BIS 420 PROGRAMMING FOR DATA SCIENCE

PRAJAKTA POHARE
CHAPTER 17 EXERCISE 17.5
ILLINOIS STATE UNIVERSITY

Write an add method for Points that works with either a Point object or a tuple:

• If the second operand is a Point, the method should return a new Point whose x coordinate is the sum of the x coordinates of the operands, and likewise for the y coordinates.

• If the second operand is a tuple, the method should add the first element of the tuple to the x coordinate and the second element to the y coordinate, and return a new Point with the result.

```
class Point:
    def __init__(self, x=0, y=0):
        self.x = x
        self.y = y


    def __str__(self):
        return f'({self.x}, {self.y})'


    def add(self, other):
        if isinstance(other, Point):


            return Point(self.x + other.x, self.y + other.y)
        elif isinstance(other, tuple) and len(other) == 2:


            return Point(self.x + other[0], self.y + other[1])
        else:
            raise TypeError("Operand must be either a Point or a tuple of length 2")
```

p1 = Point(2, 3)

p2 = Point(4, 1)

p3 = p1.add(p2)

print(p3)

p4 = p1.add((5, 6))

print(p4)

```
class Point:
    def __init__(self, x=0, y=0):
        self.x = x
        self.y = y

    def __str__(self):
        return f'({self.x}, {self.y})'

    def add(self, other):
        if isinstance(other, Point):

            return Point(self.x + other.x, self.y + other.y)
        elif isinstance(other, tuple) and len(other) == 2:

            return Point(self.x + other[0], self.y + other[1])
        else:
            raise TypeError("Operand must be either a Point or a tuple of length 2")

p1 = Point(2, 3)
p2 = Point(4, 1)
p3 = p1.add(p2)
print(p3)

p4 = p1.add((5, 6))
print(p4)
```