**1) Write a program to find out the prime numbers**

***Program :***

# Getting input from the user

mynum = int(input("Enter the Number : "))

if mynum > 1: # Prime numbers are always greater than 1,so here we checking the condition

for i in range(2,mynum):

if (mynum % i) == 0:

print(mynum,"is not a Prime Number")

break

else:

print(mynum,"is a Prime Number")

# if mynum is less than or equal to 1, else part will execute

else:

print(mynum,"is not a Prime Number")

***Output :***

Enter the Number : 4

4 is not a Prime Number

***Explanation :***

1. Getting input from the user. It is stored in the variable mynum.
2. Checking condition if mynum is greater than 1
3. Using for loop again checking the condition if mynum % i is equal to zero. If Yes then printing the number is not a Prime Number and break the execute ie stop the execution.
4. If condition is No, then printing the number is Prime Number.
5. If the condition in step 2 is false , then this step will execute. ie mynum is less than or equal to 1, then print the number is not a Prime Number.

**2) Write a program to create the equation (a+b+c) \* (a-b-c) \* ab + a^2 + b ^2 + (abc)^3**

***Program :***

# Getting a,b,c value from the user

a = int(input("Enter a value : "))

b = int(input("Enter b value : "))

c = int(input("Enter c value : "))

# Performing the equation

equation = ( (a+b+c) \* (a-b-c) \* (a\*b) + (a\*\*2) + (b\*\*2) + ((a\*b\*c)\*\*3) )

# Printing the result

print("The result of the equation is",equation)

***Output :***

Enter a value : 2

Enter b value : 3

Enter c value : 4

The result of the equation is 13567

***Explanation :***

1. Get the a,b,c value from the user.
2. The value of a,b,c passes to the equation and perform the execution.
3. The expression in the parenthesis perform first. a\*\*2 means a to the power 2. (a\*b\*c)\*\*3 means multiplication of a,b,c in the parenthesis perform first and next to the power of 3 perform next.
4. After, the multiplication to be done first and addition to be done next.
5. Finally print the result.

**3) urlist = ['wood','knife','axe'] , mylist = ['tree', 'apple', 'mango', 'melon'] – combine two lists**

***# Method 1 :***

***Program :***

urlist = ['wood','knife','axe']

mylist = ['tree','apple','mango','melon']

# Using extend method, we combine the two lists

urlist.extend(mylist)

# Assigning the combined two list to newlist and printing the newlist

newlist = urlist

print(newlist)

***Output :***

['wood', 'knife', 'axe', 'tree', 'apple', 'mango', 'melon']

***Explanation :***

1. urlist and mylist is given.
2. Using extend method, we combine the two lists. The extend method() which append the mylist at the end of the urlist.
3. Assigning the combined two list to newlist and printing the newlist.

***# Method 2 :***

***Program :***

urlist = ['wood','knife','axe']

mylist = ['tree','apple','mango','melon']

# Using for loop, we combine two lists

for i in range(len(mylist)):

urlist.append(mylist[i])

# Assigning the combined two list to newlist and printing the newlist

newlist = urlist

print(newlist)

***Output :***

['wood', 'knife', 'axe', 'tree', 'apple', 'mango', 'melon']

***Explanation :***

1. urlist and mylist is given.
2. In method 2, we use for loop for combining two lists. In for loop we take the range to the length of mylist. Ie the loop start from 0 and end with 3.
3. Using append method, we append the mylist to urlist.
4. Assigning the urlist to newlist and printing the newlist.

**4) Write a program for natural number based on user input**

***Program :***

# Getting input from user

user\_input = int(input("Enter the Number :"))

print("List of Natural numbers from 1","to",user\_input)

# Using for loop, we printing the numbers from 1 to user\_input

for x in range(1,user\_input+1):

print(x, end = ' ')

***Output :***

Enter the Number :14

List of Natural numbers from 1 to 14

1 2 3 4 5 6 7 8 9 10 11 12 13 14

***Explanation :***

1. Get the input from the user.
2. Using for loop, we print the natural numbers from 1 to user\_input. So in for loop we mentioned start is 1 and end is user\_input + 1.
3. After we print the number.

**5) Write class and function for the equation sqrt(x1-x2) ^ 2 + sqrt( y1 – y2 ) ^2 using try except**

**handling.**

***Program :***

# For performing square and sqrt, we are importing the numpy

import numpy as np

# Creating class

class Equ:

def \_\_init\_\_(self,x1,x2,y1,y2):

self.x1 = x1

self.x2 = x2

self.y1 = y1

self.y2 = y2

def equation(self):

x1\_val = self.x1

x2\_val = self.x2

y1\_val = self.y1

y2\_val = self.y2

x\_value = np.square(x1\_val - x2\_val)

y\_value = np.square(y1\_val - y2\_val)

xsqrt = np.sqrt(x\_value)

ysqrt = np.sqrt(y\_value)

res = xsqrt + ysqrt

print("The Result for the equation is",res)

# Creating objects

try:

E1 = Equ(2,4,3,5) # Passing values here

E1.equation() # Calling the function

# If any error occurs this except part will execute

except:

print("Argument missing. Could not convert string to integer. Try again...")

***Output :***

The Result for the equation is 4.0

***Explanation :***

1. Create the class Equ. In class Equ define the function \_\_init\_\_ to initialize the values of x1,x2,y1,y2.
2. The next function in the class is equation. In this function, we assigning the values of x1,x2,y1,y2 to the variable x1\_val,x2\_val,y1\_val,y2\_val.
3. Using numpy library we import it to perform the square and sqrt.
4. First we perform the square of (x1\_val – x2\_val) and square of (y1\_val – y2\_val). The result of this square is stored in x\_value and y\_value.
5. Next we perform the sqrt of x\_value and sqrt of y\_value and stored the result in xsqrt and ysqrt.
6. Next add the sqrt values and stored in res variable.
7. Print the res.
8. Creating the object E1 to the class Equ. We create the class object in try block. If we pass only the integer values the try block will execute. Otherwise except block will execute.
9. Now pass the values to the class.
10. Now call the function equation. Now it jumps to step 2. Here the values what we pass to the class is assign to the variable and perform the action which is mentioned in step 4.
11. When we pass the string values to the class, try block becomes false and except will execute.
12. While executing the except block, the msg “Argument missing. Could not convert string to integer. Try again...” will display.

**6) Name = “Guvi python” - Write a program to get “python” word from the string.**

***Program :***

Name = "Guvi python"

# Using removeprefix method in string operation, we remove the prefix from the given

string

print(Name.removeprefix("Guvi "))

***Output :***

python

***Explanation :***

1. The given string is “Guvi python”. To get the word “python” from the string, we use the method removeprefix() in string operation.
2. Using this method we remove the prefix “Guvi “ from the Name string. After removing the prefix , we get the string “python” as output.

**7) Using class and function - Write a program for palindrome Ex. Madam**

***Program :***

# Creating class

class P\_drome:

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

self.val = value

def ispalindrome(self):

val1 = self.val

rev\_val = ""

for i in range(len(val1)-1,-1,-1): # this for loop will take the value from backward of string

rev\_val = rev\_val + val1[i]

print("Reversed value :",rev\_val)

# Checking whether the given string and reversed string are equal or not

if val1 == rev\_val:

print(val1,"is Palindrome")

else:

print(val1,"is not Palindrome")

# Creating the object

# Getting input from the user

user\_input = input("Enter the value : ")

# Passing argument

check = P\_drome(user\_input)

# Calling function

check.ispalindrome()

***Output :***

Enter the value : madam

Reversed value : madam

madam is Palindrome

***Explanation :***

1. Creating a class P\_drome. Inside a class there is function called \_\_init\_\_. This function is used to initialize the value.
2. There is other function called ispalindrome(). In this function, we assigning the value to the variable val1. Keeping the rev\_val variable as empty.
3. Using for loop, we reading the letter from backward of string. So the for loop start is length of the string and end is -1 ie upto the starting letter of the string.
4. Now assigning the reverse value of string to rev\_val variable.
5. Printing the reversed value.
6. Now checking the given string and reversed string is equal or not. If it is equal then printing it is palindrome else printing not palindrome.
7. Creating the object check for the class P\_drome. Getting input from the user. Passing this input to the class.
8. After passing the input, calling the function ispalindrome(). Only when function is called, step 2 to step 6 will execute and print the result.

**8) Using file handling – Write a text file in ur system with “hello world”**

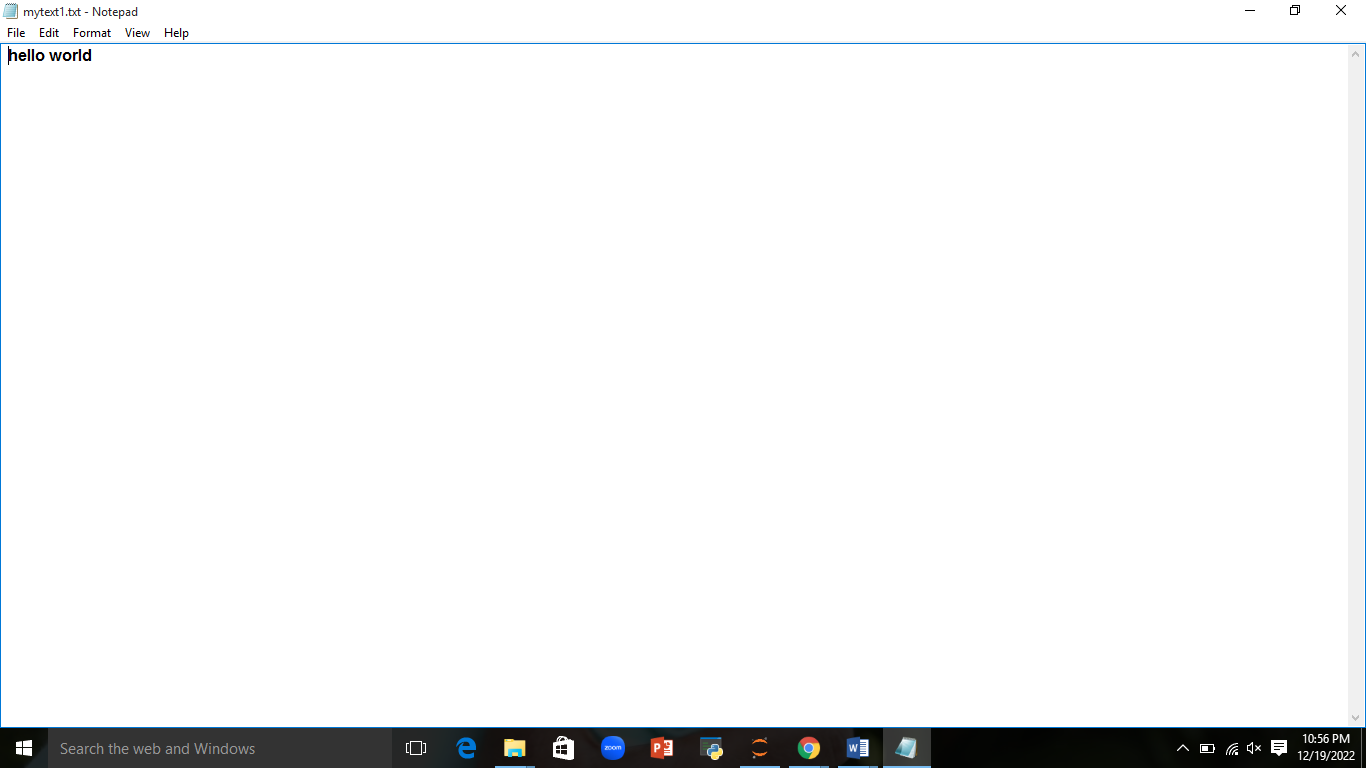
***Program :***

with open("mytext1.txt",'w') as f:

# Using write method in file handling, I write my text "hello world" to mytext1.txt file

f.write("hello world")

***Output :***

******

***Explanation :***

1. To write a file, we need to open it using with open. And in with open we are going to mention the filename and the mode. Here we are going to write a text file, so need to mention the filename in extension .txt and write mode. Here f is the file object created.
2. Using write function we going to write a string “hello world” to the text file created by us.
3. In the output, we created a text file name called mytext1.txt and we have write hello world in this file.

**9) Create option button using tkinter GUI in python.**

***Program :***

# For GUI in python, we are importing the package called tkinter

from tkinter import \*

def sel():

# To label on which option button we pressed, we are using label method

selection = "You selected the Subject " + str(var.get())

label.config(text = selection)

root = Tk()

var = IntVar()

# Option button which allows the user to choose (exactly) one of a predefined set of options

# Here we assigning the text to option button and calling the function sel() and also passing the value

R1 = Radiobutton(root, text="Maths", variable=var, value=1,

command=sel)

R1.pack( anchor = W )

R2 = Radiobutton(root, text="Science", variable=var, value=2,

command=sel)

R2.pack( anchor = W )

R3 = Radiobutton(root, text="Computer Science", variable=var, value=3,

command=sel)

R3.pack( anchor = W)

R4 = Radiobutton(root, text="Statistics", variable=var, value=4,

command=sel)

R4.pack( anchor = W)

R5 = Radiobutton(root, text="Financial Account", variable=var, value=5,

command=sel)

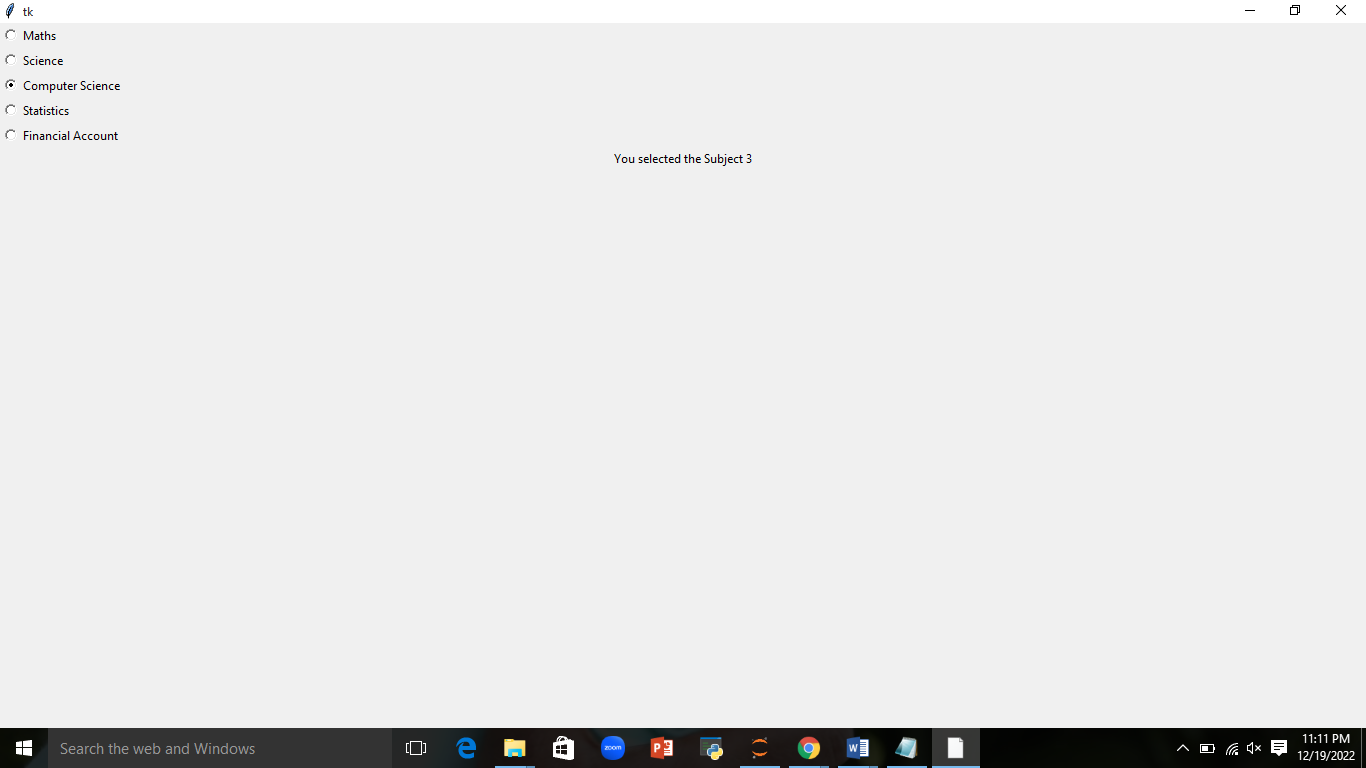
R5.pack( anchor = W)

label = Label(root)

label.pack()

root.mainloop()

***Output :***



***Explanation :***

1. For GUI in python, we are importing the package called tkinter.
2. We define the function sel(). Inside this we label the option button. For this we use the label. This label to be displayed when the user click the option button.
3. Outside the function, we assign the Tk() to root. Tk() means creating Tkinter window.
4. Assigning IntVar() to var variable. IntVar() able to store any integer value.
5. Radiobutton is otherwise called as option button, which allow the user to choose one of a predefined set of options.
6. In Radiobutton there is some of the arguments.

root – creating Tk window

text – the text to be displayed on each radiobutton

variable – it is the control variable which is used to keep track of the user’s choices.

value – the value of each radiobutton is assigned to the control variable when it is turned

on by the user.

command – this option is set to the procedure which must be called everytime when the

state of the radiobutton is changed.

1. Here we using the pack method. The pack method declares the position of widgets in relation to each other. Here the argument is anchor. The anchor represents the exact position of the text within the widget.
2. Next we call the Label where the text to be displayed when radiobutton is clicked. Next pack method.
3. Next we call the mainloop method. It is a method on the main window which we execute when we want to run one application. This method will loop forever, waiting for events from the user, until the user exits the program.

**10) Keep only numbers from the following string x = “ 89e9jcd^o38829@3%3,/mkl$w1”**

***Program :***

# We use Regular Expression regex here. So we need to import re

# Regular Expression is used for search pattern

import re

x = "89e9jcd^o38829@3%3,/mkl$w1"

# To take only number digits we use the meta character "\d+"

regex = "\d+"

# We use the function findall in Regex

# findall - return a list containing all matches

re.findall(regex,x)

***Output :***

['89', '9', '38829', '3', '3', '1']

***Explanation :***

1. Here we use Regular Expression regex here. So we need to import re.
2. Regular Expression is used for search pattern.
3. To take only number digits we use the meta character “\d+”.
4. Next we use the findall function in Regex. findall return a list containing all matches.

Hence the output gives only the numbers.