1. Creation of class and Object

class cse:

#class attributes

name = ""

rno = ""

#object creation to the class

c1 = cse()

c1.name = "ABC"

c1.rno = "99"

c2 = cse()

c2.name = "XYZ"

c2.rno = "A0"

c3 = cse()

c3.name = "PQR"

c3.rno = "A1"

c4 = cse()

c4.name = "DEF"

c4.rno = "A2"

#Display object

print(c1.name)

print(c1.rno)

print(c2.name)

print(c2.rno)

print(c3.name)

print(c3.rno)

print(c4.name)

print(c4.rno)

Output:

ABC

99

XYZ

A0

PQR

A1

DEF

A2

Capstone project 1:

Develop a python module to create a person class. Include attributes like name, country and date of birth. Implement a method to determine the person's age.

from datetime import date

class person :

def \_\_init\_\_(self,name,country,dob):

self.name = name

self.country = country

self.dob = dob

#Calculate the age of person based on their date of birth

def calage(self):

today = date.today()

age = today.year-self.dob.year

if today < date(today.year,self.dob.month,self.dob.day):

age -= 1

return age

p1 = person("ABC","India",date(2007,1,1))

print("Calculated age for each person")

print("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*")

print("Person 1: ")

print("Name:",p1.name)

print("Country:",p1.country)

print("DOB:",p1.dob)

print("Age:",p1.calage())

Output:

Calculated age for each person

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Person 1:

Name: ABC

Country: India

DOB: 2007-01-01

Age: 18

Capstone project 2:

Develop a python module to create a class representing a shopping cart. Include methods for adding, removing and calculating total price.

class ecommerce:

#initialise the ecommerce with an empty list of items

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

self.items = []

#Add an item with a name and quantity to the ecommerce

def addItem(self,itemName,qty):

item = (itemName,qty)

self.items.append(item)

#Remove an item with a specific name from the ecommerce

def removeItem(self,itemName):

for item in self.items:

if item[0] == itemName:

self.items.remove(item)

break

#Calculate and return the total price of items in ecommerce

def calculateTotal(self):

total = 0

for item in self.items:

total += item[1]

return total

#Example Usage

#Create an instance of ecommerce class

obj1 = ecommerce()

#Add items to the ecommerce class

obj1.addItem("Papaya",100)

obj1.addItem("Guava",200)

obj1.addItem("Orange",150)

#Display the current item and calculate total price

print("Current items in cart: ")

for item in obj1.items:

print(item[0], "=", item[1])

total = obj1.calculateTotal()

print("Total price is: ",total)

#Remove the item in ecommerce

obj1.removeItem("Guava")

print("\n Updated items in ecommerce after removing Guava: ")

for item in obj1.items:

print(item[0],"-",item[1])

total2 = obj1.calculateTotal()

print("Total Price: ",total2)

Output:

Current items in cart:

Papaya = 100

Guava = 200

Orange = 150

Total price is: 450

Updated items in ecommerce after removing Guava:

Papaya - 100

Orange - 150

Total Price: 250