EX NO : 1 FUNCTIONAL STYLE - CONVERTER INR TO OTHER CURRENIES

SOURCE CODE:

print("20-UCA-016")

print("SAI")

print("currency convertor")

inr = int(input("Enter value in inr:"))

res = lambda x:inr/x

print("in dollar:",res(82))

print("in singa:",res(54))

print("in dirham:",res(22))

print("in ringets:",res(40))

OUTPUT:

20-UCA-016

SAI

currency convertor

Enter value in inr:10000

in dollar: 121.95121951219512

in singa: 185.1851851851852

in dirham: 454.54545454545456

in ringets: 250.0

EX NO : 2 IMPERATIVE STYLE - CONVERSION RETURN ON INVESTMENTS

SOURCE CODE:

print("20-UCA-016")

print("SAI")

i = 50000

r =8600

m= 2000

nP =r\*12-m

ROI =(nP/i)\*100

print(ROI)

OUTPUT:

20-UCA-016

SAI

202.4

EX NO : 3 PROCEDURAL STYLE – COMPUTING AGE

SOURCE CODE:

print("20-UCA-016")

print("SAI")

def findage(name, current\_date, current\_month, current\_year,

birth\_date ,birth\_month, birth\_year):

month=[31,28,31,30,31,30,31,31,30,31,30,31]

if(birth\_date > current\_date):

current\_month = current\_month-1

current\_date = current\_date + month[birth\_month-1]

if(birth\_month > current\_month):

current\_year = current\_year-1

current\_month = current\_month + 12

calculated\_date = current\_date - birth\_date;

calculated\_month = current\_month - birth\_month;

calculated\_year = current\_year -birth\_year;

print(name,"your present Age is calculated as below:")

print("years:",calculated\_year,"months:",calculated\_month,"date:",

calculated\_date)

current\_date = 2

current\_month =2

current\_year = 2023

name ="Sai Sharan"

birth\_date = 6

birth\_month = 12

birth\_year = 2002

findage(name, current\_date, current\_month, current\_year, birth\_date,

birth\_month, birth\_year)

OUTPUT:

20-UCA-016

SAI

Sai Sharan your present Age is calculated as below:

years: 20 months: 1 date: 27

EX NO : 4 OBJECTIVE STYLE – COMPUTATION OF ALLOWANCE AND SALARY

SOURCE CODE:

print("20-UCA-016")

print("SAI")

class person:

def \_\_init\_\_(self,name,age):

self.name=name

self.age=age

class employee(person):

def \_\_init\_\_(self, name, age, empno, designation, department, salary):

super().\_\_init\_\_(name, age)

self.empno = empno

self.designation = designation

self.department = department

self.salary = salary

def myfunc(self):

print("Hello my name is "+self.name)

print("I am",self.age,"years old")

print("my employee number",self.empno)

print("my designation",self.designation)

def increment(self,amount,event = None):

if event is not None:

print(event + "Allowance"+ str(amount))

else:

print("Old Salary was:",str(self.salary))

self.salary=self.salary+amount

print("New Salary was:",str(self.salary))

p1=employee(input("Enter a name:"),int(input("enter age:")),int(input("Enter Employee No :")),

(input("enter designation is :")),(input("enter department is :")),int(input("enter salary is :")))

p1.myfunc()

p1.increment(50000)

OUTPUT:

20-UCA-016

SAI

Enter a name:SAI

enter age:20

Enter Employee No :123

enter designation is :MANAGER

enter department is :COMPUTER SCIENCE

enter salary is :20000

Hello my name is SAI

I am 20 years old

my employee number 123

my designation MANAGER

Old Salary was: 20000

New Salary was: 70000

EX NO : 5 COMPUTATION OF INCOME TAX

SOURCE CODE:

print("20-UCA-016")

print("SAI")

tax=0

income=int(input("Enter your income :"))

if(income>100000 and income<300000):

tax=(income-100000)\*0.1

print("tax is = 10%")

elif(income>300000 and income<500000):

tax=20000+(income-300000)\*0.2

print("tax is = 20%")

elif(income>500000):

tax=20000+40000+(income-500000)\*0.3

print("tax is = 30%")

else:

tax=0

print ("Your tax amount is :",tax)

OUTPUT:

20-UCA-016

SAI

Enter your income :350000

tax is = 20%

Your tax amount is : 30000.0

EX NO : 6 GENERATION OF CALENDER

SOURCE CODE:

print("20-UCA-016")

print("SAI")

num\_days=[31,28,31,30,31,30,31,31,30,31,30,31]

months=["January","February","March","April","May","June","July",

"August","September","October","November","December"]

print("Calender - 2023")

i=0

j=0

while(i<len(months)):

print(months[i])

for num\_day in range(1,num\_days[i]+1):

print(num\_day,end=" ")

if(num\_day%7==0):

print(" ")

print("\n")

i=i+1

OUTPUT:

20-UCA-016

SAI

Calender - 2023

January

1 2 3 4 5 6 7

8 9 10 11 12 13 14

15 16 17 18 19 20 21

22 23 24 25 26 27 28

29 30 31

February

1 2 3 4 5 6 7

8 9 10 11 12 13 14

15 16 17 18 19 20 21

22 23 24 25 26 27 28

March

1 2 3 4 5 6 7

8 9 10 11 12 13 14

15 16 17 18 19 20 21

22 23 24 25 26 27 28

29 30 31

April

1 2 3 4 5 6 7

8 9 10 11 12 13 14

15 16 17 18 19 20 21

22 23 24 25 26 27 28

29 30

May

1 2 3 4 5 6 7

8 9 10 11 12 13 14

15 16 17 18 19 20 21

22 23 24 25 26 27 28

29 30 31

June

1 2 3 4 5 6 7

8 9 10 11 12 13 14

15 16 17 18 19 20 21

22 23 24 25 26 27 28

29 30

July

1 2 3 4 5 6 7

8 9 10 11 12 13 14

15 16 17 18 19 20 21

22 23 24 25 26 27 28

29 30 31

August

1 2 3 4 5 6 7

8 9 10 11 12 13 14

15 16 17 18 19 20 21

22 23 24 25 26 27 28

29 30 31

September

1 2 3 4 5 6 7

8 9 10 11 12 13 14

15 16 17 18 19 20 21

22 23 24 25 26 27 28

29 30

October

1 2 3 4 5 6 7

8 9 10 11 12 13 14

15 16 17 18 19 20 21

22 23 24 25 26 27 28

29 30 31

November

1 2 3 4 5 6 7

8 9 10 11 12 13 14

15 16 17 18 19 20 21

22 23 24 25 26 27 28

29 30

December

1 2 3 4 5 6 7

8 9 10 11 12 13 14

15 16 17 18 19 20 21

22 23 24 25 26 27 28

29 30 31

EX NO : 7 INCOME VARIATION ANALYSIS USING STANDARD DEVAIATION

SOURCE CODE:

print("20-UCA-016")

print("SAI")

import math

sum=0

income=[10000,11000,12000,13000,14000]

n=len(income)

print("Total number of people",n)

print("Income=",income)

for x in income:

sum+= x

avg1=sum/n

print("Average=", avg1)

d=0

for i in income:

d+=math.pow(i-avg1,2)

d=d/n

sd=math.sqrt(d)

print("Standard Deviation=",sd)

OUTPUT:

20-UCA-016

SAI

Total number of people 5

Income= [10000, 11000, 12000, 13000, 14000]

Average= 12000.0

Standard Deviation= 1414.213562373095

EX NO : 8 COMPUTING DISTANCE FROM CUSTOMER HOME LOCATION

SOURCE CODE:

print("20-UCA-016")

print("SAI")

import math

def getDist(name1, lat1, lon1, name2, lat2, lon2):

R = 6371

dLat = deg2rad(lat2-lat1)

dLon = deg2rad(lon2-lon1)

a = math.sin(dLat/2) \* math.sin(dLat/2) + math.cos(deg2rad(lat1)) \*

math.cos(deg2rad(lat2)) \* math.sin(dLon/2) \* math.sin(dLon/2)

c = 2 \* math.atan2(math.sqrt(a),math.sqrt(1-a))

d = R\*c

math.trunc(d)

print("DISTANCE BETWEEN",name1, "AND", name2,"IS :", d,"KM")

return(d)

def deg2rad(deg):

return deg \* (math.pi/180)

print('AIR DISTANCE')

getDist("COLLEGE",13.06019,80.23349,"SAI'S HOME",13.06517,80.23197)

getDist("COLLEGE",13.06019,80.23349,"BINITH'S HOME",13.07043,80.21884)

OUTPUT:

20-UCA-016

SAI

AIR DISTANCE

DISTANCE BETWEEN COLLEGE AND SAI'S HOME IS : 0.5777084917653263 KM

DISTANCE BETWEEN COLLEGE AND BINITH'S HOME IS : 1.9530845767281753 KM

EX NO : 9 BANKING – QUEUE IMPLEMENTATION

SOURCE CODE:

print("20-UCA-016")

print("SAI")

q=[]

x=1

a=1

while (a==1):

print("BANKING QUEUE-YOUR CHOICE")

print("1.OBTAIN")

print("2.WHOSE TURN NOW")

print("3.DISPLAY")

print("4.EXIT")

ch=int(input("Enter the choice :"))

if(ch==1):

name=(input("Enter the customer name :"))

y=q.append(name)

y1=q.append(x)

x=x+1

elif(ch==2):

y=q.pop(0)

y1=q.pop(0)

print(y)

print(y1)

elif(ch==3):

print("queue",q)

elif(ch==4):

print("GOOD BYE")

break

else:

print("Invalid Choice")

OUTPUT:

20-UCA-016

SAI

BANKING QUEUE-YOUR CHOICE

1.OBTAIN

2.WHOSE TURN NOW

3.DISPLAY

4.EXIT

Enter the choice :1

Enter the customer name :SAI

BANKING QUEUE-YOUR CHOICE

1.OBTAIN

2.WHOSE TURN NOW

3.DISPLAY

4.EXIT

Enter the choice :1

Enter the customer name :SHREE

BANKING QUEUE-YOUR CHOICE

1.OBTAIN

2.WHOSE TURN NOW

3.DISPLAY

4.EXIT

Enter the choice :1

Enter the customer name :VARUN

BANKING QUEUE-YOUR CHOICE

1.OBTAIN

2.WHOSE TURN NOW

3.DISPLAY

4.EXIT

Enter the choice :3

queue ['SAI', 1, 'SHREE', 2, 'VARUN', 3]

BANKING QUEUE-YOUR CHOICE

1.OBTAIN

2.WHOSE TURN NOW

3.DISPLAY

4.EXIT

Enter the choice :2

SAI

1

BANKING QUEUE-YOUR CHOICE

1.OBTAIN

2.WHOSE TURN NOW

3.DISPLAY

4.EXIT

Enter the choice :3

queue ['SHREE', 2, 'VARUN', 3]

BANKING QUEUE-YOUR CHOICE

1.OBTAIN

2.WHOSE TURN NOW

3.DISPLAY

4.EXIT

Enter the choice :4

GOOD BYE

EX NO : 10 ONLINE PRODUCT BILLING - STACK

SOURCE CODE:

print("20-UCA-016")

print("SAI")

q = []

x=-1

a=1

while(a==1):

print("\n Online cart management\n")

print("1.Add item to cart")

print("2.Remove the new item from the cart")

print("3.display")

print("4.exit")

ch=int(input("Enter the choice:"))

if(ch==1):

name=(input("Enter the customer name :"))

x=x+1

q.insert(x,name)

elif(ch==2):

y=q.pop()

print("The person removed",y)

x=x-1

elif(ch==3):

print("Stack:",q)

elif(ch==4):

break

else:

print("Invalid choice")

OUTPUT:

20-UCA-016

SAI

Online cart management

1.Add item to cart

2.Remove the new item from the cart

3.display

4.exit

Enter the choice:1

Enter the customer name :MILK

Online cart management

1.Add item to cart

2.Remove the new item from the cart

3.display

4.exit

Enter the choice:1

Enter the customer name :PENCIL

Online cart management

1.Add item to cart

2.Remove the new item from the cart

3.display

4.exit

Enter the choice:1

Enter the customer name :NOTE

Online cart management

1.Add item to cart

2.Remove the new item from the cart

3.display

4.exit

Enter the choice:3

Stack: ['MILK', 'PENCIL', 'NOTE']

Online cart management

1.Add item to cart

2.Remove the new item from the cart

3.display

4.exit

Enter the choice:2

The person removed NOTE

Online cart management

1.Add item to cart

2.Remove the new item from the cart

3.display

4.exit

Enter the choice:3

Stack: ['MILK', 'PENCIL']

Online cart management

1.Add item to cart

2.Remove the new item from the cart

3.display

4.exit

Enter the choice:4

GOOD BYE