Solution Review: Inheritance

This lesson discusses the solution for the inheritance problem in the previous lesson.

WE'LL COVER THE FOLLOWING

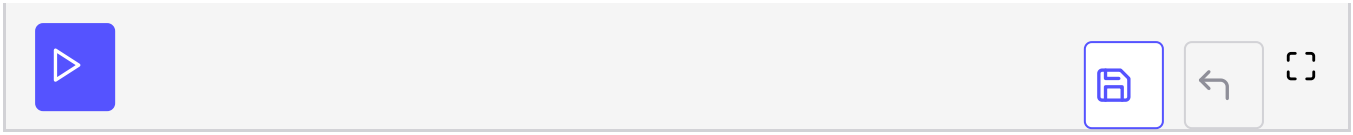
Solution:

Notice that in line 18 `Square` inherits from `Rectangle` and in line 22, it accesses the constructor of the `Rectangle` class using `super()`. If you remove `super().__init__(x1, y1, x2, y2)` from line 22, then code will give an error.

In the end, we created two instances of the `Square` class and calculated their area to test if the inheritance code is correct.

```
class Rectangle:
  def __init__(self, x1, y1, x2, y2): # class constructor
    self.x1 = x1 # class variable
    self.y1 = y1 # class variable
    self.x2 = x2 # class variable
    self.y2 = y2 # class variable

  def width(self):
    return self.x2 - self.x1

  def height(self):
    return self.y2 - self.y1

  def area(self):
    return self.width() * self.height()

#write your code here
class Square(Rectangle):
  def __init__(self, x1, y1, length):
    x2 = x1 + length
    y2 = y1 + length
    super().__init__(x1, y1, x2, y2)

# test your code here
square = Square (2, 7, 7)
print("Length: " + str(square.width()) + ", Area: " + str(square.area()))
square2 = Square (1, 3, 5)
```

```
print("Length: " + str(square2.width()) + ", Area: " + str(square2.area()))
```



In the next chapter, we will study a new concept - iterators in Python.