Python Basics 101

Python Program to Find the Square Root

1. # Python Program to calculate the square root
2. # Note: change this value for a different result
3. num = 8
4. # To take the input from the user
5. #num = float(input('Enter a number: '))
6. num\_sqrt = num \*\* 0.5
7. print('The square root of %0.3f is %0.3f'%(num ,num\_sqrt))

# Python Program to Calculate the Area of a Triangle

If a, b and c are three sides of a triangle. Then,

s = (a+b+c)/2

area = √(s(s-a)\*(s-b)\*(s-c))

1. # Python Program to find the area of triangle
2. a = 5
3. b = 6
4. c = 7
5. # Uncomment below to take inputs from the user
6. # a = float(input('Enter first side: '))
7. # b = float(input('Enter second side: '))
8. # c = float(input('Enter third side: '))
9. # calculate the semi-perimeter
10. s = (a + b + c) / 2
11. # calculate the area
12. area = (s\*(s-a)\*(s-b)\*(s-c)) \*\* 0.5
13. print('The area of the triangle is %0.2f' %area)

# Python Program to Solve Quadratic Equation

1. # Solve the quadratic equation ax\*\*2 + bx + c = 0
2. # import complex math module
3. import cmath
4. a = 1
5. b = 5
6. c = 6
7. # calculate the discriminant
8. d = (b\*\*2) - (4\*a\*c)
9. # find two solutions
10. sol1 = (-b-cmath.sqrt(d))/(2\*a)
11. sol2 = (-b+cmath.sqrt(d))/(2\*a)
12. print('The solution are {0} and {1}'.format(sol1,sol2))

# Python Program to Print the Fibonacci sequence

1. # Program to display the Fibonacci sequence up to n-th term
2. nterms = int(input("How many terms? "))
3. # first two terms
4. n1, n2 = 0, 1
5. count = 0
6. # check if the number of terms is valid
7. if nterms <= 0:
8. print("Please enter a positive integer")
9. elif nterms == 1:
10. print("Fibonacci sequence upto",nterms,":")
11. print(n1)
12. else:
13. print("Fibonacci sequence:")
14. while count < nterms:
15. print(n1)
16. nth = n1 + n2
17. # update values
18. n1 = n2
19. n2 = nth
20. count += 1

**Inheritance Basics101**

## What is Inheritance?

Inheritance is a powerful feature in object oriented programming.

It refers to defining a new [class](https://www.programiz.com/python-programming/class) with little or no modification to an existing class. The new class is called **derived (or child) class** and the one from which it inherits is called the **base (or parent) class**.

### Python Inheritance Syntax

class BaseClass:

Body of base class

class DerivedClass(BaseClass):

Body of derived class

Derived class inherits features from the base class, adding new features to it. This results into re-usability of code.

A polygon is a closed figure with 3 or more sides. Say, we have a class called Polygon defined as follows.

1. class Polygon:
2. def \_\_init\_\_(self, no\_of\_sides):
3. self.n = no\_of\_sides
4. self.sides = [0 for i in range(no\_of\_sides)]
5. def inputSides(self):
6. self.sides = [float(input("Enter side "+str(i+1)+" : ")) for i in range(self.n)]
7. def dispSides(self):
8. for i in range(self.n):
9. print("Side",i+1,"is",self.sides[i])