## Assignment Part-1

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

* Python is called general purpose because it is applied to real world class problems and called high-level because it’s easy for humans to understand

Q2. Why is Python called a dynamically typed language?

* It doesn’t know the type of variable until the code is run. So, type of the variable is determined at runtime
* Also, variables can be used without declaring them explicitly

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

* Simple and Easy
* Easy to learn
* Easy to code
* Readable
* Object-oriented and procedure-oriented
* Free and Open Source
* Less Coding
* Portable
* Interpreted
* Extensive Libraries
* Automatic memory management - Python supports automatic memory management which means, the memory is cleared and released automatically. We do not have to worry about clearing the memory.
* Enterprise Application integration because it works on different platforms like Raspberry Pi, Linux, Windows, Mac, etc. This makes python as a portable language and cross platform language.

 Disadvantages -

* Speed limitations – Python execution results are a bit slow as the code executed line by line.
* Large memory consumption – Python is not suitable to use under the limited memory restrictions. Python structures need more memory space.
* Not suitable for Mobile – Python is designed as server-side scripting language and very rarely used at the client-side. And, also very rarely used in mobile based applications.
* Design restrictions – Python is “dynamic-typed” language as it does not require a variable to be declared before using it. The programmer job is easier with this however it may cause run time errors due to invalid data movements.
* Weak database access layers – Database layers are less comparative when compared with standard database layers ODBC and JDBC.
* Error detection codes – Python executed through an interpreter instead of a compiler. So, the errors and bug cannot be detected during the program execution. Developer should have code an exception handling for catching the errors.
* Difficult to test – All the errors are caught during the runtime including syntax errors. We need to clear up all the errors to get the output. So testing is a bit difficult in Python specially when integrated with other systems.

Q4. In what all domains can we use Python?

* Machine Learning / Artificial Intelligence
* Web Development
* Desktop GUI
* Data Analyis and Data Visualization
* Game development
* Embedded systems
* Mobile App Development

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

* A symbolic name to an object or to the memory location which is storing some value

Q6. How can we take an input from the user in Python?

* Using the input() function

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

* string

Q8. What is type casting?

* Conversion of one data type to other data type in order for the operation to be successful

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

* We cannot take directly but we can decide on delimiter and use split function with input to separate the input values.

Q10. What are keywords?

* Special reserved words that have specific purpose and can’t be used for user defined purposes

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

* raise is a keyword in Python which is used to force a specific exception to occur.
* The syntax is **raise NameError(“name error”) / raise userdefinedException**
* If we try to declare a variable like **raise=’abcd’**, then python interpreter throws an syntax error because right after raise, python interpreter expects an exception name but not assignment operator

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

* Indentation means adding spaces at the beginning of a line of code
* It helps us to define the scope of block of code in Python

Q13. How can we throw some output in Python?

* We can use print function to write on console
* Using open() function and write mode to write to file

Q14. What are operators in Python?

* Operators define the type of operation to be performed between two operand
* +,-,==,\* etc

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

* It returns the quotient after dividing the first operand by the second operand. It can return decimal values
* It gives the floor value of the quotient produced by dividing the two operands.

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

```

iNeuroniNeuroniNeuroniNeuron

```

**var='iNeuron'\*4**

**print(var)**

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

num=int(input("ENter number:"))

print('even' if num%2==0 else 'odd')

Q18. What are boolean operator?

• The logical operators and, or and not are also referred to as boolean operators. While and as well as or operator needs two operands, which may evaluate to true or false, not operator needs one operand evaluating to true or false.

• Boolean and operator returns true if both operands return true.

• Boolean or operators retruns true if atleast one of the operands return true

Q19. What will the output of the following?

```

1 or 0

0 and 0

True and False and True

1 or 0 or 0

```

**1**

**0**

**False**

**1**

Q20. What are conditional statements in Python?

* It is used for decision making and executes certain block of code whenever a condition is satisfied
* We have if, if..else, if..elif..else, Nested if etc

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

* It is used for decision making and executes certain block of code whenever a condition is satisfied
* Traversal starts from beginning of the conditional statement and if any condition is satisfied, that specific block of code will be executed and the comes out of conditional construct

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".

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]

```

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

if age > 18:

    print("I can vote")

else:

    print("I can't vote")

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

num1=int(input("Enter num 1:"))

num2=int(input("Enter num 2:"))

num3=int(input("Enter num 3:"))

if (num1 >= num2) and (num1 >= num3):

 largest = num1

elif (num2 >= num1) and (num2 >= num3):

 largest = num2

else:

 largest = num3

print("The largest number between",num1,",",num2,"and",num3,"is",largest)

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]

```

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

for num in numbers:

    if num%5 ==0 :

        if num > 150 & num < 500:

            continue

        if num > 500:

            break

        print(num)

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

In Python, a string is a sequence of Unicode characters.

Strings can be created by enclosing characters inside a single quote or double-quotes. Even triple quotes can be used in Python but generally used to represent multiline strings and docstrings.

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

* str[index]

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

```

string = "Big Data iNeuron"

desired\_output = "iNeuron"

```

string = "Big Data iNeuron"

print(string[9:])

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

```

string = "Big Data iNeuron"

desired\_output = "norueNi"

```

string = "Big Data iNeuron"

tmp=string[9:]

print(tmp[::-1])

Q30. Resverse the string given in the above question.

string = "Big Data iNeuron"

print(string[::-1])

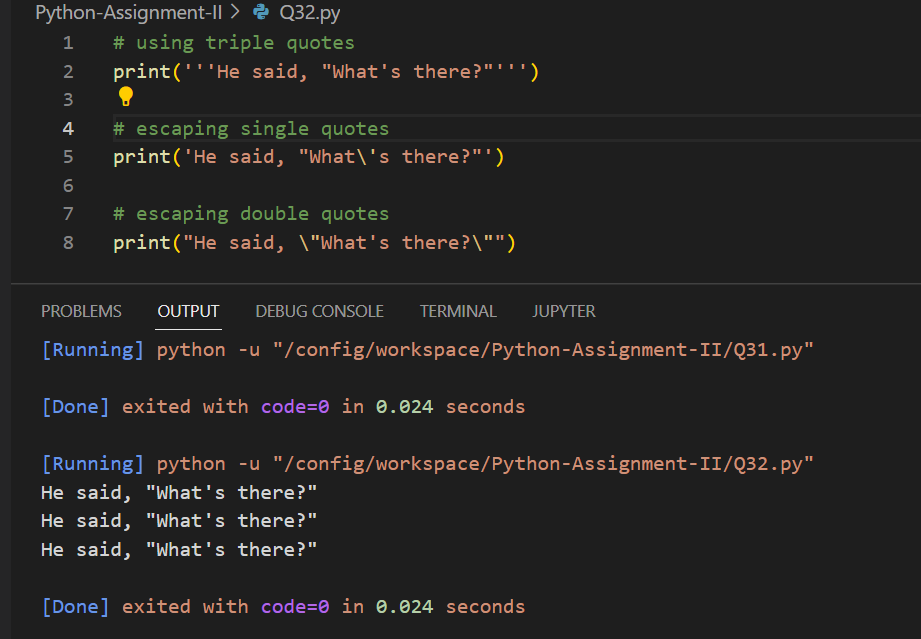
Q31. How can you delete entire string at once?

string = "Big Data iNeuron"

del string

Q32. What is escape sequence?

An escape sequence starts with a backslash and is interpreted differently. If we use a single quote to represent a string, all the single quotes inside the string must be escaped. Similar is the case with double quotes. Here is how it can be done to represent the above text.



Q33. How can you print the below string?

```

'iNeuron's Big Data Course'

```

str="'iNeuron's Big Data Course'"

print(str)

Q34. What is a list in Python?

* List is a sequence of elements
* List is a mutable data type

Q35. How can you create a list in Python?

* a list is created by placing elements inside square brackets [], separated by commas.
* Mylist=[1,2,’a’]
* Q36. How can we access the elements in a list?
* Using the index of an element in list

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

```

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

```

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

print(lst[4][2])

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

lst = []

n = int(input("Enter number of elements : "))

for i in range(0, n):

    ele = (input())

    lst.append(ele) # adding the element

print(len(lst))

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

```

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

```

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

lst.insert(2,'Big')

print(lst)

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

* A tuple is a list of items and they may be of different types
* A tuple in Python is similar to a list. The difference between the two is that we cannot change the elements of a tuple once it is assigned whereas we can change the elements of a list.
* Hence, list is said to be mutable whereas tuple is said to be mutable

Q41. How can you create a tuple in Python?

A tuple is created by placing all the items (elements) inside parentheses (), separated by commas. The parentheses are optional, however, it is a good practice to use them.

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

tup=('Teja','Swapna','Seshu')

tup=tup+("Vaishnavi",)

print(tup)

We can add an element its because tuple does not allow us to modify any immutable elements but we can add new elements or modify mutable elements

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

tup=('Teja','Swapna','Seshu')

tup=tup+("Vaishnavi",)

print(tup)

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

tup=('Teja','Swapna','Seshu')

tup=tup+("Vaishnavi",)

print(len(tup))

Q45. What are sets in Python?

A set is an unordered collection of items. Every set element is unique (no duplicates) and must be immutable (cannot be changed).

However, a set itself is mutable. We can add or remove items from it.

Sets can also be used to perform mathematical set operations like union, intersection, symmetric difference, etc.

Q46. How can you create a set?

A set is created by placing all the items (elements) inside curly braces {}, separated by comma, or by using the built-in set() function.

It can have any number of items and they may be of different types (integer, float, tuple, string etc.). But a set cannot have mutable elements like [lists](https://www.programiz.com/python-programming/list), sets or [dictionaries](https://www.programiz.com/python-programming/dictionary) as its elements.

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

var=set()

var.add('ineuron')

print(var)

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

var=set()

var.add('ineuron','ineuron1')

print(var)

**Output:-**

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

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

**Add():-** To add one element to the set

**Update():-** To add more than one element to the set

Q50. What is clear() in sets?

Removes all elements from set

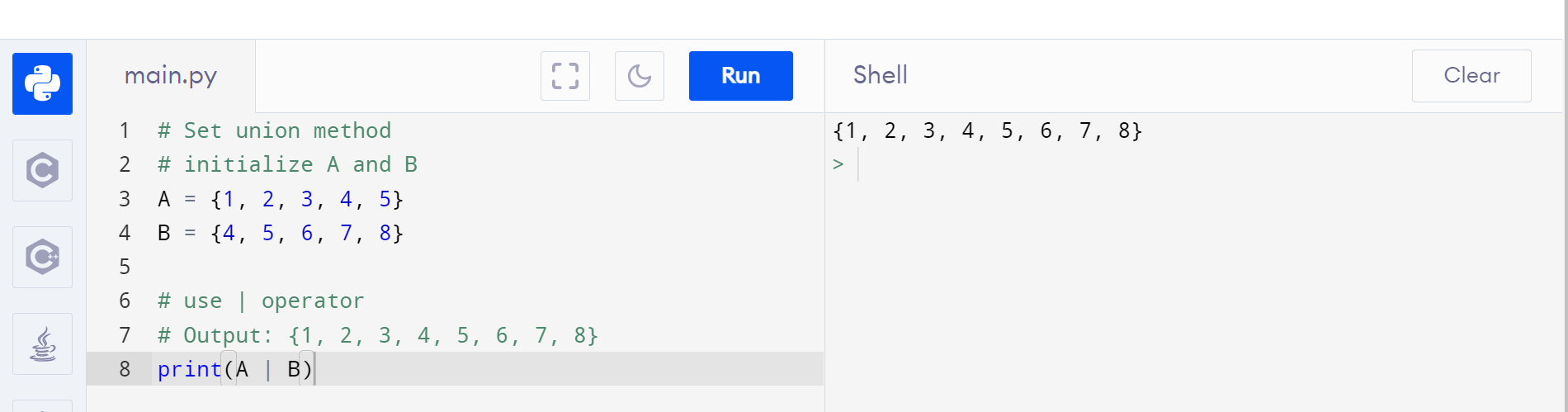
Q51. What is frozen set?

* The frozenset() function returns an immutable frozenset object initialized with elements from the given iterable.
* Frozen set is just an immutable version of a [Python set](https://www.programiz.com/python-programming/set) object. While elements of a set can be modified at any time, elements of the frozen set remain the same after creation.
* Due to this, frozen sets can be used as keys in [Dictionary](https://www.programiz.com/python-programming/dictionary) or as elements of another set. But like sets, it is not ordered (the elements can be set at any index).

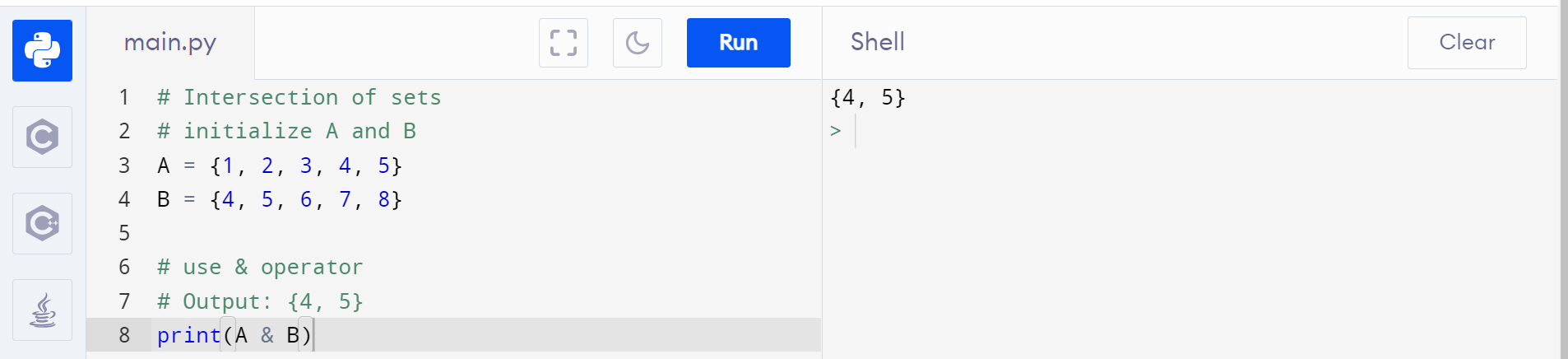
Q52. How is frozen set different from set?

• Frozen set is just an immutable version of a Python set object. While elements of a set can be modified at any time, elements of the frozen set remain the same after creation.

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

* Union of A and B is a set of all elements from both sets. 

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

* Intersection of A and B is a set of elements that are common in both the sets. V 

Q55. What is dictionary ibn Python?

* Python dictionary is an ordered collection (starting from Python 3.7) of items. Each item of a dictionary has a key/value pair.
* While the values can be of any data type and can repeat, keys must be of immutable type ([string](https://www.programiz.com/python-programming/string), [number](https://www.programiz.com/python-programming/numbers) or [tuple](https://www.programiz.com/python-programming/tuple) with immutable elements) and must be unique.

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

• Python dictionary is an ordered collection (starting from Python 3.7) of items. Each item of a dictionary has a key/value pair.

• While the values can be of any data type and can repeat, keys must be of immutable type (string, number or tuple with immutable elements) and must be unique.

* Dictionary has k,v pair while others dont

Q57. How can we delare a dictionary in Python?

# empty dictionary

my\_dict = {}

# dictionary with integer keys

my\_dict = {1: 'apple', 2: 'ball'}

# dictionary with mixed keys

my\_dict = {'name': 'John', 1: [2, 4, 3]}

# using dict()

my\_dict = dict({1:'apple', 2:'ball'})

# from sequence having each item as a pair

my\_dict = dict([(1,'apple'), (2,'ball')])

Q58. What will the output of the following?

```

var = {}

print(type(var))

```

**<class 'dict'>**

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

# Changing and adding Dictionary Elements

my\_dict = {'name': 'Jack', 'age': 26}

# update value

my\_dict['age'] = 27

#Output: {'age': 27, 'name': 'Jack'}

print(my\_dict)

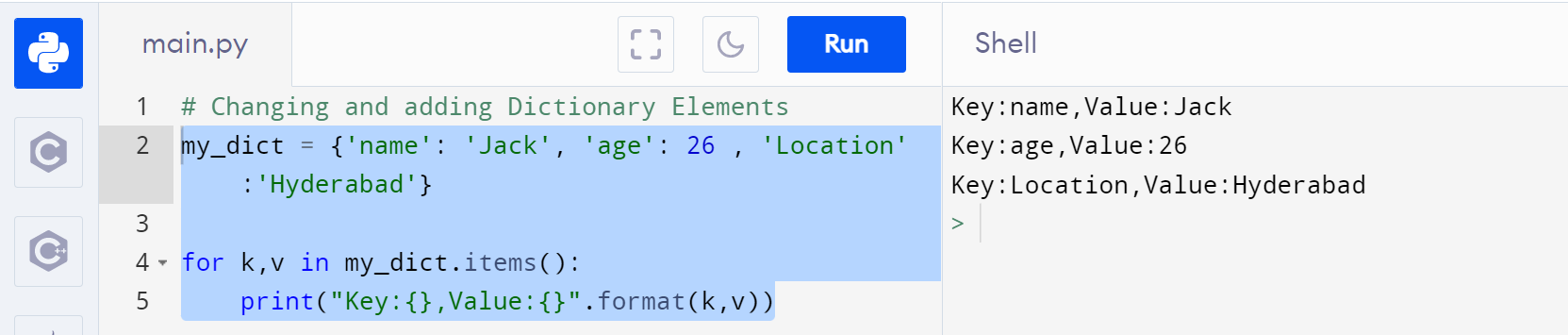
# add item

my\_dict['address'] = 'Downtown'

# Output: {'address': 'Downtown', 'age': 27, 'name': 'Jack'}

print(my\_dict)

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



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

people = {1: {'name': 'John', 'age': '27', 'sex': 'Male'}, 2: {'name': 'Marie', 'age': '22', 'sex': 'Female'}, 3:{'name':'Abha','age':'25'}

}

def recur\_dir(d):

for k,v in d.items():

if isinstance(v, dict):

recur\_dir(v)

else:

print (k,":",v)

recur\_dir(people)

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

Returns the value of the key. If the key does not exist, returns d (defaults to None).

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

Return a new object of the dictionary's items in (key, value) format.

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

|  |  |
| --- | --- |
| [pop(key[,d])](https://www.programiz.com/python-programming/methods/dictionary/pop) | Removes the item with the key and returns its value or d if key is not found. If d is not provided and the key is not found, it raises KeyError. |

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

|  |  |
| --- | --- |
| [popitem()](https://www.programiz.com/python-programming/methods/dictionary/popitem) | Removes and returns an arbitrary item (key, value). Raises KeyError if the dictionary is empty. |

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

|  |  |
| --- | --- |
| [keys()](https://www.programiz.com/python-programming/methods/dictionary/keys) | Returns a new object of the dictionary's keys. |

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

|  |  |
| --- | --- |
| v[alues()](https://www.programiz.com/python-programming/methods/dictionary/values) | Returns a new object of the dictionary's values |

Q68. What are loops in Python?

Loop in Python is used to iterate over a sequence ([list](https://www.programiz.com/python-programming/list), [tuple](https://www.programiz.com/python-programming/tuple), [string](https://www.programiz.com/python-programming/string)) or other iterable objects. Iterating over a sequence is called traversal.

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

1. While loop
2. For loop
3. Nested loop

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

Both for loop and while loop is used to execute the statements repeatedly while the program runs. The major difference between for loop and the while loop is that for loop is used when the number of iterations is known, whereas execution is done in the while loop until the statement in the program is proved wrong

Q71. What is the use of continue statement?

The continue statement is used to skip the rest of the code inside a loop for the current iteration only. Loop does not terminate but continues on with the next iteration.

Q72. What is the use of break statement?

The break statement terminates the loop containing it. Control of the program flows to the statement immediately after the body of the loop.

If the break statement is inside a nested loop (loop inside another loop), the break statement will terminate the innermost loop.

Q73. What is the use of pass statement?

 the pass statement is a null statement which can be used as a placeholder for future code.

Suppose we have a loop or a function that is not implemented yet, but we want to implement it in the future. In such cases, we can use the pass statement.

The syntax of the pass statement is:

pass

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

The range() function returns a sequence of numbers, starting from 0 by default, and increments by 1 (by default), and stops before a specified number.

Q75. How can you loop over a dictionary?

for k,v in dict.items()

for k in dict:

### Coding problems

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

num = 7

factorial = 1

if num < 0:

print("Sorry, factorial does not exist for negative numbers")

elif num == 0:

print("The factorial of 0 is 1")

else:

for i in range(1,num + 1):

factorial = factorial\*i

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

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

def simple\_interest(p,t,r):

print('The principal is', p)

print('The time period is', t)

print('The rate of interest is',r)

si = (p \* t \* r)/100

print('The Simple Interest is', si)

return si

simple\_interest(8, 6, 8)

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

def compound\_interest(principle, rate, time):

Amount = principle \* (pow((1 + rate / 100), time))

CI = Amount - principle

print("Compound interest is", CI)

compound\_interest(10000, 10.25, 5)

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

**num = 103**

**if num > 1:**

**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:**

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

Q80. Write a Python program to check Armstrong Number.

**n = 153**

**s = n**

**b = len(str(n))**

**sum1 = 0**

**while n != 0:**

**r = n % 10**

**sum1 = sum1+(r\*\*b)**

**n = n//10**

**if s == sum1:**

**print("The given number", s, "is armstrong number")**

**else:**

**print("The given number", s, "is not armstrong number")**

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

**def Fibonacci(n):**

**if n<= 0:**

**print("Incorrect input")**

**elif n == 1:**

**return 0**

**elif n == 2:**

**return 1**

**else:**

**return Fibonacci(n-1)+Fibonacci(n-2)**

**print(Fibonacci(10))**

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

**lst=[1,2,3]**

**n=len(lst)**

**lst[0],lst[n-1]=lst[n-1],lst[0]**

**print(lst)**

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

**lst=[1,2]**

**lst[0],lst[1]=lst[1],lst[0]**

**print(lst)**

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

**l = [1000,298,3579,100,200,-45,900]**

**n = 4**

**l.sort()**

**print(l[-n:])**

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

**lst=[1,4,6,8,104,11]**

**temp=0**

**for i in range(0,len(lst)):**

**temp=temp+lst[i]**

**print(temp)**

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

**def isPalindrome(str):**

**for i in range(0, int(len(str)/2)):**

**if str[i] != str[len(str)-i-1]:**

**return False**

**return True**

**s = "malayalam"**

**ans = isPalindrome(s)**

**if (ans):**

**print("Yes")**

**else:**

**print("No")**

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

**str='JohnSmith'**

**index=3**

**str=str[0:index]+str[index+1:]**

**print(str)**

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

**str='JohnSmith'**

**print('Smith' in 'JohnSmith')**

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

**list=['John','Smith','Vaishnavi','Teja','Bapatu']**

**n=4**

**for word in list:**

**if len(word) > n:**

**print(word)**

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

**test\_dict = {'a': [5, 6, 7, 8],**

**'b': [10, 11, 7, 5],**

**'c': [6, 12, 10, 8],**

**'d': [1, 2, 5]}**

**value={val for vals in test\_dict.values() for val in vals}**

**print(value)**

Q91. Write a Python program to merge two dictionary.

**dict\_1 = {1: 'a', 2: 'b'}**

**dict\_2 = {2: 'c', 4: 'd'}**

**dict\_1.update(dict\_2)**

**#dict\_3.update(dict\_1)**

**print(dict\_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}

```

**input=[('Sachin', 10), ('MSD', 7), ('Kohli', 18), ('Rohit', 45)]**

**dic={x[0]:x[1] for x in input}**

**print(dic)**

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)]

```

**list = [9, 5, 6]**

**out = [(x,x \*\* 3)for x in list]**

**print(out)**

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)]

```

**test\_tuple1 = (7, 2)**

**test\_tuple2 = (7, 8)**

**out=[(x,y) for x in test\_tuple1 for y in test\_tuple2]**

**out=out+[(x,y) for x in test\_tuple2 for y in test\_tuple1]**

**print(out)**

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)]

```

**lst=[('for', 24), ('Geeks', 8), ('Geeks', 30)]**

**print(lst.sort(key=lambda x:x[1]))**

Q96. Write a python program to print below pattern.

```

\*

\* \*

\* \* \*

\* \* \* \*

\* \* \* \* \*

```

Q97. Write a python program to print below pattern.

```

\*

\*\*

\*\*\*

\*\*\*\*

\*\*\*\*\*

```

Q98. Write a python program to print below pattern.

```

\*

\* \*

\* \* \*

\* \* \* \*

\* \* \* \* \*

```

**ef triangle(n):**

**k = n - 1**

**for i in range(0, n):**

**for j in range(0, k):**

**print(end=" ")**

**k = k - 1**

**for j in range(0, i+1):**

**# printing stars**

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

**# ending line after each row**

**print("\n")**

**n = 5**

**triangle(n)**

Q99. Write a python program to print below pattern.

```

1

1 2

1 2 3

1 2 3 4

1 2 3 4 5

```

**def alphapat(n):**

**for i in range(1, n+1):**

**for j in range(1, i+1):**

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

**print("\n")**

**n = 5**

**alphapat(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

```

**def alphapat(n):**

**num = 65**

**for i in range(0, n):**

**for j in range(0, i+1):**

**ch = chr(num)**

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

**num = num + 1**

**print("\n")**

**n = 5**

**alphapat(n)**