Object-Oriented Programming with Python (Snake Game)

* Create a virtual environment and install the pygame module (pip install pygame)
* Create a Point class.
  + In the \_\_init\_\_ method of this class, let it accept an x and y coordinate. It should create protected attributes for the coordinates.
  + Create a collides method that takes another point and returns True if the 2 collide, else False
  + Create getters for these protected attributes.
  + Create a method that returns the coordinates as a tuple (x\_coordinate and y\_coordinate).
* Create a Rectangle class that inherits from Point
  + The rectangle \_\_init\_\_ method accepts x and y coordinates as well as a length, a width and a unit. It should make a call to it’s parent \_\_init\_\_ method and also set the length and unit as protected attributes.
  + Override the collides method from the Point class. It should take a point and check if that point exists in any section of the rectangle.
  + Create a collides\_boundary method that accepts a point and returns True if that point is on the boundary of the rectangle.
  + Create getters for these protected attributes.
* Create a Snake class that inherits from Rectangle
  + Its \_\_init\_\_ method should have all the arguments of the Rectangle’s \_\_init\_\_ method and should include the window’s rectangle (A rectangle object made from the window), a direction which is “RIGHT” by default, on\_eat\_fruit, on\_eat\_self and on\_hit\_wall which are all None by default. The \_\_init\_\_ method should do the following
    - Create a protected attribute (\_position) for the position of the snake’s head. This is a point with the x and y coordinates that were passed in the \_\_init\_\_ method.
    - Create a \_body attribute which contains all the points starting from the head to the tail, depending on the length and unit (all passed to \_\_init\_\_)
    - A public fruit\_position which is a rectangle.
    - A private \_\_direction attribute which is set to the direction passed to \_\_init\_\_.
    - A private \_\_number\_of\_fruits\_eaten which is 0
    - A public on\_eat\_fruit that is set to the on\_eat\_fruit passed to \_\_init\_\_
    - A public on\_eat\_self that is set to the on\_eat\_self passed to \_\_init\_\_
    - A public on\_hit\_wall that is set to the on\_hit\_wall passed to \_\_init\_