-point value with two digits after the decimal point, which can be achieved as follows:
`print(f'{your_value:.2f}')`

Ex: If the input is:

`20.0`

`3.1599`

the output is:

`1.58`

`7.90`

`63.20`

Your program must define and call a function:
`def driving_cost(miles_per_gallon, dollars_per_gallon, miles_driven)`

Solution

in_class.py

```
def driving_cost(miles_per_gallon, dollars_per_gallon, miles_driven):
    for _ in miles_driven:
        print(
            f"Gas cost for {_} miles: ${_/miles_per_gallon * dollars_per_gallon:.2f}")


def main():
    miles_per_gallon = float(input("Miles per gallon? "))
    dollars_per_gallon = float(input("Dollars per gallon? $"))
    driving_cost(miles_per_gallon, dollars_per_gallon,
                 miles_driven=[10, 50, 400])


if __name__ == "__main__":
    main()
```

PROBLEMS DEBUG CONSOLE TERMINAL bash

```
[suz@archlinux 11]$ python in_class.py
Miles per gallon? 20.0
Dollars per gallon? $3.1599
Gas cost for 10 miles: $1.58
Gas cost for 50 miles: $7.90
Gas cost for 400 miles: $63.20
[suz@archlinux 11]$
```

Problem # 2

A pedometer treats walking 1 step as walking 2.5 feet. Define a function named feet_to_steps that takes a float as a parameter, representing the number of feet walked, and returns an integer that represents the number of steps walked. Then, write a main program that reads the number of feet walked as an input, calls function feet_to_steps() with the input as an argument, and outputs the number of steps as an integer.

Use floating-point arithmetic to perform the conversion.

Ex: If the input is: `150.5`
the output is: `60`

The program must define and call the following function:
`def feet_to_steps(user_feet)`

Solution

in_class_2.py

```
def feet_to_steps(user_feets):
    return user_feets//2.5


def main():
    print(
        f"Equivalent number of steps: {int(feet_to_steps(float(input('Number of feet walked: '))))}")


if __name__ == "__main__":
    main()
```

PROBLEMS DEBUG CONSOLE TERMINAL

```
[suz@archlinux 11]$ python in_class_2.py
Number of feet walked: 150.5
Equivalent number of steps: 60
```