**Python classes:**

**OBJECT** <------------ class

**OBJECT** <------------- istance

Ex:

class *ClassName*(baseclass):

suite

class *MyClass*():

pass

myObj **=** *MyClass*() -------> **istance call**

**Classes Attributes:**

Class *MyClass*:

myAttr = 10

n1 = *MyClass*()

n2 = *MyClass*()

BASH INPUT: >> n1.myAttr

BASH OUTPUT: 10

As we can see, if we put some attribute inside our class, we can call them back using **className**.*paramName.*

**Classes methods:**

**Ex\_1:**

class *MyClass*:

def *myMethod*(self):

print(id(self))

BASH INPUT:

>> n1=*myClass*()

>> n1.*myMethod*()

BASH OUTPUT: 154589152

In this case, the program calls our class method and it prints out the instance id.

**Ex\_2:**

class MyClass:

def *myMethod*(self,message):

print(message)

BASH INPUT:

>> n1 = *MyClass*()

>> n1.*myMethod*(‘python’)

BASH OUTPUT: ‘*python’*

**Instance attributes:**

**Ex\_1:**

class *MyClass*:

def *setMessage*(self,message):

self.message = message ---------> INSTANCE attribute

def *printMessage*(self):

print(self.message)

BASH INPUT:

>> n1 = *MyClass*()

>> n1 = *MyClass*()

>> n1.*setMessage*(‘primo’)

>> n2.*setMessage*(‘secondo’)

>> n1.*printMessage*()

>> n2.*printMessage*()

BASH OUTPUT:

*‘primo’*

*‘secondo’*

Here we’ve some new stuff, like *setMessage* and *printMessage* methods. If we add declaration of a new instance, but we don’t call it the *setMessage* method, when we’ll call *printMessage* on that instance it will return an exception.

**Initialization constructor:**

**Ex\_1:**

class *MyClass:*

def *\_\_init\_\_*(self,message):

self.message = message

def *printMessage*(self):

print(self.message)

BASH INPUT:

>> n1 = *MyClass*(‘primo’)

>> n1.*printMessage*()

BASH OUTPUT:

*‘primo’*

In this case we’ve created a constructor that can be called by an instance to istant set a series of attributes.

**Ex\_1:**

class *MyClass:*

counter = 0

def *\_\_init\_\_(*self*)*:

MyClass.counter += 1

@classmethod ---------------> It defines that the next definitions are reffered to the class

def *istanze*(cls):

print(cls.*counter*)

BASH INPUT:

>> n1 = *MyClass*()

>> n2 = *MyClass*()

>> n3 = *MyClass*()

>> MyClass.istanze()

BASH OUTPUT:

*3*

This function counts how many instances of MyClass have been set.

**Inheritance:**

**EX\_1:**

class *BClass*:

pass

class *AClass*(*BClass*):

pass

BASH INPUT:

>> n1 = *AClass*()

>> *isinstance*(n1,AClass)

>> *isinstance*(n1,BClass)

BASH OUTPUT:

*True*

*True*

**EX\_2:**

class *BClass*:

def *setMessage*(self,message):

self.message=message

def *printMessage*(self):

print(self.message)

class *AClass*(BClass):

def *printMessage*(self):

print(self.message)

BASH INPUT:

>> n1 = *AClass*()

>> n1.*setMessage*(‘PYTHON’)

>> n1.*printMessage()*

BASH OUTPUT:

*‘PYTHON’*

**EX\_3:**

class *BClass*:

def *\_\_init­­\_\_*(self,message):

self.message=message

def printMessage(self):

print(self.message)

class *AClass*(BClass):

def *\_\_init\_\_*(self,valore):

self.valore=valore

BASH INPUT:

>> n1 = *AClass*(*20*)

>> n1.*printMessage*

>> *n1.valore*

BASH OUTPUT:

*EXCEPTION*

*20*

EXCERCIZE:

1. Define a CC class
2. Define a 3 params initializer ( name , idCC, starter\_import )
3. Class must have 3 instance attributes (nome,conto,saldo)
4. Define an instance method Withdraw w/ “amount” param w/ no return param, that decreases money on the specified bank account
5. Define an instance method Deposit w/ “amount” w/ no return param, that increases money on the specified bank account
6. Define an instance method “description” that shows nome,conto,saldo

PROGRAM:

class CC:

    def \_\_init\_\_(self,nome,conto,importo\_iniziale):

        self.nome=nome

        self.conto=conto

        self.importo\_iniziale=importo\_iniziale

    def preleva(self,importo):

        self.importo\_iniziale -= importo

    def deposita(self,importo):

        self.importo\_iniziale += importo

    def descrizione(self):

        print("Nome: ", self.nome)

        print("Numero conto: ", self.conto)

        print("Saldo: ", self.importo\_iniziale)

nome = input("Inserisci un nome: ")

numConto = input("Inserisci il numero di conto: ")

importoIniziale = int(input("Inserisci importo iniziale: "))

conti = CC(nome,numConto,importoIniziale)

preleva = int(input("Inserisci soldi da prelevare: "))

conti.preleva(preleva)

conti.descrizione()

deposita = int(input("Inserisci soldi da depositare: "))

conti.deposita(deposita)

conti.descrizione()