



Oefening 2



In deze reeks maken we oefeningen op klassen. Wanneer in de opgave enkel gevraagd wordt om een bepaalde methode te implementeren, dan is het voldoende om enkel deze methode te geven in het code block. De grader gaat alle nodige stukken code uit vorige opdrachten halen en bij elkaar zetten. Het is daarom belangrijk om de eerste oefeningen in volgorde te maken en ervoor te zorgen dat je oplossingen correct zijn. Merk op dat nadat je een klasse gemaakt hebt, de volgende oefeningen telkens functies zijn met als eerste parameter `self`.

Voorbeeld

Oefening 1:

```
class User:
    def __init__(self, naam):
        self.naam = naam
```

Oefening 2:

```
def toon_naam(self):
    print(self.naam)
```

Your answer passed the tests! Your score is 100.0%. [Submission #671ce6ff3baf09455ab9a934 (2024-10-26 12:56:31)]



Question 1: Rectangle

✓ Perfect



Maak een klasse `Rectangle` met 3 attributen: de linkeronderhoek (`corner`; dit punt is van het type `Point`), de breedte (`width`) en de hoogte (`height`).

```
>>> Rectangle(Point(3,4),1,2)
```

Deze opdracht maakt gebruik van de model-oplossing voor de klasse `Point`, je moet je eigen code hier dus niet importeren.

```
1 class Rectangle:
2     def __init__(self, corner, width, height):
3         self.corner = corner
4         self.width = width
5         self.height = height
```

Submit

Question 2: Area

✓ Perfect



Voeg een methode `area` toe aan de klasse `Rectangle` die de oppervlakte van de rechthoek teruggeeft.

```
1 def area(self):
2     return self.width * self.height
```

Submit

Question 3: Equals

✓ Perfect



Voeg een methode `equals` toe die nakijkt of twee rechthoeken hetzelfde zijn en `True` teruggeeft als die gelijk zijn. De methode krijgt een andere rechthoek als parameter en kijkt na of de twee rechthoeken dezelfde hoogte, breedte en hoekpunt hebben (controleer de x- en y-coördinaten afzonderlijk).

```
>>> r1 = Rectangle(Point(3,4),1,2)
>>> r2 = Rectangle(Point(3,4),1,2)
>>> r3 = Rectangle(Point(3,4),2,2)
>>> r1.equals(r2)
True
>>> r1.equals(r3)
False
```

```
1 def equals(self, r):
2     return self.corner.x == r.corner.x and self.corner.y == r.corner.y and self.width == r.width and
   self.height == r.height
```

Submit