xje4ytzu9

March 8, 2023

```
[89]: ## ASSIGNMENT 7
      ## KARTIK THAKUR
```

0.0.1 1. Create a bank account class that has two attributes:

owner balance and two methods: deposit withdraw As an added requirement, withdrawals may not exceed the available balance. Instantiate your class, make several deposits and withdrawals, and test to make sure the account can't be overdrawn.

```python
[88]: class BankAccount():
          def __init__(self, owner, balance):
              self.owner = owner
              self.balance = int(balance)



          def deposit(self, deposit):
              self.balance = self.balance + deposit
              print("you depostited : \u20B9",deposit)
              print("your new balane is now :\u20B9",self.balance)

          def withdraw(self, withdraw):

              if withdraw <= self.balance:
                self.balance = self.balance - withdraw
                print("you withdrew :\u20B9" ,withdraw)
                print("balance after withdrawl :\u20B9",self.balance)

              else:
                  print("ALETRT! , WITHDRAW AMOUNT CAN NOT EXCCED DEBIT BALANCE\nYOU␣
      ↪TRIED TO WITHDRAW \u20B9",withdraw,"\u20B9 10 FINE WIL BE DEDUCTED NEXT␣
      ↪TIME")
```

```python
kartik = BankAccount("Kartik",100)


print("STARTED BALANCE\u20B9",kartik.balance)

kartik.withdraw(10)
kartik.deposit(1000)
kartik.withdraw(99)
kartik.deposit(122)
# now if i withdraw more than my balance or status
kartik.withdraw(10000)
```

```
STARTED BALANCE  100
you withdrew :  10
balance after withdrawl :  90
you depostited :   1000
your new balane is now :  1090
you withdrew :  99
balance after withdrawl :  991
you depostited :   122
your new balane is now :  1113
ALETRT! , WITHDRAW AMOUNT CAN NOT EXCCED DEBIT BALANCE
YOU TRIED TO WITHDRAW   10000   10 FINE WIL BE DEDUCTED NEXT TIME
```

0.0.2   2. Give example of Polymorphism in addition operator

```python
[99]:  # poly means many , morpism means to adapt to new enviornmnet or situation
       # for + operator polymorpism is quite obvious when it tries to add different
        ↪data types like

       # adding numbers and numbers
       a = 12 # let any no.
       b= 12

       a+b # this is first example
       print(a+b)

       #2nd adding strings with strings

       c= "kartik"
       d="thakur"

       print(c+d)   # even tough + is for numbers but still it addds two strings
        ↪together called concatination

       #3 adding digits and strings ..even tough WE CAN'T ADD NO. AND STRINGS
```

```
# just to show , we can convert number into string ..which is widely used in␣
 ↪user interface appls

e = "kartik"
f = str("21")
e+f

# so this is polymorphism for you , where one operator can perfrom operations␣
 ↪based on situation
# python is developed keeping this concept in mind which will be applied in␣
 ↪oopos soon ..
```

```
24
kartikthakur
```

[99]: 'kartik21'

[ ]: