```
class Bank():
    def __init__(self,owner,balance = 0):
        self.owner = owner
        self.balance = balance

    def deposit(self, balance):
        self.balance +=balance
        return self.balance

    def withdraw(self,balance):
            if self.balance >= balance:
                self.balance -=balance
            else:
                return "Not Possible only " + str(self.balance) + ' is avaliable balance.
            return self.balance


bankholder = Bank("Python" , 7000)


bankholder.deposit(777)
```

```
7777
```

```
bankholder.withdraw(707)
```

```
7070
```

```
bankholder.withdraw(7777)
```

```
'Not Possible only 7070 is avaliable balance. Can you try it again'
```

```
import math
class Cone():
    def __init__(self , Radius = 1 , Height = 1):
        self.Radius = Radius
        self.Height = Height
        self.Volume =0
        self.Surface_Area = 0
    def Volume_Of_Cone(self):
        self.Volume = math.pi * self.Radius * self.Radius * (self.Height / 3)
        return self.Volume
    def Surface_Area_Of_Cone(self):
        Base = math.pi * self.Radius * self.Radius
        Side = math.pi * self.Radius * math.sqrt(self.Radius ** 2 + self.Height ** 2)
        self.Surface_Area = Base + Side
        return self.Surface_Area


Cone_1 = Cone(15 , 7)


Cone_1.Volume_Of_Cone()
```

```
1649.3361431246414
```

```
Cone_1.Surface_Area_Of_Cone()
```

```
1486.897520001703
```