***#(1) write a program to create one array from another array.***

x= [343,55,63,2,2]

print("original array")

print(x)

y=[]

y[:] = x

print("New Array\n",y)

*#(2) create a program to retrieve, display and update only a range of elements from an array using indexing and slicing in arrays.*

import numpy as np

array = np.array([43,55,22,3,533])

for x in array:

    print(x)

array[0] = 500

print(array)

print(array[0:5:2])

*#note => range of elements word*

***#(3)=> write a program to understand various methods of array class mentioned: append, insert , remove , pop , index , tolist and count***

import array as arr

a = arr.array('i', [52,262,13,22,3,2,52,1])

print(a)

a.append(50)

print("array after append")

a.insert(1,633)

print("array after inserting 633 at 2nd position / 1st index\n",a)

a.remove(50)

print("remove 50 from array\n",a)

a.pop()

print("pop last element of array\n",a)

print("5th index of array",a[5])

print("Array to list=>",a.tolist())

print("Count of 52",a.count(52))

***#(4)=> write a program to sort the array elemnts using bubble sort techniques.***

import array as a

def bubblesort(arr):

    n = len(arr)

    for i in range(n-1):

        for j in range(0,n-i-1):

            if arr[j] > arr[j+1]:

                arr[j],arr[j+1] = arr[j+1],arr[j]

    return arr

a= a.array('i',[53,6,2,2,26,6,263,1,2])

print(bubblesort(a))

***#(5)=> create a program to search the position of an element in an array using index() method of array class.***

import array as np

a = np.array('i',[535,256,6,6,2,6,2,46,2])

pos = a.index(int(input("Enter Element from above list=>")))

print(pos)

'''

element= int(input("Enter the search element=>"))

try:

   pos = a.index(element)

   print(f"element {element} found at position {pos}")

except ValueError:

    print(f" element {element} not found")

'''

***#(6) => write a program to gnerate prime numbers with the help of a function to test prime or not.***

def checkprime():

    num = int(input("Enter a  number->"))

    if num <= 1:

        print(num, " is a not prime number")

        return

    c=0

    for i in range(1,num+1):

        if num % i == 0:

            c += 1

    if c == 2:

        print(num," is a prime number")

    else:

        print(num, " is not a prime number")

checkprime()

***#(7) => write a python program that remove any repeated items from a list so that each item appears at most once. for instance, the list [1,1,2,3,4,3,0,0] would become [1,2,3,4,0]***

list1 = [1,1,2,3,4,0,0]

list2= []

for i in list1:

    if i not in list2:

        list2.append(i)

    '''

    else:

        print(i," value is repeating")

    '''

print(list2)

***#(8) write a program to pass list to a function and display it***

def display\_list(list1):

    for i in list1:

        print(i)

list1=[435,5,23,1,3]

display\_list(list1)

***#(9)=> write a program to demostrate the use of Positional argument, keyword argument and default arguments***

'''

#Positonal Arguments

This type of argument structure means you must provide all values in the exact order expected by the function.

'''

def pos(x,y,z):

    print(x,y,z)

pos("a","b","c")

*#Keyword Argument*

def kw(x,y,z):

    print(x,y,z)

kw(z="c",y="e",x="o")

*#Default argument*

def da(x,y,z="c"):

    print(x,y,z)

da("c","d")

da("a","b","d")

***#(10)=> Write a program to show variable length argument and its use.***

def variable\_length(\*args):

    for i in args:

        print(i)

variable\_length([53,33,6,3,2])

def kwargs(\*\*kwargs):

    for key,value in kwargs.items():

        print("Key=>",key,"\nValue=>",value)

data = {"1":"abc","2":"xyz"}

kwargs(\*\*data)

***#(11)=> write a lambda/anonymouse function to find bigger number in two given numbers.***

val = lambda x,y:max(x,y)

print(val(53,52))

***#(12)=> create a decorator function to increase the value of function by 3.***

def increase(func):

    def wrapper(number):

        return func(number)+3

    return wrapper

@increase

def plus\_three(number):

    return number

*#plus\_three = increase(plus\_three)*

print(plus\_three(10))

'''

**(13)=> create a program name "employee".py and implements the functions DA,HRA,PF and ITAX. create anohter program that uses the function of employee module and calculate gross and net salaries of an employee.**

'''

from employee import DA,HRA,PF,ITAX *#\**

def cal\_salary(basic\_salary):

    da = DA(basic\_salary)

    hra = HRA(basic\_salary)

    pf = PF(basic\_salary)

    gross\_salary = basic\_salary + da + hra

    itax = ITAX(gross\_salary)

    net\_salary = gross\_salary - pf - itax

    return gross\_salary, net\_salary

basic\_salary = float(input("Enter the basic salary->"))

gross\_salary, net\_salary = cal\_salary(basic\_salary)

print(f"Gross salary => {gross\_salary:.2f}")

print(f"Net Salary => {net\_salary}")

***#(14)=> write a programm to create a list using range functions and***

***# perform append,update and delete elements operations in it.***

list1 = [i for i in range(0,9)]

'''

list1 = []

for i in range(0, 9):

    list1.append(i)

'''

print(list1)

list1.append(50)

print(list1)

list1.insert(2,40)

print(list1)

list1.remove(40)

print(list1)

***#(15)=> write a program to combine twolist, perform repetition of lists and***

***#create cloning of lits.***

list1= [i for i in range(1,7)]

list2 =[i for i in range(7,13)]

print(list1)

print(list2)

list3 = list1+list2

print(list3) *#[1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12]*

*#perform repetition*

new\_list = [i for i in list3 for x in (0,1)]

print(new\_list)  *#[1, 1, 2, 2, 3, 3, 4, 4, 5, 5, 6, 6, 7, 7, 8, 8, 9, 9, 10, 10, 11, 11, 12, 12]*

list4=list2[:]

print(list4)

print(list2)

***#(16)=> create a sample list of 7 elements and implements the list methods***

***#mentioned: append,insert,copy, extend, count,remove, pop,sort,reverse and clear***

list1=[i for i in range(0,7)]

print(list1)

*#append*

list1.append(50)

print(list1)

*#insert*

list1.insert(2,50)

print(list1)

*#copy*

list2=list1.copy()

print(list2)

*#extend*

list3=[i for i in range(7,12)]

list1.extend(list3)

print(list1)

*#count*

print(list1.count(50))

*#remove*

list1.remove(50)

print(list1)

*#pop*

print(list1.pop())

print(list1)

*#reverse*

list1.reverse()

print(list1)

*#clear*

list2.clear()

print(list2)

***#(17)=> write a program to create nested list and display its elements.***

list1=[[1,2,3],[4,5,6],[7,8,9]]

for x in list1:

    print(f"\n{x}")

*#print(" ".join(map(str,x)))*

***#(18)=> Write a program to accpect elements in the form of a tuple***

***#and display its minimum, maximum, sum and average.***

tup = (1,23,4,5,6,63,6,3,4)

print("max=>",max(tup))

print("min=>",min(tup))

print("sum=>",sum(tup))

print("Avg=>",sum(tup)/len(tup))

***#(19)=> Creae a program to sort tuple with nested tuples.***

nested\_tuples = ((3, 'apple'), (1, 'banana'), (2, 'orange'))

sorted\_tuples = sorted(nested\_tuples, key=lambda x: x[0])

print(sorted\_tuples)

'''

typ = [(1, (2, 3)), (3, (2, 1)), (2, (2, 1))]

sort\_tup = sorted(typ, key=lambda t: (t[1][1], t[0]))

print(sort\_tup) #[(3, (2, 1)), (2, (2, 1)), (1, (2, 3))]

EXP

'''

***#(20)=> Write a program to create a dictionary from the user and display the elements.***

my\_dict = {'name':"xyz","age":17,"address":"idk dude"}

print(my\_dict["name"])

print(my\_dict["age"])

print(my\_dict["address"])

***#(21)=> Create a dictionary that will accpect cricket players name and***

***#score  in a match. Also we are retrieving runs by entering the player'name***

score\_dict = {}

def add\_player\_score(name,score):

    score\_dict[name] = score

def get\_player\_score(name):

    return score\_dict.get(name,"Player not found")

add\_player\_score("virat kohli",75)

add\_player\_score("rohit sharma",6)

add\_player\_score("ms",57)

player\_name = input("Enter player name=>")

score = get\_player\_score(player\_name)

print(f"{player\_name} => {score}")

***#(22)=> write a program to convert the elements of two list into key-value***

***# pairs of dictionary.***

value=['python','c++','java','ruby']

index = [1,2,3]

dict1=dict(zip(index,value))

print(dict1)

***#(23)=> Create a python function to accpect python function as a dictionary***

***# and display its elements.***

def func(d):

    for key in d:

        print("Key=>",key,"Value=>",d[key])

dict = {1:'a',2:'b'}

func(dict)