**Assignment Part-1**

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

Python is a computer programming language often used to build websites and software, automate tasks, and conduct data analysis. Python is a general-purpose language, meaning it can be used to create a variety of different programs and isn’t specialized for any specific problems. This versatility, along with its beginner-friendliness, has made it one of the most-used programming languages today.

Python is commonly used for developing websites and software, task automation, data analysis, and data visualization. Since it’s relatively easy to learn, Python has been adopted by many non-programmers such as accountants and scientists, for a variety of everyday tasks, like organizing finances.

**Q2. Why is Python called a dynamically typed language?**

Strong typing means that variables do have a type and that the type matters when performing operations on a variable. Dynamic typing means that the type of the variable is determined only during runtime.Due to dynamic typing, in Python the same variable can have a different type at different times during the execution. Dynamic typing allows for flexibility in programming, but with a price in performance.

|  |  |
| --- | --- |
| Pros | Cons |
| Beginner-friendly | Issues with design |
| Large Community | Slower than compiled languages |
| Flexible and Extensible | Security |
| Extensive Libraries | Work Environment |
| Embeddable | High memory consumption |
| Highly Scalable | Dynamically-typed language |
| IoT Opportunities | Complex multithreading |
| Portable | Garbage collection leads to potential memory losses |

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

**Q4. In what all domains can we use Python?**

**Python**proudly holds the position as one of the world’s most popular programming languages. In most of the domains, it is giving its share. Employing python allows the user to work on multiple domains ranging from**Data Science, Machine Learning, Deep Learning, Artificial Intelligence, Scientific Computing Scripting, Networking, Game Development to Web Development**. Owing to the programming language’s multiple domain integrity, it is greatly loved by communities.

**1. Data Science:**Data science is an interdisciplinary domain that comprises of three distinct and overlapping areas:

* How to model and summarize data like a statistician
* How to design and use algorithms to store, process, and visualize data like a computer scientist
* How to formulate the right questions and put answers in the right context like a domain expert

Python is an entrusted language for scientific computing tasks, including analysis and visualization of large data sets. The use cases of Python in data science stems from the large and active ecosystem of third-party packages, such as NumPy for manipulation of homogeneous array-based data, Pandas for manipulation of heterogeneous and labeled data, SciPi for computing tasks, Sci-Kit Learn for machine learning, etc. So, a data scientist combines statistical techniques with Python programming language to analyze and interpret complex data.

**2. Automation:** There are times when tasks are too niche to have standard software for it. That is when scripting comes into play. Python allows developers to script custom automation and bring efficiency in tasks in less time. Automating repetitive tasks such as sending emails & voicemails, organizing files & folders, launching programs, filling out forms, etc. can be automated through scripting.

**3. Application Development :** Python is a favourable choice for web application development. Python integrates well with other programming languages and has some amazing web frameworks that boost the application development cycle. [Python web frameworks](https://insights.daffodilsw.com/blog/top-10-python-frameworks-for-web-application-development) basically make it easy to build common backend logic. For example, it helps to deal with the databases, map URLs with the codebase, generate HTML files, and more.

**4. AI & Machine Learning :** Python’s pre-built libraries are the reason why it is preferred for AI & machine learning development. Numpy for scientific computation, Pybrain for machine learning tasks, Scipy for technical and scientific computing are some of the python libraries that make it a supportive technology for AI and ML development.

Moreover, there are python libraries for Natural Language & Text Processing. The Natural Language Toolkit (NLTK) contains open-source Python modules, linguistic data & documentation for R&D in natural language processing and text analytics.

In addition to this, there are some general python libraries for Artificial Intelligence. AIMA, SimpleAI, EasyAI, pyDatalog are a few of them.

**5. Audio/Video Applications: P**ython is a sought-after programming language for creating audio/video applications. An example of this is the [Spotify app](https://insights.daffodilsw.com/blog/how-spotify-works-business-model-and-revenue-streams) that’s built using Python programming language. Some of the python libraries such as PyDub, OpenCV support building audio/video streaming apps with high-level functionality.

**6. Console Applications:** A console application is a computer program that’s designed for the text-only interfaces, such as the command-line interface of Unix, DOS operating system. For building console applications, there are advanced python libraries that help in the development of fully-fledged apps for the command-line interface.

**7. Desktop GUI:**

Python is known for its huge collection of libraries. Just like the areas mentioned above. Python has libraries for building desktop GUI applications as well. Camelot, Kivy, PyGTK are some of the amazing libraries and tool-kits available for building desktop GUI.

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

Variables are containers for storing data values.

Creating Variables

Python has no command for declaring a variable.

A variable is created the moment you first assign a value to it.

Example

x = 10258963

y = "Abhishek"

print(x)

print(y)

Output:

10258963

Abhishek

Variables do not need to be declared with any particular type, and can even change type after they have been set.

Example

x = 4263 # x is of type int

x = "welcome to our assignment" # x is now of type str

print(x)

Output:

welcome to our assignment

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

Input() function:-

The input function is used in all latest version of the Python. It takes the input from the user and then evaluates the expression. The [Python](https://www.javatpoint.com/python-tutorial) interpreter automatically identifies the whether a user input a string, a number, or a list. Let's understand the following example.

For example:

Name = input(“Enter your name:”)

print(Name)

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

By default input() function helps in taking user input as string.

For example:

name = input(“Enter your name:”)

age = input(“Enter your age”)

print(“Your name and age is: ”, name, age)

**Q8. What is type casting?**

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.

#### Implicit Type casting

Implicit type conversion is performed automatically by the interpreter, without user ntervention. Python automatically converts one data type to another data type. This process doesn’t need any user involvement Python promotes the conversion of lower data type, for example, integer to higher data type says float to avoid data loss. This type of conversion or type casting is called UpCasting.

#### Explicit Type casting

In Explicit Type conversion, the user or programmer converts the data type of an object to the required data type. In Python we use predefined functions like int(), float(), str(), bool() etc to perform explicit type conversion.

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

No, we can not take multiple input from user using single input() function because it is used declare to take one input from user.

If we want to take multiple input from user the we will using split().

split( ) function helps us get multiple inputs from the user and assign them to the respective variables in one line. This function is generally used to separate a given string into several substrings. However, you can also use it for taking multiple inputs. **The function generally breaks the given input by the specified separator**and in case the separator is not provided then **any white space is considered as a separator.**

**Q10. What are keywords?**

Python keywords are special reserved words that have specific meanings and purposes and can’t be used for anything but those specific purposes. These keywords are always available—you’ll never have to import them into your code.Python keywords are different from Python’s [built-in functions and types](https://docs.python.org/3/library/functions.html). The built-in functions and types are also always available, but they aren’t as restrictive as the keywords in their usage.An example of something you can’t do with Python keywords is assign something to them. If you try, then you’ll get a SyntaxError. You won’t get a SyntaxError if you try to assign something to a built-in function or type, but it still isn’t a good idea. For a more in-depth explanation of ways keywords can be misused, check out [Invalid Syntax in Python: Common Reasons for SyntaxError](https://realpython.com/invalid-syntax-python/#misspelling-missing-or-misusing-python-keywords).

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

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 and they must be written as they are.

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

Indentation is a very important concept of Python because without properly indenting the Python code, you will end up seeing IndentationError and the code will not get compiled.

## Python Indentation:

Python indentation refers to adding white space before a statement to a particular block of code. In another word, all the statements with the same space to the right, belong to the same code block.

Statement (line 1), if condition (line 2), and statement (last line) belongs to the same block which means that after statement 1, if condition will be executed. and suppose the if condition becomes False then the Python will jump to the last statement for execution.

The nested if-else belongs to block 2 which means that if nested if becomes False, then Python will execute the statements inside the else condition.

Statements inside nested if-else belong to block 3 and only one statement will be executed depending on the if-else condition.

Python indentation is a way of telling a Python interpreter that the group of statements belongs to a particular block of code. A block is a combination of all these statements. Block can be regarded as the grouping of statements for a specific purpose. Most programming languages like C, C++, and Java use braces { } to define a block of code. Python uses indentation to highlight the blocks of code. Whitespace is used for indentation in Python. All statements with the same distance to the right belong to the same block of code. If a block has to be more deeply nested, it is simply indented further to the right. You can understand it better by looking at the following lines of code.

**Q13. How can we throw some output in Python?**

The basic way to do output is the **print statement**. To end the printed line with a newline, add a print statement without any objects. This will print to any object that implements write(), which includes file objects.To easily display single and multiple variables in Python, use the print() statement. For multiple variables, **use the comma operators**.

**Q14. What are operators in Python?**

There are different types of Python operators are:

Arithmetic operators:- +,-,/,\*.%,\*\*

Assignment Operators:- =, +=, -=,\*=

Comparison Operators:- >=,<=, !=, > ,<

Logical Operators:- and, or

Bitwise Operators:- &, |, ~, ^

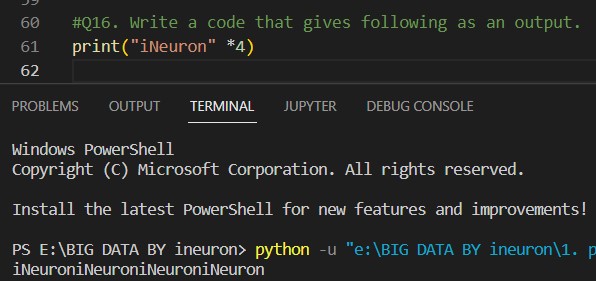
**Q15. What is difference between / and // operators?**

**/ :=** These operator used for simple division

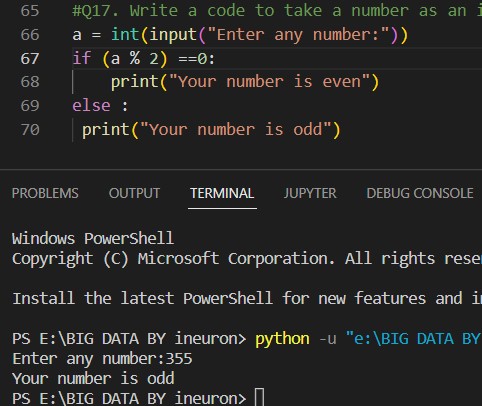
**// :=** The division of operands where the result is the quotient in which the digits after the decimal point are removed. But if one of the **operands is negative,** the result is floored , i.e., rounded away from zero.

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

**iNeuroniNeuroniNeuroniNeuron**



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



**Q18. What are boolean operator?**

In programming, we use Boolean data type in comparisons and flow of control.

True or False are called Boolean values which are keywords in Python. Instructions that combine operators and values to perform mathematical or logical computations are called expressions.

Expressions that yield these Boolean values and are formed with operators called Boolean operators are called Boolean expressions. The operations that take place in the process are called Boolean operations.

**Q19. What will the output of the following?**

**1 or 0**

**0 and 0**

**True and False and True**

**1 or 0 or 0**

****

**Q20. What are conditional statements in Python?**

The if statement is a conditional statement in python, that is used to determine whether a block of code will be executed or not. Meaning if the program finds the condition defined in the if statement to be true, it will go ahead and execute the code block inside the if statement.

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

Decision making is required when we want to execute a code only if a certain condition is satisfied.

The if…elif…else statement is used in Python for decision making.

### Python if Statement Syntax

Here, the program evaluates the test expression and will execute statement(s) only if the test expression is True.

If the test expression is False, the statement(s) is not executed.

In Python, the body of the if statement is indicated by the indentation. The body starts with an indentation and the first unindented line marks the end.

Python interprets non-zero values as True. None and 0 are interpreted as False.

### Python if Statement Flowchart

Flowchart of if statement in Python programming

In the above example, num > 0 is the test expression.

The body of if is executed only if this evaluates to True.

When the variable num is equal to 3, test expression is true and statements inside the body of if are executed.

If the variable num is equal to -1, test expression is false and statements inside the body of if are skipped.

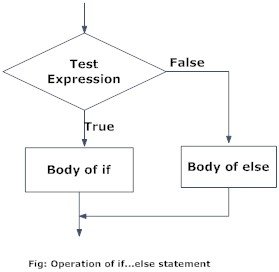
The print() statement falls outside of the if block (unindented). Hence, it is executed regardless of the test expression.

## Python if...else Statement

The if..else statement evaluates test expression and will execute the body of if only when the test condition is True.

If the condition is False, the body of else is executed. Indentation is used to separate the blocks.

### Python if..else Flowchart

Flowchart of if...else statement in Python

In the above example, when num is equal to 3, the test expression is true and the body of if is executed and the body of else is skipped.

If num is equal to -5, the test expression is false and the body of else is executed and the body of if is skipped.

If num is equal to 0, the test expression is true and body of if is executed and body of else is skipped.

## Python if...elif...else Statement

### Syntax of if...elif...else

The elif is short for else if. It allows us to check for multiple expressions.

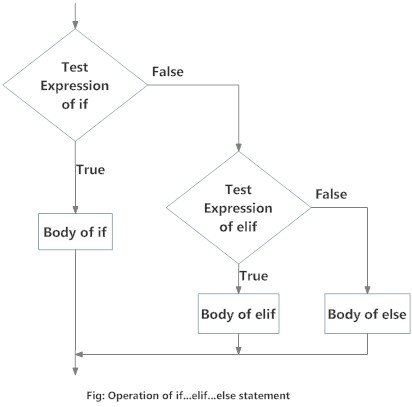
If the condition for if is False, it checks the condition of the next elif block and so on.

If all the conditions are False, the body of else is executed.

Only one block among the several if...elif...else blocks is executed according to the condition.

The if block can have only one else block. But it can have multiple elif blocks.

### Flowchart of if...elif...else

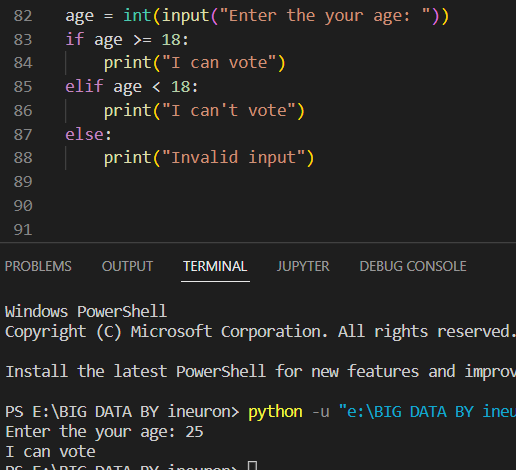
Flowchart of if...elif....else statement in Python

## Python Nested if statements

We can have a if...elif...else statement inside another if...elif...else statement. This is called nesting in computer programming.

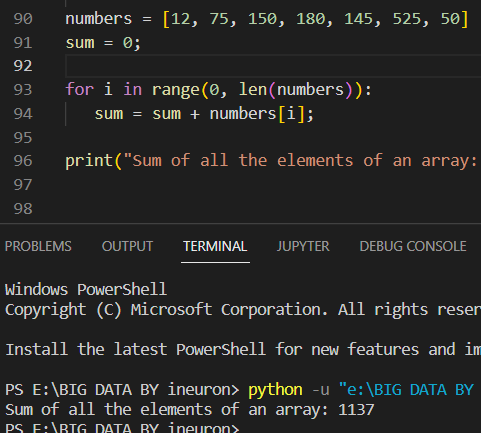
Any number of these statements can be nested inside one another. Indentation is the only way to figure out the level of nesting. They can get confusing, so they must be avoided unless necessary.

**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".**

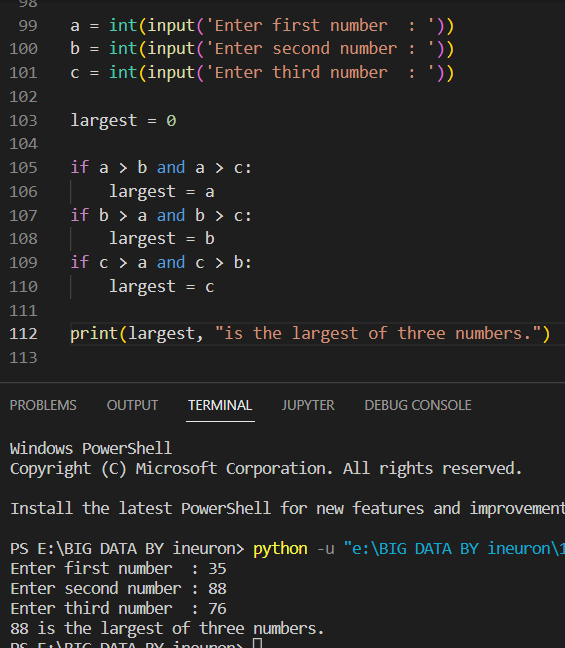


**Q23. Write a code that displays the sum of all the even numbers from the given list.**

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



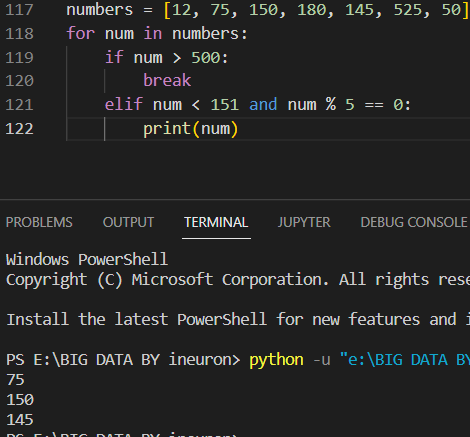
**Q24. Write a code to take 3 numbers as an input from the user and display the greatest no as output.**



**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]**



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

A string is a data structure in Python that represents a sequence of characters. It is an immutable data type, meaning that once you have created a string, you cannot change it. Strings are used widely in many different applications, such as storing and manipulating text data, representing names, addresses, and other types of data that can be represented as text.

Python does not have a character data type, a single character is simply a string with a length of 1. Square brackets can be used to access elements of the string.

For example: print("welcome to python mega assignment")

Output: welcome to our mega assignment

Creating a String in Python:

Strings in Python can be created using single quotes or double quotes or even triple quotes.

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

Often in programming languages, individual items in an ordered set of data can be accessed directly using a numeric index or key value. This process is referred to as indexing.

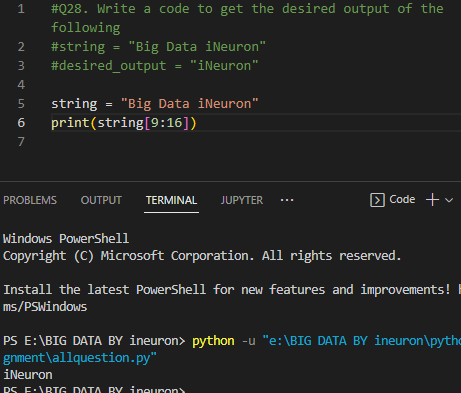
In Python, strings are ordered sequences of character data, and thus can be indexed in this way. Individual characters in a string can be accessed by specifying the string name followed by a number in square brackets ([]).

String indexing in Python is zero-based: the first character in the string has index 0, the next has index 1, and so on. The index of the last character will be the length of the string minus one.

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

**string = "Big Data iNeuron"**

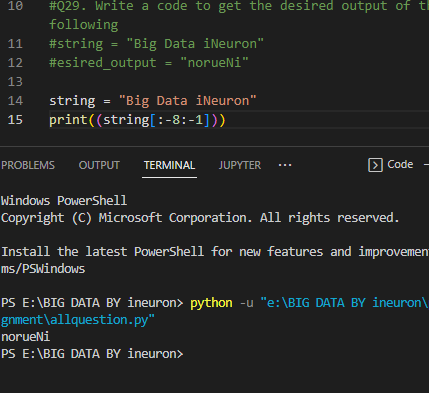
**desired\_output = "iNeuron"**

****

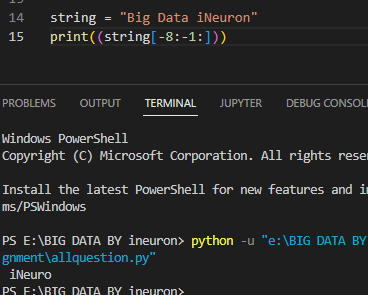
**Q29. Write a code to get the desired output of the following**

**string = "Big Data iNeuron"**

**desired\_output = "norueNi"**

****

**Q30. Resverse the string given in the above question.**

****

**Q31. How can you delete entire string at once?**

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

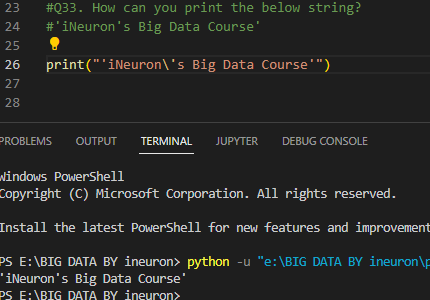
del string1

**Q32. What is escape sequence?**

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. So escape sequences are formed using two things: the first is a backslash (\\), and the second is the set of one or more characters following that backslash (\\).

**Q33. How can you print the below string?**

**'iNeuron's Big Data Course'**

****

**Q34. What is a list in Python?**

The list is a most versatile datatype available in Python which can be written as a list of comma-separated values (items) between square brackets. Important thing about a list is that items in a list need not be of the same type.

Creating a list is as simple as putting different comma-separated values between square brackets. For example −

list1 = ['physics', 'chemistry', 1997, 2000];

list2 = [1, 2, 3, 4, 5 ];

list3 = ["a", "b", "c", "d"]

Similar to string indices, list indices start at 0, and lists can be sliced, concatenated and so on

**Q35. How can you create a list in Python?**

To create a list in Python, we use square brackets ([]). Here's what a list looks like:

ListName = [ListItem, ListItem1, ListItem2, ListItem3, ...]

Note that lists can have/store different data types. You can either store a particular data type or mix them.

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

Indexing and slicing are the most common means that are used to access lists. We can also access items in a list with loops like the for loop.

1) Indexing

A Python list uses the zero-based numbering system. Meaning, all its items are uniquely identified by an index number starting from 0 to n-1 where n is the length of the list

2) Slicing

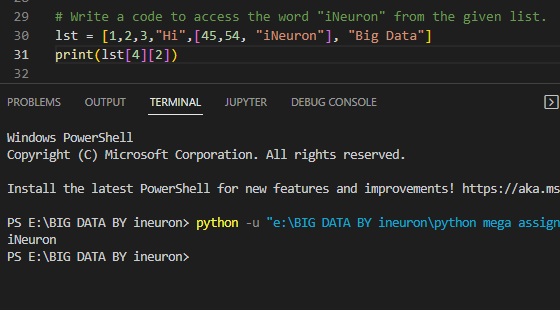
Unlike indexing that only returns one item, slicing on the other hand can return a range of items.

3) Using loops

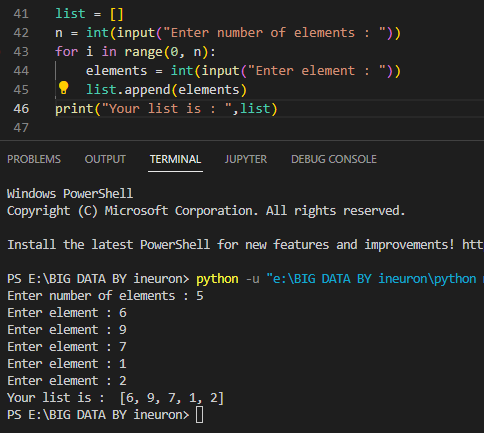
Loops are mostly used to access items in a list in order to manipulate the items. So, in case we want to operate on the items of a list, we can use the for loop to access the items and pass them over to be operated on.

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

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

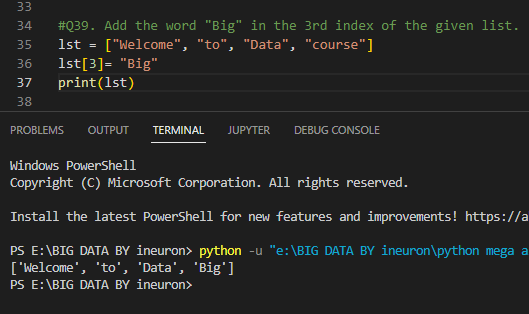
****

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

****

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

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

****

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

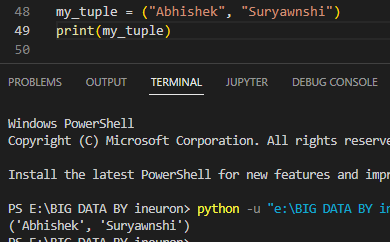
In Python, tuples are allocated large blocks of memory with lower overhead, since they are immutable; whereas for lists, small memory blocks are allocated. Between the two, tuples have smaller memory. This helps in making tuples faster than lists when there are a large number of elements. In simple terms, the size would mean the amount of memory a tuple is storing if it is small or large memory. The size can be calculated using the built-in len() function.

**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.

A tuple can have any number of items and they may be of different types (integer, float, list, [string](https://www.programiz.com/python-programming/string), etc.).

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

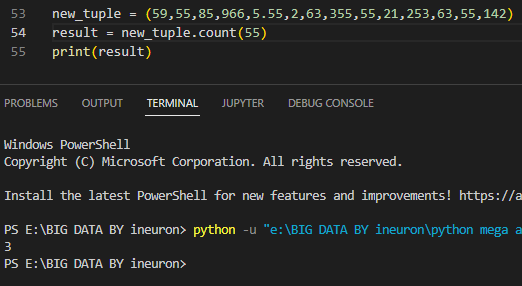
****

Tuple accept similar datatype, different datatypes, nested tuple that shows tuple flexibiltity.

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

* You can't add elements to a tuple because of their immutable property. There's no append() or extend() method for tuples,
* You can't remove elements from a tuple, also because of their immutability. Tuples have no remove() or pop() method,
* You can find elements in a tuple since this doesn't change the tuple.
* You can also use the in operator to check if an element exists in the tuple.

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

****

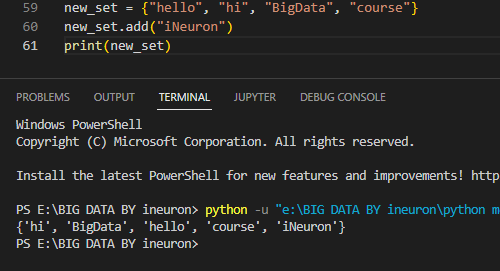
**Q45. What are sets in Python?**

Sets are used to store multiple items in a single variable. Set is one of 4 built-in data types in Python used to store collections of data, the other 3 are [List](https://www.w3schools.com/python/python_lists.asp), [Tuple](https://www.w3schools.com/python/python_tuples.asp), and [Dictionary](https://www.w3schools.com/python/python_dictionaries.asp), all with different qualities and usage. A set is a collection which is unordered*,*unchangeable\**,* and unindexed*.*

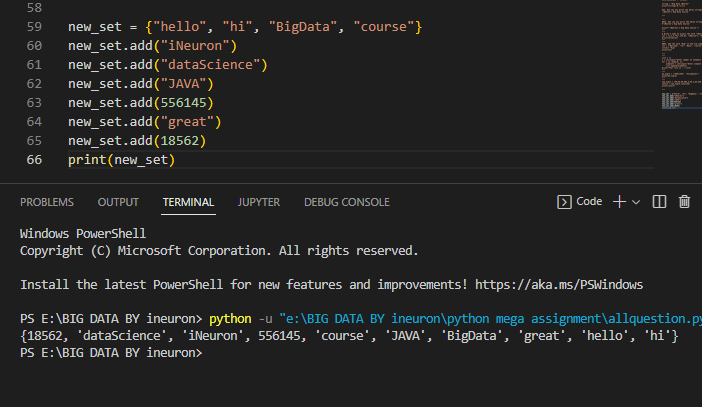
**Q46. How can you create a set?**

Sets are used to store multiple items in a single variable. Set is one of 4 built-in data types in Python used to store collections of data, the other 3 are List, Tuple, and Dictionary, all with different qualities and usage. A set is a collection which is unordered, unchangeable\*, and unindexed. Set items are unchangeable, but you can remove items and add new items. Sets are written with curly brackets.

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

****

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



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

We use add() method to add single value to a set. We use update() method to add sequence values to a set.

**Q50. What is clear() in sets?**

Python Set clear() method removes all elements from the set.

Python Set clear() Method Syntax:

Syntax: set.clear()

parameters**:**

The clear() method doesn’t take any parameters*.*

**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 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 or as elements of another set

**Q52. How is frozen set different from set?**

A set is a mutable object while frozenset provides an immutable implementation.Additionally, the collections library includes the Counter object which is an implementation of a multiset, it stores both the unique items and a count as to how many times it has been added to the container.

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

Return a set containing the union of sets.

x = {"orange", "pineapple", "cherry"}

y = {"google", "amazon", "Mango"}

z = x.union(y)

print(z)

Output: {"orange", "pineapple", "cherry", "google", "amazon", "Mango"}

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

The intersection() method returns a set that contains the similarity between two or more sets.

Meaning: The returned set contains only items that exist in both sets, or in all sets if the comparison is done with more than two sets.

x = {"a", "b", "c", ”d”}

y = {"c", "d", "e"}

z = {"f", "g", "c", ”d”}

result = x.intersection(y, z)

print(result)

output: {‘c’,’d’}

**Q55. What is dictionary ibn Python?**

A dictionary is a kind of [data structure](https://www.simplilearn.com/tutorials/data-structure-tutorial/what-is-data-structure) that stores items in key-value pairs. A key is a unique identifier for an item, and a value is the [data](https://www.simplilearn.com/tutorials/python-tutorial/data-types-in-python) associated with that key. Dictionaries often store information such as words and definitions, but they can be used for much more. Dictionaries are mutable in Python, which means they can be changed after they are created. They are also unordered, indicating the items in a dictionary are not stored in any particular order.

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

The dictionary Data Structure in Python is an unordered collection of items. While other Data Structures use only one value as the element, the dictionary is a slightly more compound data structure. It makes use of two elements i.e. a pair of elements, namely, a key and a value.

**Q57. How can we delare a dictionary in Python?**

Dictionary in Python is a collection of keys values, used to store data values like a map, which, unlike other data types which hold only a single value as an element.

Example of Dictionary in Python : Dictionary holds key:value pair. Key-Value is provided in the dictionary to make it more optimized.

Dict = {1: 'Geeks', 2: 'For', 3: 'Geeks'}

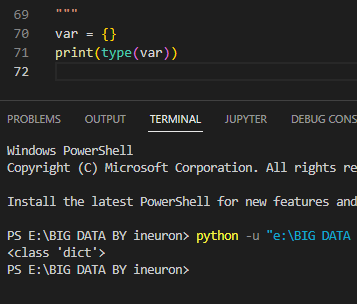
print(Dict)

Output:- {1: 'Geeks', 2: 'For', 3: 'Geeks'}

**Q58. What will the output of the following?**

**var = {}**

**print(type(var))**

****

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

To add an item to a Python dictionary, you should assign a value to a new index key in your dictionary. Unlike lists and tuples, there is no add(), insert(), or append() method that you can use to add items to your data structure. Instead, you have to create a new index key, which will then be used to store the value you want to store in your dictionary.

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

When working with lists, you access list items by mentioning the list name and using square bracket notation. In the square brackets you specify the item's index number (or position).You can't do exactly the same with dictionaries. When working with dictionaries, you can't access an element by referencing its index number, since dictionaries contain key-value pairs .Instead, you access the item by using the dictionary name and square bracket notation, but this time in the square brackets you specify a key. Each key corresponds with a specific value, so you mention the key that is associated with the value you want to access.

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

In Python, a nested dictionary is a dictionary inside a dictionary. It's a collection of dictionaries into one single dictionary.

nested\_dict = { 'dictA': {'key\_1': 'value\_1'},

'dictB': {'key\_2': 'value\_2'}}

Here, the nested\_dict is a nested dictionary with the dictionary dictA and dictB. They are two dictionary each having own key and value.

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

Python Dictionary get() Method return the value for the given key if present in the dictionary. If not, then it will return None (if get() is used with only one argument).

Python Dictionary get() Method Syntax:

Syntax : Dict.get(key, default=None)

Parameters:

key: The key name of the item you want to return the value from

Value: (Optional) Value to be returned if the key is not found. The default value is None.

Returns: Returns the value of the item with the specified key or the default value.

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

Dictionary in Python is an unordered collection of data values, used to store data values like a map, which unlike other Data Types that hold only single value as an element, Dictionary holds key : value pair.

In Python Dictionary, items() method is used to return the list with all dictionary keys with values.

Syntax: dictionary.items()

Parameters: This method takes no parameters.

Returns: A view object that displays a list of a given dictionary’s (key, value) tuple pair

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

The pop() method **removes the element at the specified position**.

Python list method pop() removes and returns last object or obj from the list.

Syntax :

Following is the syntax for pop() method −

list.pop(obj = list[-1])

Parameters :

obj − This is an optional parameter, index of the object to be removed from the list.

Return Value:

This method returns the removed object from the list.

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

Python dictionary popitem() method removes the last inserted key-value pair from the dictionary and returns it as a tuple.

Python Dictionary popitem() Method Syntax:

Syntax : dict.popitem()

Parameters : None

Returns : A tuple containing the arbitrary key-value pair from dictionary. That pair is removed from dictionary.

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

The keys() method returns a view object. The view object contains the keys of the dictionary, as a list.

The view object will reflect any changes done to the dictionary, see example below.

car = {

"brand": "Ford",

"model": "Mustang",

"year": 1964

}

x = car.keys()

car["color"] = "white"

print(x)

Output:- dict\_keys(['brand', 'model', 'year', 'color'])

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

values() is an inbuilt method in Python programming language that **returns a view object**. The view object contains the values of the dictionary, as a list. If you use the type() method on the return value, you get “dict\_values object”. It must be cast to obtain the actual list.

**Q68. What are loops in Python?**

Python programming language provides the following types of loops to handle looping requirements. Python provides three ways for executing the loops. While all the ways provide similar basic functionality, they differ in their syntax and condition checking time.

In python, a while loop is used to execute a block of statements repeatedly until a given condition is satisfied. And when the condition becomes false, the line immediately after the loop in the program is executed.

A for loop in Python is a control flow statement that is used to repeatedly execute a group of statements as long as the condition is satisfied. Such a type of statement is also known as an iterative statement.

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

There are three types of loop in python:

* For Loop. A for loop in Python is used to iterate over a sequence (list, tuple, set, dictionary, and string). Flowchart: ...
* While Loop. The while loop is used to execute a set of statements as long as a condition is true. ...
* Nested Loop. If a loop exists inside the body of another loop, it is called a nested loop

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

The for and while loops are both conditional statements. A for loop is a single-line command that will be executed repeatedly. While loops can be single-lined or contain multiple commands for a single condition.

Both the for loop and the while loop are important in computer languages for obtaining results. The condition is met if the command syntax is correct.

Both the for loop and the while loop are iteration statements, but they have distinct characteristics. The for loop declares everything (initialization, condition, iteration) at the top of the loop body. In contrast, only initialization and condition are at the top of the body of the loop in a while loop, and iteration can be written anywhere in the body of the loop.

**Q71. What is the use of continue statement?**

The continue keyword is used to end the current iteration in a for loop (or a while loop), and continues to the next iteration..

Use the [continue](https://www.w3schools.com/python/ref_keyword_continue.asp) keyword to end the current iteration in a loop, but continue with the next.

i = 1

while i < 9:

print(i)

if i == 3:

continue

i += 1

output:

​

1

2

4

5

6

7

8

9

**Q72. What is the use of break statement?**

The break keyword is used to break out a for loop, or a while loop.

i = 1

while i < 9:

print(i)

if i == 3:

break

i += 1

output:

​

1

2

3

**Q73. What is the use of pass statement?**

The pass statement is used as a placeholder for future code.

When the pass statement is executed, nothing happens, but you avoid getting an error when empty code is not allowed.

Empty code is not allowed in loops, function definitions, class definitions, or in if statements.

having an empty function definition like this, would raise an error without the pass statement

**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.

range(start, stop, step)

start: Optional. An integer number specifying at which position to start. Default is 0

stop: Required. An integer number specifying at which position to stop (not included).

step: Optional. An integer number specifying the incrementation. Default is 1

**Q75. How can you loop over a dictionary?**

You can loop through a dictionary 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.

For Example:

thisdict = {

"brand": "Ford",

"model": "Mustang",

"year": 1964

}

for x in thisdict:

print(x)

Output:

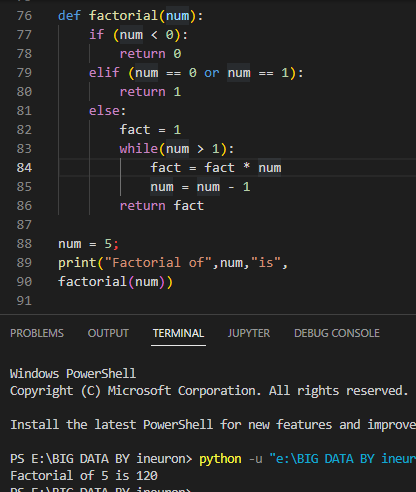
brand

model

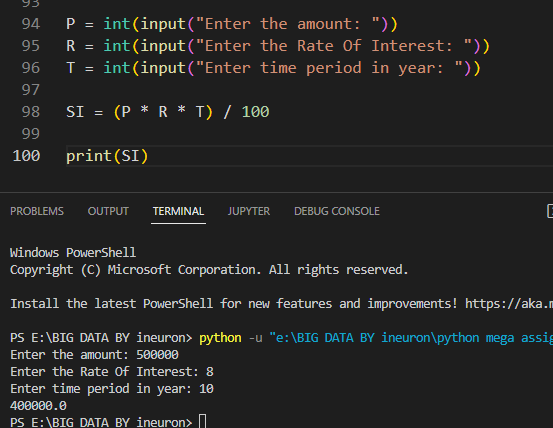
year

**Coding problems**

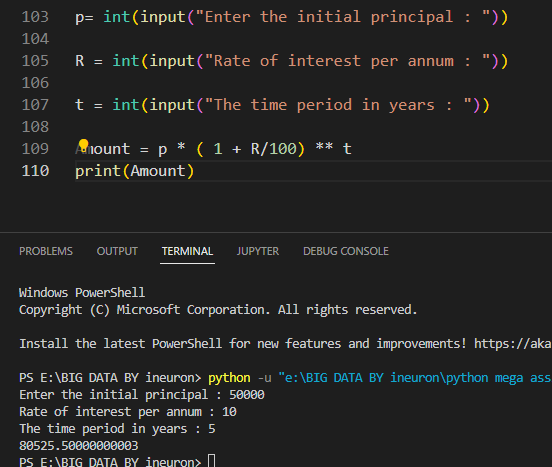
**Q76. Write a Python program to find the factorial of a given number.**

****

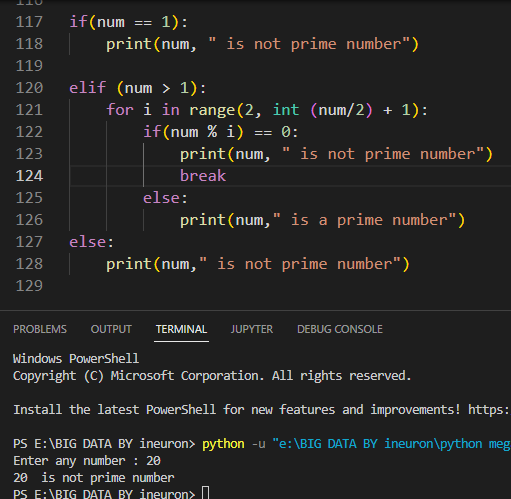
**Q77. Write a Python program to calculate the simple interest. Formula to calculate simple interest is SI = (P*R*T)/100**

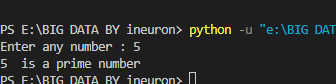
****

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

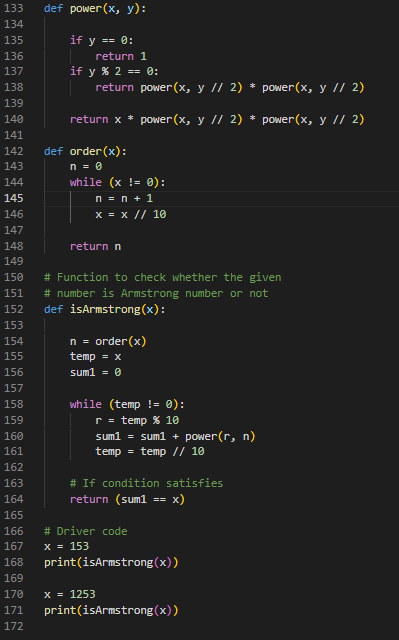
****

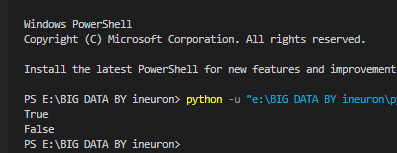
**Q79. Write a Python program to check if a number is prime or not.**

****

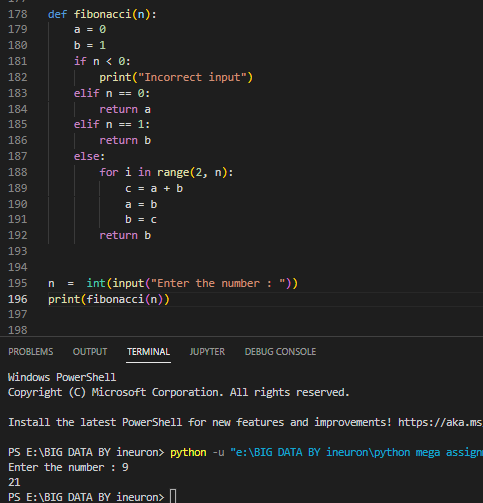
****

**Q80. Write a Python program to check Armstrong Number.**

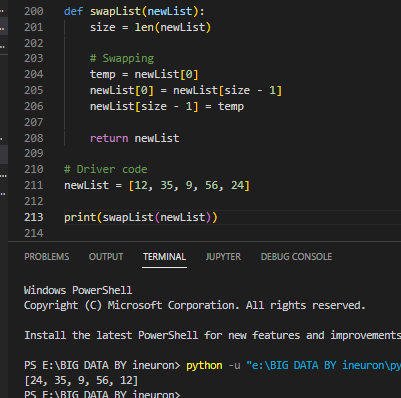
****

****

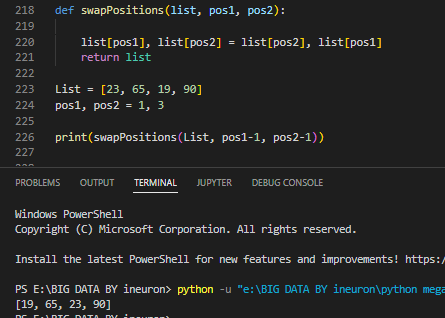
**Q81. Write a Python program to find the n-th Fibonacci Number.**

****

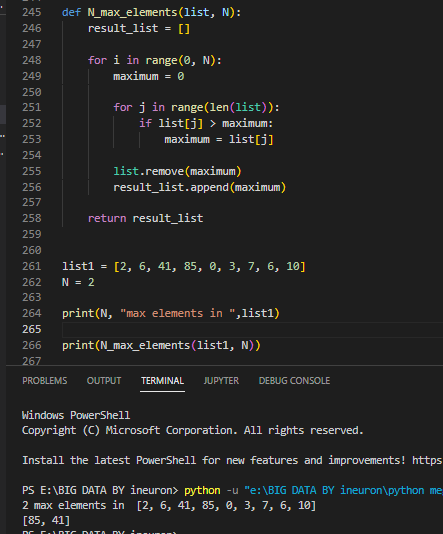
**Q82. Write a Python program to interchange the first and last element in a list.**

****

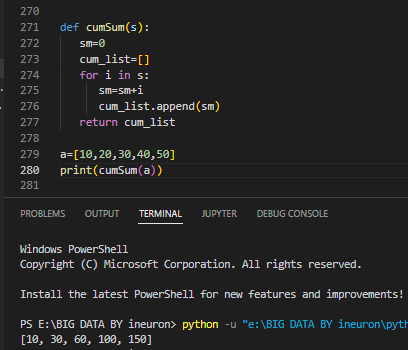
**Q83. Write a Python program to swap two elements in a list.**

****

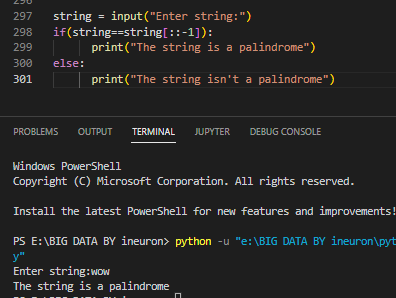
**Q84. Write a Python program to find N largest element from a list.**

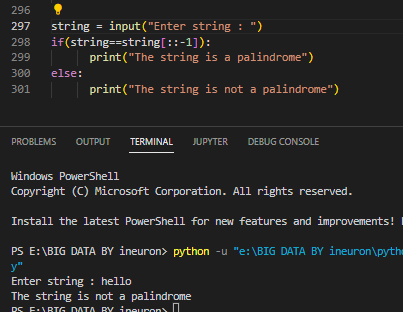
****

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

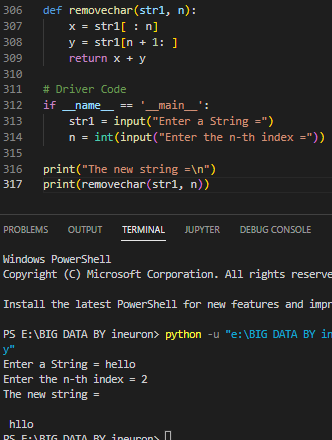
****

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

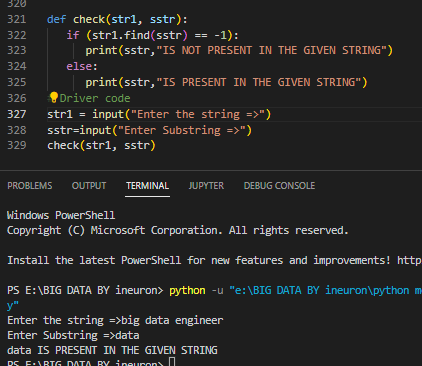
****

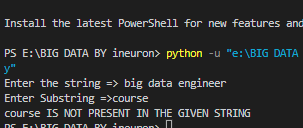
****

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

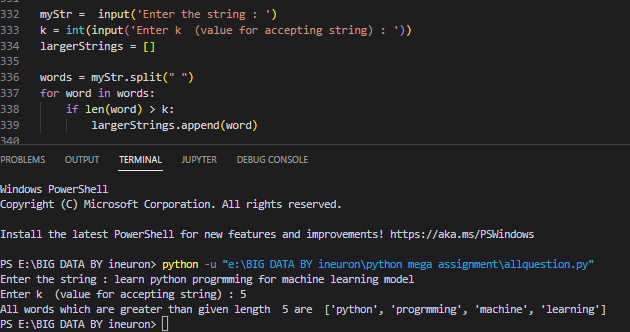
****

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

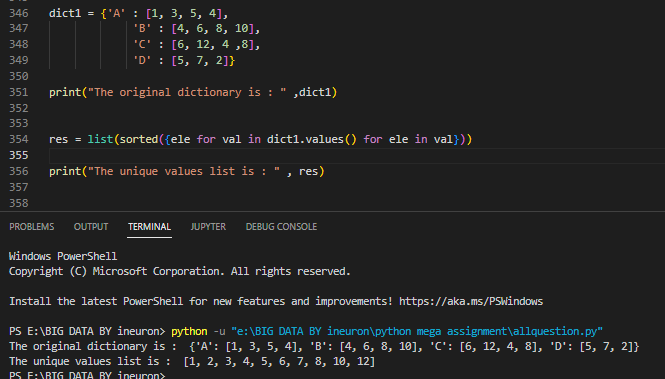
****

****

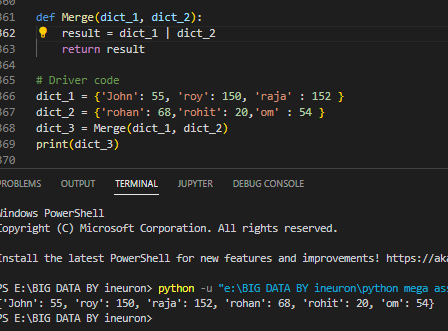
**Q89. Write a Python program to find words which are greater than given length k.**

****

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

****

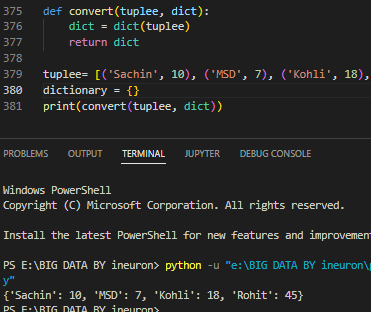
**Q91. Write a Python program to merge two dictionary.**

****

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

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

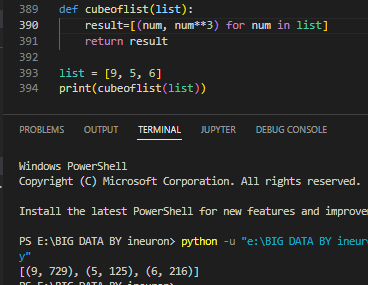
**Output : {'Sachin': 10, 'MSD': 7, 'Kohli': 18, 'Rohit': 45}**

****

**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]**

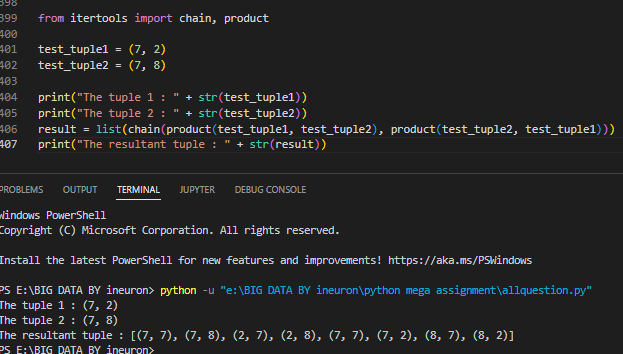
**Output: [(9, 729), (5, 125), (6, 216)]**

****

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

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

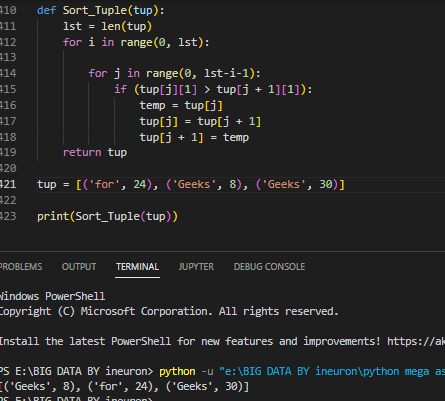
**Output : [(7, 7), (7, 8), (2, 7), (2, 8), (7, 7), (7, 2), (8, 7), (8, 2)]**

****

**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)]**

****

**Q96. Write a python program to print below pattern.**

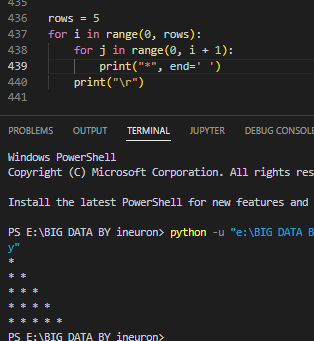
**\***

**\* \***

**\* \* \***

**\* \* \* \***

**\* \* \* \* \***

****

**Q97. Write a python program to print below pattern.**

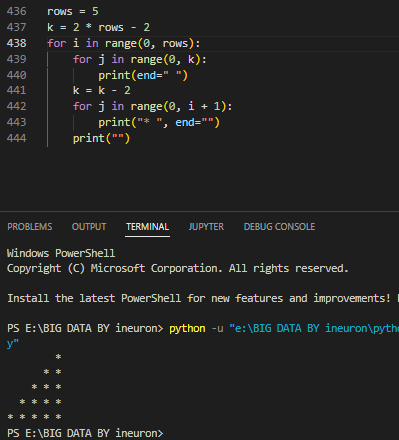
**\***

**\*\***

**\*\*\***

**\*\*\*\***

**\*\*\*\*\***

****

**Q98. Write a python program to print below pattern.**

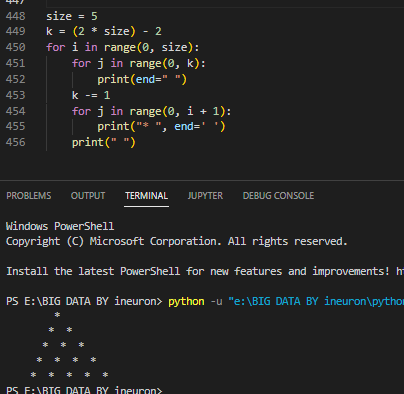
**\***

**\* \***

**\* \* \***

**\* \* \* \***

**\* \* \* \* \***

****

**Q99. Write a python program to print below pattern.**

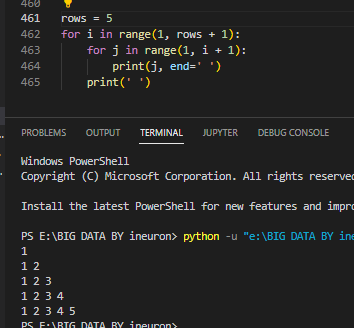
**1**

**1 2**

**1 2 3**

**1 2 3 4**

**1 2 3 4 5**

****

**Q100. Write a python program to print below pattern.**

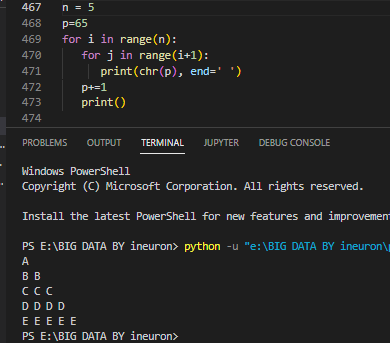
**A**

**B B**

**C C C**

**D D D D**

**E E E E E**

****