Practice Problems

OOPs

Practice Problems on OOPs

1. Create a Python class called **BankAccount** which represents a bank account, having as attributes: **accountNumber** (numeric type), name (name of the account owner as string type), balance.
2. Create a constructor with parameters: **accountNumber**, name, balance.
3. Create a Deposit() method which manages the deposit actions.
4. Create a **Withdrawal**() method which manages withdrawals actions.
5. Create an **bankFees**() method to apply the bank fees with a percentage of 5% of the balance account.
6. Create a **display**() method to display account details.
7. Give the complete code for the BankAccount class.

Next Page is having Solution

```python
class BankAccount:
    # create the constuctor with parameters: accountNumber, name and balance
    def   init  (self,accountNumber, name, balance):
        self.accountNumber = accountNumber
        self.name = name
        self.balance = balance

    # create Deposit() method
    def Deposit(self , d ):
        self.balance = self.balance + d

    # create Withdrawal method
    def Withdrawal(self , w):
        if(self.balance < w):
            print("impossible operation! Insufficient balance !")
        else:
            self.balance = self.balance - w
    # create bankFees() method
    def bankFees(self):
        self.balance = (95/100)*self.balance

    # create display() method
    def display(self):
        print("Account Number : " , self.accountNumber)
        print("Account Name : " , self.name)
        print("Account Balance : " , self.balance , " $")
```

Practice Question 2

Define a **Circle class** allowing to create a **circleC (O, r)** with center **O(a, b)** and **radius r** using the constructor:

```
def __init__(self,a,b,r):
    self.a = a
    self.b = b
    self.r = r
```

2 - Define a **Area() method** of the class which calculates the area of the circle.

3 - Define a **Perimeter() method** of the class which allows you to calculate the perimeter of the circle.

4 - Define a **testBelongs()** method of the class which allows to test whether a point A(x, y) **belongs** to the circle C(O, r) or not.

```python
from math import pi
class Circle:
    def   init _(self, a, b, r):
        self.a = a
        self.b = b
        self.r = r

    def perimeter (self):
        return  2 * pi * self.r


    def area (self):
        return  pi * self.r**2


    # form of the cercle equation
    def formEquation (self, x, y):
        return (x-self.a)**2 + (y-self.b)**2 - self.r**2

    # method to test if given point blong to the circle or not
    def test belong (self, x, y):
        if (self.formEquation (x, y) == 0):
            print ("the point: (", x, y, ") belongs to the circle C")
        else:
            print ("the point: (", x, y, ") does not belong to the circle C"
```