EXERCISES:

OVERLOADING

Exercise(s):

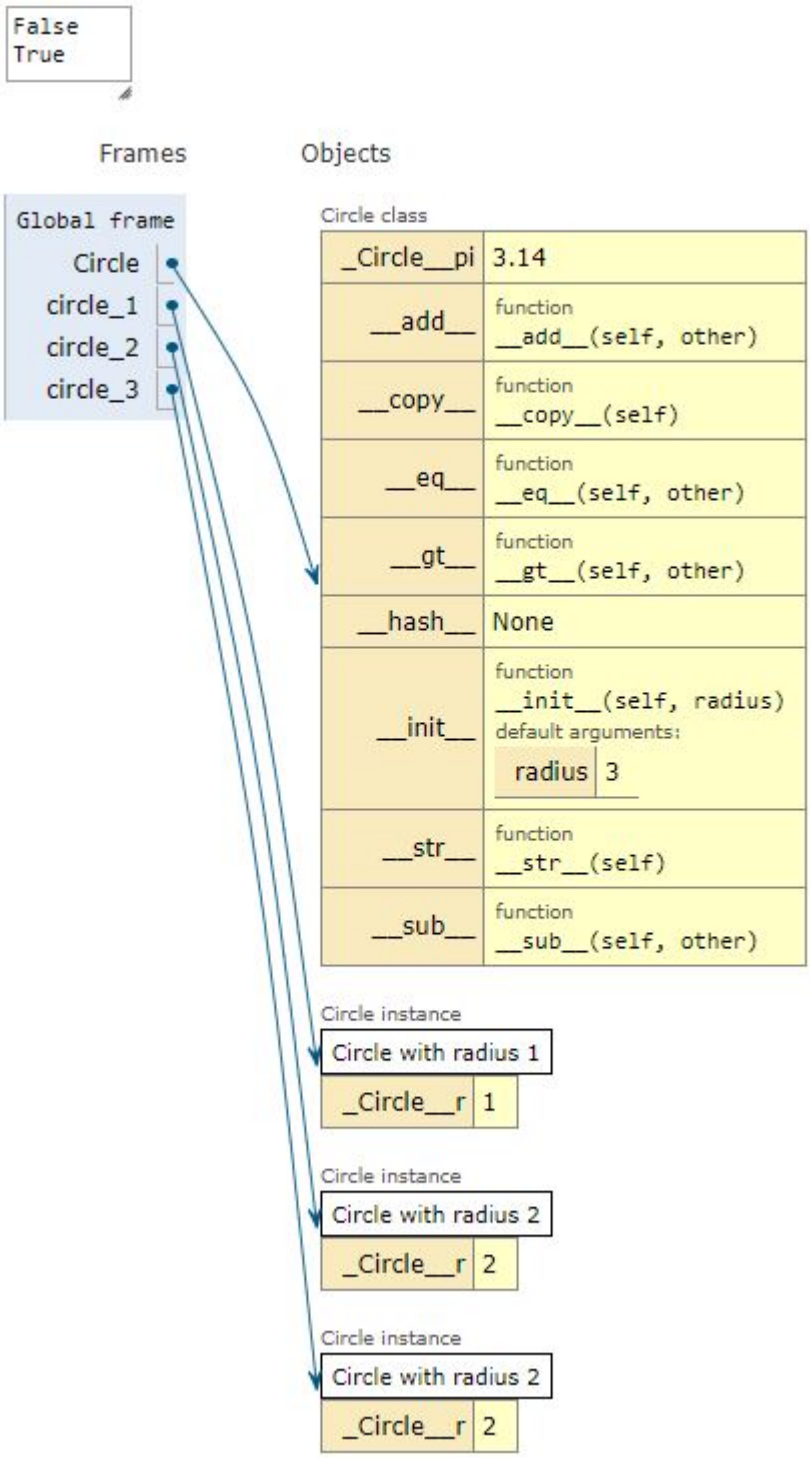
- for the class *Circle* define the following:
  1. '+': new circle with $r = \max(r_1, r_2)$
  2. '−': new circle with $r = \min(r_1, r_2)$
  3. '==': True if both radii are the same
  4. '>': True if $r_1 > r_2$
- run the following script

```
circle_1 = circle(1)
circle_2 = circle(2)
circle_3 = circle_1 + circle_2
print(circle_2 == circle_3)
print(circle_2 >  circle_1)
```

Solution:

```
# for brevity, we omit some methods
class Circle():
    __pi = 3.14
    def __init__(self, radius = 3):
        self.__r  = radius

    def __str__(self):
        return "Circle with radius {}"\
                    .format(self.__r)

    def __add__(self, other):
            if self.__r >= other.__r:
            return Circle(self.__r)
        else:
            return Circle(other.__r)

    def __sub__(self, other):
            if self.__r <= other.__r:
            return Circle(self.__r)
        else:
            return Circle(other.__r)
```

```
    def __eq__(self, other):
        return (self.__r == other.__r)

        def __gt__(self, other):
            return (self.__r > other.__r)


circle_1 = Circle(1)
circle_2 = Circle(2)
circle_3 = circle_1 + circle_2
print(circle_2 == circle_3)
print(circle_2 > circle_1)
```


False
True
Frames
Objects
Global frame
Circle
circle_1
circle_2
circle_3
Circle class
_Circle__pi
3.14
__add__
function
__add__(self, other)
__copy__
function
__copy__(self)
__eq__
function
__eq__(self, other)
__gt__
function
__gt__(self, other)
__hash__
None
__init__
function
__init__(self, radius)
default arguments:
radius
3
__str__
function
__str__(self)
__sub__
function
__sub__(self, other)
Circle instance
Circle with radius 1
_Circle__r
1
Circle instance
Circle with radius 2
_Circle__r
2
Circle instance
Circle with radius 2
_Circle__r
2