Name_____ 05.04 Classes and Methods

This assignment has three parts.

Part One: Design the program

Write a program to define a Superhero class, set attributes and call the methods. Use the following guidelines to write your program:

1. Look over the code that starts the Superhero class. You may use this code as a starting point. Come up with **two** additional attributes and **one** method to add to the class. Be creative! Some attributes could be a motto, villain, strength, weakness, or alter ego. An action might be saveWorld() or transformHero().

```
class Superhero:

# Superhero class represents the facts related to a
superhero.

    def __init__(self, name = "", strengthPts = 0, power =
“”):

# Create a new Superhero with a name and other attributes

        self.name = name

        self.strengthPts = strengthPts

    def addStrengthPts(self, points):

# Adds points to the superhero's strength.

        self.strengthPts = self.strengthPts + points
```

2. Update the class by including at least two new attributes and one new method.
3. In the main() method, create your superhero. Be sure to assign values to its attributes and call its methods.
4. Create output that describes your superhero in a story fashion or a series of events.
5. Write the pseudocode for this program. Be sure to include any needed input, calculations, and output.

Insert your pseudocode here:

- **Class Superhero**
  - **Define initializer**
  - **Define addStrengthPts**
    - **Add point to strengthPts**
  - **Define saveWorld**
    - **If strength is greater or equal to 50**
      - **Print name + " has successfully saved the world from " + villain + "! Motto: \"" + motto + "\""**
    - **Else**
      - **Print name + " needs more strength to defeat " + villain + " and save the world.")**
- **Define main**
  - **Set hero to new Superhero where name is “Captain Valor”, strength is 30, motto is “Justice Always Prevails!” and villain is "Dr. Chaos"**
  - **Print "Introducing our hero: " +name + ". They stand for: \"" +motto + "\"."**
  - **Print "Their current strength points: " + strengthPts + "."**
  - **Print "Their arch-nemesis is: " +villain + "."**
  - **Print "\nCaptain Valor is training to increase their strength..."**
  - **Hero add strength 25**
  - **Print "Captain Valor's updated strength points: " + strengthPts + "."**
  - **Print "\nThe world is in danger!"**
  - **Call hero#saveWorld**

Part Two: Code the program

Use the following guidelines to code your program:

1. To code the program, use the Python IDLE.
2. If desired, copy and paste the Superhero class code above into the IDLE.
3. Using comments, type a heading that includes your name, today's date, and a short description of the program.
4. Follow the Python style conventions regarding indentation and the use of white space to improve readability.
5. Use meaningful variable names.

Example of expected output: The output for your program should resemble the following. Your specific results will vary depending on the choices you make and the input provided.

Output:

```
Let me tell you about my superhero.
Her name is Captain Python and her super power is exterminating logic bugs.
When facing the villain Sinister Syntax, she shouts out her motto
Batteries Included! Then rushes to save the world!
The villain is defeated when he falls into an infinite loop!
This heroic action earns her 100 strength points.
Captain Python now has 121 points and has saved the day!
```

Insert a copy of your code from IDLE here:

```
# Jonathan Meyer
# 11.2.2024
# a program to define a Superhero class, set attributes and call the
methods to save the world.

# Implementation in Python:
class Superhero:
    # Superhero class represents the facts related to a superhero.

    def __init__(self, name="", strengthPts=0, motto="", villain=""):
        # Create a new Superhero with a name and other attributes.
        self.name = name
        self.strengthPts = strengthPts
```

```
        self.motto = motto
        self.villain = villain

    def addStrengthPts(self, points):
        # Adds points to the superhero's strength.
        self.strengthPts = self.strengthPts + points

    def saveWorld(self):
        # If strength points are greater than a threshold, the superhero
saves the world.
        if self.strengthPts >= 50:
            print(self.name + " has successfully saved the world from " +
self.villain + "! Motto: \"" + self.motto + "\"")
        else:
            print(self.name + " needs more strength to defeat " +
self.villain + " and save the world.")

# Main function to create and use the Superhero class
def main():
    # Creating an instance of Superhero
    my_hero = Superhero(name="Captain Valor", strengthPts=30,
motto="Justice Always Prevails!", villain="Dr. Chaos")

    # Describe the hero
    print("Introducing our hero: " + my_hero.name + ". They stand for: \""
+ my_hero.motto + "\".")
    print("Their current strength points: " + str(my_hero.strengthPts) +
".")
    print("Their arch-nemesis is: " + my_hero.villain + ".")

    # Add strength points
    print("\nCaptain Valor is training to increase their strength...")
    my_hero.addStrengthPts(25)
    print("Captain Valor's updated strength points: " +
str(my_hero.strengthPts) + ".")

    # Hero tries to save the world
```

```
    print("\nThe world is in danger!")
    my_hero.saveWorld()

# Calling the main function
main()
```

Part Three: Post Mortem Review

Complete the Post Mortem Review (PMR). Write a thoughtful 2-3 sentence response to each of the questions in the PMR chart.

Review Question	**Response**
What was the purpose of your program?	a program to define a Superhero class, set attributes and call the methods to save the world.
How could your program be useful in the real world?	It could be used to automate workflows through algorithms based on numbers.
What is a problem you ran into, and how did you fix it?	I didnt run into any issues while running nor writing my program.
Describe one thing you would do differently the next time you write a program.	Make the program like a game. Make it an inventory based system.