MODULE 3 – PYTHON ASSIGNMENT

NAME: B NISHANTH

ROLLNO: 7179KCTKCTKCTKCTKCTKCTKCT19BCS093

EMAIL: nishanth.19cs@kct.sc.in

**List creation:**

list=[]

list.append(20)

list.append(22)

list.append(20)

print(list)

**Insert an integer at specific position:**

#insert an integer at specified position

list.insert(1,21)

print(list)

**Print the list:**

#print the list

print("List: ")

print(list)

**Remove duplicate integer from the list:**

#remove duplicate integers from list

print("\nList after removing duplicate element:")

list.remove(20)

print(list)

**Insert an integer at the end of the list:**

#insert an interger at the end of the list

list.append(23)

print("\nList after adding integer at last:")

print(list)

**Sort the list:**

#sort the list

list.sort()

print("\nList in sorted order:")

print(list)

**Pop the last element from the list:**

#pop the last element from the list

list.pop()

print("\nList after pop operation:")

print(list)

**Reverse the list:**

#reverse the list

list.reverse()

print("\nList in reversed order:")

print(list)

**Write a calculator program in python?**

def add(P, Q):

return P + Q

def subtract(P, Q):

return P - Q

def multiply(P, Q):

return P \* Q

def divide(P, Q):

return P / Q

li=['a','b','c','d']

a=1

while(a==1):

print("\n\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*CALCULATOR\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*")

choice = input("\nPress\na for addition\nb for subtraction\nc for multiplication\nd for division\ne for exit\n")

if choice =='e':

print("Exiting!!!")

break

if choice not in li:

print("Please enter a valid choice")

continue

P = int (input ("Please enter the first number: "))

Q = int (input ("Please enter the second number: "))

if choice == 'a':

print (P, " + ", Q, " = ", add(P, Q))

elif choice == 'b':

print (P, " - ", Q, " = ", subtract(P, Q))

elif choice == 'c':

print (P, " \* ", Q, " = ", multiply(P, Q))

elif choice == 'd':

print (P, " / ", Q, " = ", divide(P, Q))

**Write a program to concatenate, reverse and slice a string?**

**• Concatenate:**

a="My Project name is"

b="Skill Job Recommender"

c=a+" "+b

print("Concatenation of two string:")

print(c)

**• Reverse:**

#Reverse a string

print("\nReversed String:")

reversedString=c[::-1]

print(reversedString)

**• Slice:**

#Slice a string

print("\nSliced String:"+c[26:41])

**Why is python a popular programming language?**

•  It uses a simplified syntax with an emphasis on natural language, for a much easier learning curve for beginners. And, because Python is free to use and is supported by an extremely large ecosystem of libraries and packages, it’s often the first-choice language for new developers.

• Python is often described as a general-purpose programming language. This means that unlike domain-specific languages which are designed only for certain application types, Python may be used to develop nearly any kind of application in any industry or field.

• Python’s ease of use, support, and flexibility have made it an essential tool for those who work with machine learning, cloud computing, and big data.

Python is particularly effective for analyzing and organizing data sets. In fact, for data science and analytics projects, Python is second only to R language in terms of popularity

• Python is efficient and reliable, allowing developers to create powerful applications with a minimum of effort. Completing coding projects is easy rather than time-consuming, and the results are able to stand toe to toe with applications created using more-demanding languages

• Python has become an automation standard across industries. In fact, even when working with other programming languages, developers will often write their automation scripts using Python.

**What are the other frameworks that can be used with python?**

• Django • TurboGears

• Pyramid • Flask

• Web2py • Sanic

• CherryPy

**What is the full form for WSGI?**

• WSGI stands for "Web Server Gateway Interface".

• The Web Server Gateway Interface (WSGI) is a standard interface between web server software and web applications written in Python. Having a standard interface makes it easy to use an application that supports WSGI with a number of different web servers.

• It is used to forward requests from a web server (such as Apache or NGINX) to a backend Python web application or framework.