MODULE 2 - PYTHON OOPS

Q1. Create a class, Triangle. Its init() method should take self, angle1, angle2, and angle3 as arguments.

Q2. Create a class variable named number_of_sides and set it equal to 3.

Q3. Create a method named check_angles. The sum of a triangle's three angles should return True if the sum is equal 180, and False otherwise. The method should print whether the angles belong to a triangle or not.

Q4. Write methods to verify if the triangle is an acute triangle or obtuse triangle.

Q5. Create an instance of triangle class and call all the defined methods.

Q6. Create three child classes of triangle class - isosceles_triangle, right_triangle and equilateral_triangle.

Q7. Define methods which check for their properties.

Q8. Create a class isosceles_right_triangle which inherits from isosceles_triangle and right_triangle.

Q9. Define method to check for properties.

Solution:

Defining the base class and methods

```
class Triangle:
  def __init__(self,angle1,angle2,angle3):
    self.angle1 = angle1
    self.angle2 = angle2
    self.angle3 = angle3
    self.triangle_type = 'Non-Triangle'

  def check_angles(self):
    if self.angle1 + self.angle2 + self.angle3 == 180:
      print('Angles belong to a triangle')
      self.triangle_type = 'Triangle'
    else:
      print('Angles do not belong to a triangle')

  def check_triangle_type(self):
    self.check_angles()
    if self.triangle_type == 'Triangle':
      if self.angle1 < 90 and self.angle2 < 90 and self.angle3 < 90:
        print('Triangle is an acute triangle')
      elif self.angle1 > 90 or self.angle2 > 90 or self.angle3 > 90:
        print('Triangle is an obtuse triangle')
      else:
        print('Triangle is a right triangle')

triangle1 = Triangle(50,60,70)
triangle1.check_triangle_type()
```

Defining three child classes

```
class Isosceles_Triangle(Triangle):
  def __init__(self,angle1,angle2,angle3):
    Triangle.__init__(self,angle1,angle2,angle3)

  def check_isosceles(self):
    self.check_angles()
    if self.triangle_type == 'Triangle':
        if self.angle1==self.angle2 or self.angle2==self.angle3 or self
.angle1==self.angle3:
          print('Triangle is an isosceles triangle')
          return True
        else:
          print('Triangle is not an isosceles triangle')
          return False

class Right_Triangle(Triangle):
  def __init__(self,angle1,angle2,angle3):
    Triangle.__init__(self,angle1,angle2,angle3)

  def check_right(self):
    self.check_angles()
    if self.triangle_type == 'Triangle':
        if self.angle1==90 or self.angle2==90 or self.angle3==90:
          print('Triangle is a right triangle')
          return True
        else:
          print('Triangle is not a right triangle')
          return False

class Equilateral_Triangle(Triangle):
  def __init__(self,angle1,angle2,angle3):
    Triangle.__init__(self,angle1,angle2,angle3)

  def check_equilateral(self):
    self.check_angles()
    if self.triangle_type == 'Triangle':
        if self.angle1==60 and self.angle2==60 and self.angle1==60:
          print('Triangle is an equilateral triangle')
          return True
        else:
          print('Triangle is not an equilateral triangle')
          return False

triangle2 = Equilateral_Triangle(60,60,60)
```

```
triangle2.check_equilateral()


class Isosceles_Right_Triangle(Isosceles_Triangle,Right_Triangle):
  def __init__(self,angle1,angle2,angle3):
    Triangle.__init__(self,angle1,angle2,angle3)

  def check_isosceles_right(self):
    if self.check_isosceles() == True and self.check_right() == True:
      print('Triangle is an isosceles right triangle')
    else:
      print('Triangle is not an isosceles right triangle')

triangle3 = Isosceles_Right_Triangle(45,45,90)
triangle3.check_isosceles_right()
```