

7. Shape Calculator

Program :

```
from abc import ABC, abstractmethod  
import math
```

```
# Abstraction
```

```
class Shape(ABC):
```

```
    @abstractmethod
```

```
    def area(self):
```

```
        pass
```

```
# Inheritance
```

```
class Rectangle(Shape):
```

```
    def __init__(self, length, breadth):
```

```
        self.length = length
```

```
        self.breadth = breadth
```

```
# Polymorphism (Overriding)

def area(self):
    return self.length * self.breadth
```

```
class Circle(Shape):
```

```
    def __init__(self, radius):
        self.radius = radius
```

```
# Polymorphism (Overriding)

def area(self):
    return math.pi * self.radius * self.radius
```

```
# ---- Main Program ----
```

```
choice = input("Enter Shape (Rectangle/Circle): ")
```

```
if choice.lower() == "rectangle":
```

```
length = float(input("Enter Length: "))

breadth = float(input("Enter Breadth: "))

shape = Rectangle(length, breadth)

print("Area of Rectangle:", shape.area())

elif choice.lower() == "circle":

    radius = float(input("Enter Radius: "))

    shape = Circle(radius)

    print("Area of Circle:", round(shape.area(), 2))

else:

    print("Invalid Shape")
```

Output :

Enter Shape (Rectangle/Circle): Circle

Enter Radius: 10

Area of Circle: 314.16