**Assignment**

Q1. Why do we call Python as a general purpose and high-level programming language?

Ans-Python is an object-oriented, high-level programming language. Object-oriented means this language is based around objects (such as data) rather than functions, and high-level means it's easy for humans to understand.

Q2. Why is Python called a dynamically typed language?

Ans-In python, there is no need to define the types of variables. It allows to directly use the variables as its type-checking will be done during the execution of the program. The interpreter checks the program line-by-line and also examines the data type of the variable.

Q3. List some pros and cons of Python programming language?

Ans:-

**The pros of Python**

* Python is easy to learn and read.
* Python enhances productivity.
* Python has a vast collection of libraries.
* Python is free, open-source, and has a vibrant community.
* Python is a portable programming language.
* Python is an interpreted language.
* The cons of python programming language.

**The cons of Python**

* Speed Limitations. We have seen that Python code is executed line by line.
* Weak in Mobile Computing and Browsers. While it serves as an excellent server-side language, Python is much rarely seen on the client-side.
* Design Restrictions.
* Underdeveloped Database Access Layers.

Q4. In what all domains can we use Python?

Ans- Python is now one of the most popular and widely used programming languages in the world. Besides web and software development, Python is used for **data analytics, machine learning, Game and web development, Data science , os development ,scientific programming and even design**

Q5. What are variable and how can we declare them?

Ans- Variables are **the basic unit of storage in a programming language**. These variables consist of a data type, the variable name, and the value to be assigned to the variable.

**A variable declaration always contains two components: the type of the variable and its name**. Also, the location of the variable declaration, that is, where the declaration appears in relation to other code elements, determines the scope of the variable.

Q 6. How can we take an input from the user in Python?

Ans- Python allows for user input .That means we are able to ask the user for input.

Example:-

Name = input("Enter your name: ")

print("Your Name is :", Name)

Output:-

Your Name is:\_\_\_\_\_\_\_\_\_\_\_

Q7. What is the default datatype of the value that has been taken as an input using input() function?

Ans- The value returned by the input() is **a string**. Any data type can be used to convert the contents of an input. For example: The user can convert the value entered into an integer variable

Q8. What is type casting?

Ans-Type Casting is **the method to convert the variable data type into a certain data type in order to the operation required to be performed by users**.

Example:-

# Python program to demonstrate

# type Casting

# int variable

a = 5

# typecast to float

n = float(a)

print(n)

print(type(n))

Output:-

5.0

<class ’float’>

Q9. Can we take more than one input from the user using single input() function? If yes, how? If no, why?

Q10. What are keywords?

Ans- Keywords are predefined, reserved words used in Python programming that have special meanings to the compiler. We cannot use a keyword as a [variable](https://www.programiz.com/python-programming/variables-datatypes) name, [function](https://www.programiz.com/python-programming/function) name, or any other identifier. They are used to define the syntax and structure of the Python language. All the keywords except True ,False and None  are in lowercase . The list of all the keywords is given below.

False, await, elif , if, else, import, pass, None, break, exept, in, raise, True, class, finally, is, return, and, continue, for, lambda, try, as, def, from, nonlocal, while , ,assert, del, global, not, with, async ,or, yield.

Q11. Can we use keywords as a variable? Support your answer with reason.

Ans- Keywords are some predefined and reserved words in python that have special meanings. **The keyword cannot be used as an variable name**

**Reason**: Because Compiler will get confused (or it'll create ambiguity) whether to take it as a **keyword** or a **variable**. Compiler isn't a Human Brain, like Human Brain can differentiate Homophones (as they've same pronunciation)

**Example**: Suppose**while** keyword in python. If you use this **keyword** as **variable,**then compiler will get confused whether it is a variable or keyword.

Q12. What is indentation? What's the use of indentation in Python?

Ans- Indentation refers to the spaces at the beginning of a code line. Where in other programming languages the indentation in code is for readability only, the indentation in Python is very important. Python uses indentation **to indicate a block of code**.

 It is mainly used for code inside looping statements, control structures, functions etc. as good intended code is easy to maintain and is working. It makes the code more readable and easy to understand.

Q13. How can we throw some output in Python?

Ans:- The basic way to do output is the **print statement**.

Q14. What are operators in Python?

Ans- Operators are used to perform operations on variables and values

Python divides the operators in the following groups:

* Arithmetic operators (Example +,-,/,\*,%,\*\*)
* Assignment operators (Example =,++,=+,\*=,%,etc)
* Comparison operators (Example =, !+, <, >, <= ,>+ )
* Logical operators (Example and, or , not)
* Identity operators (Example is ,is not)
* Boolean operators (Example True, False)
* Bitwise operators (Example &,|,^,~,<<,>>)

Q15. What is difference between / and // operators?

Ans:- AND and OR are defined as logical operators. Both require two operands which may evaluate to true or false. **The and operator returns True only if both operands are True.** **The or operator returns True if either operand is true**.

Q16. Write a code that gives following as an output.

```

iNeuroniNeuroniNeuroniNeuron

```

Ans:- I =( " iNeuroniNeuroniNeuroniNeuron " )

print(I)

Q17. Write a code to take a number as an input from the user and check if the number is odd or even.

Ans:- num =int(input("Enter any number to check whether it is odd or even:"))

if (num % 2) == 0:

print("The he number is even")

else:

print ("The provided number is odd")

output :-

Enter any number to check whether it is odd or even:

Q18. What are boolean operator?

Ans:- It’s used to represent the truth value of an expression. For example, the expression 1 <= 2 is True, while the expression 0 == 1 is False.

The Boolean has to possible value

1.True

2. False

Q19. What will the output of the following?

```

1 or 0

0 and 0

True and False and True

1 or 0 or 0

```

Ans:-

```

1 or 0 = 1

0 and 0 =0

True and False and True = False

1 or 0 or 0 = 1

```

Q20. What are conditional statements in Python?

Ans:- Python also has some predefined conditional statements. A conditional statement as the name suggests itself, is used to handle conditions in your program. These statements guide the program while making decisions based on the conditions encountered by the program.

Conditional statement are “if……..elif………else”

Q21. What is use of 'if', 'elif' and 'else' keywords?

Ans:- if… elif…else are conditional statements that provide you with the decision making that is required when you want to execute code based on a particular condition. The if… elif…else statement used in Python **helps automate that decision making process**.

Q22. Write a code to take the age of person as an input and if age >= 18 display "I can vote". If age is < 18 display "I can't vote".

Ans:-

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

if age>= 18:

print("I can vote")

else:

print("I can't vote")

Q23. Write a code that displays the sum of all the even numbers from the given list.

```

numbers = [12, 75, 150, 180, 145, 525, 50]

```

Ans:-numbers = [12, 75, 150, 180, 145, 525, 50]

print(sum([i for i in numbers if i % 2 == 0]))

Output:- 392

Q24. Write a code to take 3 numbers as an input from the user and display the greatest no as output.

Ans:-

a = int(input('Enter first number : '))

b = int(input('Enter second number : '))

c = int(input('Enter third number : '))

largest = 0

if a > b and a > c:

largest = a

if b > a and b > c:

largest = b

if c > a and c > b:

largest = c

print(largest, "is the largest of three numbers.")

Q25. Write a program to display only those numbers from a list that satisfy the following conditions

- The number must be divisible by five

- If the number is greater than 150, then skip it and move to the next number

- If the number is greater than 500, then stop the loop

```

numbers = [12, 75, 150, 180, 145, 525, 50]

```

Ans:-

a =[12,75,150,180,145,525,50]

b =[]

for i in a:

if i >500:

break

if i>150:

continue

if i % 5==0:

b.append(i)

print(b)

Q26. What is a string? How can we declare string in Python?

Ans:- A string is **a sequence of characters**. For example, "hello" is a string containing a sequence of characters 'h' , 'e' , 'l' , 'l' , and 'o' . We use single quotes or double quotes to represent a string in Python.

Q27. How can we access the string using its index?

Ans:- You can access the characters in a string by referring to its index number inside square brackets [] .

#Python progran to check can we access the string using its index

str = "we access the string using its index"

str\_1 = str.index("its")

#print

print(str\_1)

Output

27

Q28. Write a code to get the desired output of the following

```

string = "Big Data iNeuron"

desired\_output = "iNeuron"

```

Ans:- string = "Big Data iNeuron"

# print(len(string))

print(string[9:16])

Output:- iNeuron

Q29. Write a code to get the desired output of the following

```

string = "Big Data iNeuron"

desired\_output = "norueNi"

```

Ans:-

#Python program  to get the desired output of the following

string = "Big Data iNeuron"

# print(len(string))

str\_1 = string[9:16]

#Reverse the string

print(str\_1[::-1])

Output:

norueNi

Q30. Resverse the string given in the above question.

Ans:- string = "Big Data iNeuron"

print(string[ ::-1])

Output:-

norueNi ataD giB

Q31. How can you delete entire string at once?

Ans:- Python will not allow deleting a particular character in a string. Where as you can remove the entire string variable using the del command.

Q32. What is escape sequence?

Ans:- An escape sequence is a sequence of characters that, when used inside a character or string, does not represent itself but is converted into another character or series of characters.

Example

|  |  |
| --- | --- |
| \’ | Single quotation |
| \\ | Backslash |
| \n | New Line |
| \r | Carriage return |
| \t | Tab |

Q33. How can you print the below string?

```

'iNeuron's Big Data Course'

Ans:- print("'iNeuron's Big Data Course'")

```

Q34. What is a list in Python?

Ans:- Lists are used to store multiple items in a single variable.

Lists are created using square brackets

For ex:- [iNeuron, python ,java , php , C , C++ ]

Q35. How can you create a list in Python?

Ans:- Put all of the items (elements) inside square brackets [], separated by commas. It can include an unlimited number of elements of various data types (integer, float, string, etc.). Python Lists can also be created using the built-in list() method.

Ex:- A=['java',100,70.6,true]

Q36. How can we access the elements in a list?

Ans:- In python, each item in a list is associated with a number. The number is known as a list index.

We can access elements of an list using the index number **(0, 1, 2 …)**. For example

languages = ["Python", "Big data", "C++"]

# access item at index 0

print(languages[0]) # Python

# access item at index 2

print(languages[2]) # C++

|  |  |  |
| --- | --- | --- |
| “Python” | “Big data” | “C++” |

**index** 0 1 2

The list index always starts with 0. Hence, the first element of a list is present at index 0, not 1

Q37. Write a code to access the word "iNeuron" from the given list.

```

lst = [1,2,3,"Hi",[45,54, "iNeuron"], "Big Data"]

```

Ans:-

#Python program  to access the word "iNeuron" from the given list

lst = [1,2,3,"Hi",[45,54, "iNeuron"], "Big Data"]

print(lst[4][2])

Output:-

iNeuron

Q38. Take a list as an input from the user and find the length of the list.

Ans:

#Python program to make a list as an input from the user and find the length

Of the list

#Creating a list (empty)

lit = []

# number of elements as input

n = int(input("enter the number of element in list "))

#itrating the range

for i in range(0 , n):

   lit\_1 = int(input())

   list.append(lit\_1)

print("Input form user : ",lit)

print("lenght of list is : " ,len(lit))

Output:

Input form user : [1, 2, 3, 4, 5, 6, 7, 8]

lenght of list is : 8

Q39. Add the word "Big" in the 3rd index of the given list.

```

lst = ["Welcome", "to", "Data", "course"]

```

Ans:- lst = ["Welcome", "to", "Data", "course"]

#Using insert method

lst.insert(2,"Big")

print(lst)

Output

['Welcome', 'to', 'Big', 'Data', 'course']

Q40. What is a tuple? How is it different from list?

Ans:- A tuple is an immutable object, which means it cannot be changed, and we use it to represent fixed collections of items.  a tuple is represented as a comma-delimited list of the elements, enclosed in parentheses. For example, "(5, 9, 11, 3, 22, 14)" .

The primary difference between tuples and lists is that tuples are immutable as opposed to lists which are mutable. Therefore, it is possible to change a list but not a tuple. The contents of a tuple cannot change once they have been created in Python due to the immutability of tuples.

Q41. How can you create a tuple in Python?

Ans:- A  tuple is created by placing all the items (elements) inside parentheses () , separated by commas. A tuple can have any number of items and they may be of different types (integer, float, list, string, etc.).

Ex:-

# Empty tuple

my\_tuple = ()

print(my\_tuple)

# tuple with mixed datatypes

my\_tuple = (1, "Hello", 3.4)

print(my\_tuple)

Output:-

()

(1, “hello”, 3.4)

Q42. Create a tuple and try to add your name in the tuple. Are you able to do it? Support your answer with reason.

Ans:- Once a tuple is created, you cannot change its values. Tuples are unchangeable, or immutable as it also is called.

Q43. Can two tuple be appended. If yes, write a code for it. If not, why?

Ans:- Tuple is immutable, although you can use the + operator to concatenate several tuples. The old object is still present at this point, and a new object is created

Ex:-

a=(33,4,55,21,54)

a\_append = a + (1,2,3,5)

print(a\_append)

Output:-

(33, 4, 55, 21, 54, 1, 2, 3, 5)

Q44. Take a tuple as an input and print the count of elements in it.

Ans

x = input('Enter the some input : ')

x = tuple(int(a) for a in x. split(","))

print(x.count(5))

Output

Enter the some input :1,2,3,4,5,6,7,8,5

2

Q45. What are sets in Python?

Ans:- A Set is an unordered collection data type that is iterable, mutable, and has no duplicate elements.

Set are represented by { } (values enclosed in curly braces)

Q46. How can you create a set?

Ans:- We create sets by **placing all the elements inside curly braces {} , separated by comma**. A set can have any number of items and they may be of different types (integer, float, tuple, string etc.).

thisset = {"apple", "banana", "cherry", "apple"}

print(thisset)

output:-

{'banana', 'apple', 'cherry'}

Q47. Create a set and add "iNeuron" in your set.

Ans:-

#Python program to Create a set and add "iNeuron" in your set.

#Set of group

a = { "java" , "c" , "python" , "php"}

#Adding "iNeuron"

a.add("iNeuron")

print("Output the given Question is : " , a)

#Again adding iNeuron

a.add("iNeuron")

print("Output the given Question is : " , a)

Q48. Try to add multiple values using add() function.

Ans:-

#Python program to add multiple values using add() function

set\_1 = {1,2,3,4,5,6,7}

#Adding valve in set

set\_1.add(8,0)

print(set\_1)

Output:-

set\_1.add(8,0)

TypeError: set.add() takes exactly one argument (2 given)

Q49. How is update() different from add()?

Ans:-

* add() is intended for a single element , while . update() is for the introduction of other sets
* Use add() function to add a single element. Whereas use update() function to add multiple elements to the set.
* add() is faster than update().
* add () accepts immutable parameters only. Whereas accepts iterable sequences.
* add() accepts a single parameter, whereas update() can accept multiple sequences.

Q50. What is clear() in sets?

Ans:- Clear() method is used to remove all the elements from a Set. Using the clear() method only clears all the element from the set and not deletes the set.

 set of vowels

vowels = {'a', 'e', 'i', 'o', 'u'}

print('Vowels (before clear):', vowels)

# clear vowels

vowels.clear()

print('Vowels (after clear):', vowels)

Output:- Vowels (before clear): {'o', 'a', 'e', 'u', 'i'}

Vowels (after clear): set()

Q51. What is frozen set?

Ans:- Frozenset () Method **creates an immutable Set object from an iterable**. It is a built-in Python function. As it is a set object therefore we cannot have duplicate values in the frozenset.

# tuple of letters

letters = ('m', 'r', 'o', 't', 's')

fSet = frozenset(letters)

print('Frozen set is:', fSet)

print('Empty frozen set is:', frozenset())

Q52. How is frozen set different from set?

Ans:- Python has two built-in types for sets: set and frozenset . A set is a mutable object while frozenset provides an immutable implementation.

Q53. What is union() in sets? Explain via code.

Ans:- **Union** of two given sets is the set which contains all the elements of both the sets. The union of two given sets A and B is a set which consists of all the elements of A and all the elements of B such that no element is repeated.

set1 = {2, 4, 5, 6}

set2 = {4, 6, 7, 8}

set3 = {7, 8, 9, 10,11}

# union of two sets

print("set1 U set2 : ", set1.union(set2))

# union of three sets

print("set1 U set2 U set3 :", set1.union(set2, set3))

Output:-

set1 U set2 : {2, 4, 5, 6, 7, 8}

set1 U set2 U set3 : {2, 4, 5, 6, 7, 8, 9, 10, 11}

Q54. What is intersection() in sets? Explain via code.

An:- The intersection of two given sets is the largest set, which contains all the elements that are **common** to both sets. The intersection of two given sets A and B is a set which consists of all the elements which are common to both A and B.

set1 = {2, 4, 5, 6}

set2 = {4, 6, 7, 8}

set3 = {4, 6, 8}

# intersection of two sets

print("set1 intersection set2 : ",

      set1.intersection(set2))

# intersection of three sets

print("set1 intersection set2 intersection set3 :",

      set1.intersection(set2, set3))

Output:-

set1 intersection set2 : {4, 6}

set1 intersection set2 intersection set3 : {4, 6}

Q55. What is dictionary ibn Python?

Ans:- Python dictionary is an ordered collection of items. It stores elements in **key/value** pairs. Here, **keys** are unique identifiers that are associated with each **value**.

Q56. How is dictionary different from all other data structures.

Ans:-Because of a dictionary has a set of keys and each key has a single associated value.

Q57. How can we declare a dictionary in Python?

Ans:- A Dictionary in python is declared by **enclosing a comma-separated list of key-value pairs using curly braces({})**.

mydict = {

    "aditya" :"I m working in hyundai ",

    "company" :"harsh hyuandai",

}

print(mydict['aditya'])

print(mydict['company'])

Output:-

I m working in a Hyundai

Harsh hyuandai

Q58. What will the output of the following?

```

var = {}

print(type(var))

```

Ans:- var = {}

print(type(var))

Output:-

<class 'dict'>

Q59. How can we add an element in a dictionary?

Ans:- We add a new element to the dictionary by using a new key as a subscript and assigning it a value

#Python program to create we add an element in a dictionary

player = { "Ms dhoni" : 7, "Virat" : 18, "KL rahul" : 1, "Rohit" :45,}

print(player)

player["Suryakumar"]=63

print(player)

Output

{'Ms dhoni': 7, 'Virat': 18, 'KL rahul': 1, 'Rohit': 45}

{'Ms dhoni': 7, 'Virat': 18, 'KL rahul': 1, 'Rohit': 45, 'Suryakumar': 63}

Q60. Create a dictionary and access all the values in that dictionary.

Ans:-

Mydict ={

    "fast ":"in a quick manner",

    "Aditya" :"coder",

    "mark" : [55,78,90,99,44],

    "other"{'lucky':' paaji '},

    4:9

}

print(mydict.values()) # print the values of dict

Output:-

dict\_values(['in a quick manner', 'coder', [55, 78, 90, 99, 44], {'lucky': ' paaji '}, 9])

Q61. Create a nested dictionary and access all the element in the inner dictionary.

Ans. A dictionary can contain dictionaries, this is called nested dictionaries.

#Python program to  creat nested dictionary and access all the element in

# the inner dictionary

my\_dict = {

    "Mechanical" : 5,

    "Software" :7,

    "Electrical" : 3,

    "Chemical" : 9,

    "automobile" : 1

}

for i in my\_dict.items():

    print(i[0], "has value : ", i[1] )

Output

Mechanical has value : 5

Software has value : 7

Electrical has value : 3

Chemical has value : 9

automobile has value : 1

Q62. What is the use of get() function?

Ans:- The get() method returns the value of the item with specified key

car = {

  "brand": "Ford",

  "model": "Mustang",

  "year": 1964,

  "award": "Best in design"

}

a = car.get("award")

print(a)

Output

Best in design

Q63. What is the use of items() function?

Ans:- Then item() menthod return a view object. The view object contain the key value pair of the dictionary, a tuples in list.

player = {

  "Ms dhoni ": "cricketer",

  "country": "India",

  "world cup": 1983

}

x = player.items()

player["world cup"] = 2011

print(x)

Output:-

dict\_items([('Ms dhoni', 'cricketer'), ('country', 'India'), ('world cup', 2011)])

Q64. What is the use of pop() function?

Ans:- A pop() function is used to remove the element at the specified position from dictionary.

Example

car = {

  "brand": "Ford",

  "model": "Mustang",

  "year": 1964,

  "award": "Best in design"

}

car.pop("model")

print(car)

Output:-

{'brand': 'Ford', 'year': 1964, 'award': 'Best in design'}

Q65. What is the use of popitems() function?

Ans:- The popitem() method removes the item that was last inserted into the dictionary.

player = {

  "Ms dhoni": "cricketer",

  "country": "India",

  "world cup": 2011

}

player.popitem()

print(player)

Output:-

{'Ms dhoni': 'cricketer', 'country': 'India'}

Q66. What is the use of keys() function?

Ans:- The key() method return a view object. The view object contain the keys of the dictionary , as al list. Example

player = {

  "Ms dhoni": "cricketer",

  "country": "India",

  "world cup": 2011

}

x = player.keys()

print(x)

Output:-

dict\_keys(['Ms dhoni', 'country', 'world cup'])

Q67. What is the use of values() function?

Ans :- The values() method return a view object. The view object contain the keys of the dictionary, as al list. Example

player = {

  "Ms dhoni": "cricketer",

  "country": "India",

  "world cup": 2011

}

x = player.values()

print(x)

Output:-

dict\_values(['cricketer', 'India',2011])

Q68. What are loops in Python?

Ans:- Repeatingsomething over and over until a particular condition is satisfied is called loop.

Q69. How many type of loop are there in Python?

Ans:- There are 3 types of loop in python

1. For loop

2. While loop

3. Nested loop

Q70. What is the difference between for and while loops?

Ans:- In the case of a for loop, the initialization is done once at the start, so there is no need to initialize it again. But in the case of a while loop, we require to initialize the loop manually by taking a variable that is further modified (incremented, decremented, multiplied, etc.) as per our requirement.

Q71. What is the use of continue statement?

Ans The continue statement in Python is used to skip the remaining code inside a loop for the current iteration only.

fruits = ["apple", "banana", "cherry", "mango"]

for x in fruits:

  if x == "mango":

    continue

  print(x)

Output :-

Apple

Banana

cherry

Q72. What is the use of break statement?

Ans:-Break statement in Python is a loop control statement. It is used to control the sequence of the loop. Suppose you want to terminate a loop and skip to the next code after the loop; break will help you do that. A typical scenario of using the Break in Python is when an external condition triggers the loop's termination.

fruits = ["apple", "banana", "cherry" , "mango"]

for x in fruits:

  if x == "banana":

    break

  print(x)

Output:-

apple

Q73. What is the use of pass statement?

Ans:- A pass statement signals to a loop that there is “no code to execute here.” It's a placeholder for future code.

for x in [0, 1, 2, 3, 4, 5, 6, 7,8]:

  pass

Output:-

Q74. What is the use of range() function?

Ans:- The range() function return a sequence of numbers, starting from 0 by default and increment by 1 by default and stop before a specific number Example:-

for x in range(1,10):

 print(x)

Output:

1

2

3

4

5

6

7

8

9

Q75. How can you loop over a dictionary?

Ans: By using a for loop. When looping through a dictionary, the return value are the keys of the dictionary, but there are methods to return the values as well.

### Coding problems

Q76. Write a Python program to find the factorial of a given number.

Ans: #Python program to find the factorial of a given number

number = int(input("Enter a number to find factorial: "))

factorial = 1

#condition check

if number < 0:

   print(" Factorial does not exist for negative numbers")

elif number == 0:

   print("The factorial of 0 is 1")

else:

   for i in range(1,number + 1):

       factorial = factorial\*i

   print("The factorial of", number ,"is" ,factorial )

Output

Enter a number to find factorial: 5

The factorial of 5 is 120

Q77. Write a Python program to calculate the simple interest. Formula to calculate simple interest is SI = (P\*R\*T)/100

Ans :-

#Python program to calculate the simple interest

# Enter the amount

P = int(input("Enter principle amount : "))

# Enter the number of time

T = int(input("Enter the number of year : "))

# Enter the rate of interest

R = float(input("Enter rate of interest : "))

# Calculate the simple interest

SI = (P \* T \* R)/100

print("The simple interest is:", SI)

Q78. Write a Python program to calculate the compound interest. Formula of compound interest is A = P(1+ R/100)^t.

Ans :-

#Python program to calculate the compound interest

#Enter the amount

P = int(input("Enter principle amount : "))

# Enter the number of time in year

T = int(input("Enter the number of year : "))

# Enter the rate of interest

R = float(input("Enter rate of interest : "))

# Calculate the compound Amount

A = P\*(1+ R/100)\*\*T

# Calculate the compound interest

compound\_interest = (A - P)

print("Compound Amount " , A)

print("The compound\_interest is:", compound\_interest )

Q79. Write a Python program to check if a number is prime or not.

Ans:-

# Program to check if a number is prime or not

num = int(input("Enter a number: ")) # Input from the user

if num > 1:   # if no. is greater than 1

   # Check if factor exist

   for i in range(2,num):

       if (num % i) == 0:

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

           break

   else:

       print(num,"is a prime number")

# Else if the input number is less than or equal to 1

else:

   print(num,"is not a prime number")

Q80. Write a Python program to check Armstrong Number.

Ans:

#Python program to check Armstrong Number.

number = int(input("Enter a number to check to Armstrong no or not : "))

#Find the sum cube each digits

temp = number

add\_sum = 0

while temp != 0:

    k = temp % 10

    add\_sum += k\*k\*k

    temp = temp//10

if add\_sum == number:

    print("Given number is a Armstrong Number")

else:

    print("Given number is not an Armstrong Number")

Output:

Enter a number to check to Armstrong no or not : 153

Given number is a Armstrong Number

Enter a number to check to Armstrong no or not : 144

Given number is not an Armstrong Number

Q81. Write a Python program to find the n-th Fibonacci Number.

Ans:-

#Python program to find the n-th Fibonacci Number

num = int(input("Enter the how many numbers : "))

#first two term

n1, n2 = 0 , 1

count = 0

if num <= 0:

       print("Enter the positive integer value")

#if there is only one term, return n1

elif num == 1:

   print("Sequence of fibonacci number upto", num ,":")

   print(n1)

#Sequence of fibonacci number

else:

   print("Sequence of fibonacci number:")

   while count < num:

       print(n1)

       nth = n1 + n2

#update the value

       n1 = n2

       n2 = nth

       count += 1

Output:

Enter the how many numbers: 0

Enter the positive integer value

Enter the how many numbers: 1

Sequence of fibonacci number upto 1:

0

Enter the how many numbers: 7

Sequence of fibonacci number:

0

1

1

2

3

5

8

Q82. Write a Python program to interchange the first and last element in a list.

Ans:-

#Python program to interchange the first and last element in a list

my\_list = [1 , 2 , 3 , 4 , 5 , 6 , 7 , 8 , 9 , 10 ]

print("Intial list : " , my\_list)

#Find the lenght of list

length = len(my\_list) #Python program to get all combinations of 2 tuples

from itertools import chain, product

#tuples

test\_tuple1 = (7, 2)

test\_tuple2 = (7, 8)

print("The tuple 1 : " + str(test\_tuple1))

print("The tuple 2 : " + str(test\_tuple2))

# All pair combinations of 2 tuples

result = list(chain(product(test\_tuple1, test\_tuple2), product(test\_tuple2, test\_tuple1)))

#Result

print("The resultant tuple : " + str(result))

# Swapping first and last element

A = my\_list[0]

my\_list[0] = my\_list[length - 1]

my\_list[length - 1] = A

print("List after swapping : " , my\_list)

Q83. Write a Python program to swap two elements in a list.

Ans:-

#Python program to swap two elements in a list.

car = ["hyundai" , "tata" , "maruti" , "toyota" ]

print("List of car before swap : " , car)

#Swap the to index 0 to 2 and swap 3 to 1

Car[0],car[2] = car[2],car[0]

car[1],car[3] = car[3],car[1]

print("List of car after swap : " , car)

Output

List of car before swap : ['hyundai', 'tata', 'maruti', 'toyota']

List of car after swap : ['maruti', 'toyota', 'hyundai', 'tata']

Q84. Write a Python program to find N largest element from a list.

Ans :-

#Python program to find N largest element from a list

list = [18 , 45 , 10 , 33 , 100 , 91 , 1818 , 69]

print("List" , list)

n = 1  #the value of n

list.sort() #sort the list

print("Largest integer from the given list is " , list[-n:])

Output:

List [18, 45, 10, 33, 100, 91, 1818, 69]

Largest integer from the given list is [1818]

Q85. Write a Python program to find cumulative sum of a list.

Ans:-

#Python program to find cumulative sum of a list.

def number\_cumulative\_sum(number\_list): #use define function

  return [sum(number\_list[:i+1]) for i in range(len(number\_list))]

print(number\_cumulative\_sum([10, 20, 30, 40, 50, 60, 70 , 80 , 90 , 100 ,200 , 250]))

Output

[10, 30, 60, 100, 150, 210, 280, 360, 450, 550, 750, 1000]

Q86. Write a Python program to check if a string is palindrome or not.

Ans:-

#Python program to check if a string is palindrome or not

my\_string = input ("Enter the string to check if it is a palindrome: ")

# make it suitable for caseless comparison

my\_string = my\_string.casefold()

# reverse the string

rev\_str = reversed(my\_string)

# check if the string is equal to its reverse

if list(my\_string) == list(rev\_str):

   print("The string is a palindrome.")

else:

   print("The string is not a palindrome.")

Output

Enter the string to check if it is a palindrome: adjjjjdA

The string is a palindrome.

Q87. Write a Python program to remove i'th element from a string.

Ans:

#Python program to remove i'th element from a string

string\_1="iNeuron"

i=1

print("Before removeing i'th element from a string :",string\_1)

# Initialise new string

str= ""

#Removing character

for k in range(len(string\_1)):

    if k!=i-1:

        str=str+string\_1[k]

# After removal of character

print("After removeing i'th element from a string :",str)

Output

Before removeing i'th element from a string : iNeuron

After removeing i'th element from a string : Neuron

Q88. Write a Python program to check if a substring is present in a given string.

Ans:-

#Python program to check if a substring is present in a given string

str = "Live as if you were to die tomorrow. learn as if you were to live forever"

#check condition

if "learn" in str:

    print("Substring is present in a given string")

else :

    print("Substring is not present in a given string")

Output

Substring is present in a given string

Q89. Write a Python program to find words which are greater than given length k.

Ans:-

#Python program to find words which are greater than given length k

str\_1 = input("Enter the string : ")

k=4

length\_of\_string = []

#Finding words with length greater than k

words = str\_1.split(" ")

for i in words:

    if len(i) > k:

        length\_of\_string.append(i)

print("Words which are greater than given length k is: " ,k )

print(length\_of\_string)

output:-

Enter the string : Live as if you were to die tomorrow. learn as if you were to live forever

Words which are greater than given length k is: 4

['tomorrow.', 'learn', 'forever']

Q90. Write a Python program to extract unquire dictionary values.

Ans:-

#Python program to extract unquire dictionary values.

my\_dict = {

    "Mechanical" : 5,

    "Computer science" :7,

    "Electrical" : 3,

    "Chemical" : 9,

    "automobile" : 1

}

uniqueValues = list({val for val in my\_dict.values() })

#Extracting unique values using set comprehension

print("Dictionary :-", end = " ")

print(my\_dict) #Print dictionary

print("Uniques\_values :-", end = " ")

print(uniqueValues) #printing uniques value

Output

Dictionary :- {'Mechanical': 5, 'Computer science': 7, 'Electrical': 3, 'Chemical': 9, 'automobile': 1}

Uniques\_values :- [1, 3, 5, 7, 9]

Q91. Write a Python program to merge two dictionary .

Ans:

my\_dict = {

    "Mechanical" : 5,

    "Software" :7,

    "Electrical" : 3,

    "Chemical" : 9,

    "automobile" : 1

}

my\_dict2 = {

    "Chemical" : 8,

    "Civil" : 4,

    "Industrial" : 2,

    "Agriculture" : 6

}

#This is return none

print(Merge(my\_dict,my\_dict2))

#Merge both dictionary

print(my\_dict2)

Output

None

{'Chemical': 9, 'Civil': 4, 'Industrial': 2, 'Agriculture': 6, 'Mechanical': 5, 'Software': 7, 'Electrical': 3, 'automobile': 1}

Q92. Write a Python program to convert a list of tuples into dictionary.

```

Input : [('Sachin', 10), ('MSD', 7), ('Kohli', 18), ('Rohit', 45)]

Output : {'Sachin': 10, 'MSD': 7, 'Kohli': 18, 'Rohit': 45}

```

Ans

# Python program to convert a list of tuples into dictionary.

list\_of\_tuples = [('Sachin', 10), ('MSD', 7), ('Kohli', 18), ('Rohit', 45)]

a = dict(list\_of\_tuples)

print(a)

Output

{'Sachin': 10, 'MSD': 7, 'Kohli': 18, 'Rohit': 45}

Q93. Write a Python program to create a list of tuples from given list having number and its cube in each tuple.

```

Input: list = [9, 5, 6]

Output: [(9, 729), (5, 125), (6, 216)]

```

Ans:-

#Python program to create a list of tuples from given list having number

#And its cube in each tuple

def cubeoflist(list):

#List of tuples

    Result = [(num, num\*\*3) for num in list]

    return result

#Given list

list = [9, 5, 6]

print("list of tuples from given list : ")

print(list)

#Print the result

print ("The final output of given list of tuple : ")

print(cubeoflist(list))

Q94. Write a Python program to get all combinations of 2 tuples.

```

Input : test\_tuple1 = (7, 2), test\_tuple2 = (7, 8)

Output : [(7, 7), (7, 8), (2, 7), (2, 8), (7, 7), (7, 2), (8, 7), (8, 2)]

```

Ans:

#Python program to get all combinations of 2 tuples

from itertools import chain, product

# tuples

test\_tuple1 = (7, 2)

test\_tuple2 = (7, 8)

print("The tuple 1: " + str(test\_tuple1))

print("The tuple 2: " + str(test\_tuple2))

# All pair combinations of 2 tuples

result = list(chain(product(test\_tuple1, test\_tuple2), product(test\_tuple2, test\_tuple1)))

#Result

print("The resultant tuple: " + str(result))

Output

The tuple 1: (7, 2)

The tuple 2: (7, 8)

The resultant tuple: [(7, 7), (7, 8), (2, 7), (2, 8), (7, 7), (7, 2), (8, 7), (8, 2)]

Q95. Write a Python program to sort a list of tuples by second item.

```

Input : [('for', 24), ('Geeks', 8), ('Geeks', 30)]

Output : [('Geeks', 8), ('for', 24), ('Geeks', 30)]

```

Ans:-

#Python program to sort a list of tuples by second item

#Function to sort

def tuple\_sort(my\_tuple):

#Sort the tuples by the second item using the lambda function

   return(sorted(my\_tuple, key = lambda x: x[1]))

#list of tuples

my\_tuple =[('for', 24), ('Geeks', 8), ('Geeks', 30)]

print("The list of tuple is : ")

print(my\_tuple)

print("After sorting, the list of tuple  : ")

print(tuple\_sort(my\_tuple))

Q96. Write a python program to print below pattern.

```

\*

\* \*

\* \* \*

\* \* \* \*

\* \* \* \* \*

```

Ans:-

#Python program to print half pyramid pattern.

n = 5

a = 1; b = 0

#using while loop and check the condition become false. if it's true then enter into

#The loop and print the pattern

while (a<=n):

        while(b<=a-1):

                print(" \* ", end = "")

                b+=1

        print("\r") #print next line for row

        b=0; a+=1

Q97. Write a python program to print below pattern.

```

\*

\*\*

\*\*\*

\*\*\*\*

\*\*\*\*\*

```

n = 5

for i in range(n):

    print( " "\*(n-i-1) + "\*"\*(2\*i+1) )

Q98. Write a python program to print below pattern.

```

\*

\* \*

\* \* \*

\* \* \* \*

\* \* \* \* \*

```

Ans:-

#Python program to print pyramid by using \*

rows = 5

for i in range(0, rows):

    for j in range(0,rows-i-1):

        print(end=" ")

    for j in  range(0, i+1):

        print("\*", end=" ")

    print("\r")

Q99. Write a python program to print below pattern.

```

1

1 2

1 2 3

1 2 3 4

1 2 3 4 5

```

Ans:-

#Python program to print half pyramid by using number below pattern.

#Number of row

r = 5

#For using for loop and range function

for i in range(r):

        for j in range(i+1):

                print(j+1 , end= " ")

        print("\n")

Q100. Write a python program to print below pattern.

```

A

B B

C C C

D D D D

E E E E E

```

Ans:-

#Python program to print half pyramid by using alphabet.

#Number of rows

rows = 5

#ASCII is the number 65 corresponds to the capital letter 'A'

ascii\_value = 65

for i in range(rows):

    for j in range(i+1):

#chr()function returns the character that represents the specified Unicode.

        a = chr(ascii\_value)

        print(a, end= " ")

    ascii\_value += 1

    print("\n")