#### What are the built-in data types in Python?

There are multiples built-in data types in Python. They are int,  float, complex, bool, list, tuple, set, dict, str.

Note: You don’t have to tell all the data types present in Python. Mention some of them you mostly use. The interviewer may ask questions based on your answer.

1. What’s the difference between **list** and **tuple**?

Both [list](https://geekflare.com/python-linked-lists/) and tuple are used to store the collection of objects. The main difference between the list and tuple is “the list is mutable object whereas tuple is an immutable object”.

#### What are mutable and immutable data types?

Mutable data types can be changed after creating them. Some of the mutable objects in Python are list, set, dict.

Immutable data types can’t be changed after creating them. Some of the immutable objects in Python are str, tuple.

#### Explain some methods of the list.

1. append – the method is used to add an element to the list. It adds the element to the end of the list.

a.append(5)

2. pop – the method is used to remove an element from the list. It will remove the last element if we don’t provide any index as an argument or remove the element at the index if we provide an argument.

a.pop()

3. remove – the method is used to remove an element from the list. We need to provide the element as an argument that we want to remove from the list. It removes the first occurrence of the element from the list.

4. sort – the method used to sort the list in ascending or descending order.

5. reverse – the method is used to reverse the list elements.

#### Explain some methods of string

1. split – the method is used to split the string at desired points. It returns the list as a result. By default, it splits the string at spaces. We can provide the delimiter as an argument for the method.

2. join – the method is used to combine the list of string objects. It combines the string objects with the delimiter we provide.

#### What’s the negative indexing in lists?

The index is used to access the element from the lists. Normal indexing of the list starts from 0.

Similar to normal indexing, negative indexing is also used to access the elements from the lists. But, negative indexing allows us to access the index from the end of the list. The start of the negative indexing is -1. And it keeps on increasing like -2, -3, -4, etc.., till the length of the list.

#### Explain some methods of dict

1. items – the method returns key: value pairs of dictionaries as a list of tuples.

2. pop – the method is used to remove the key: value pair from the dictionary. It accepts the key as an argument and removes it from the dictionary.

#### What is slicing in Python?

Slicing is used to access the subarray from a sequence data type. It returns the data from the sequence data type based on the arguments we provide. It returns the same data type as the source data type.

Slicing accepts three arguments. They are the start index, end index, and increment step. The syntax of slicing is variable[start:end:step]. Arguments are not mandatory for slicing. You can specify an empty colon (:) which returns the entire data as the result.

#### Which data types allow slicing?

We can use slicing on list, tuple, and str data types.

#### What are unpacking operators in Python? How to use them?

The \* and \*\* operators are unpacking operators in Python.

The \* unpacking operator is used to assign multiple values to different values at a time from sequence data types.

The \*\* unpacking operator is used with dict data types. The unpacking in dictionaries doesn’t work like unpacking with sequence data types.

The unpacking in dictionaries is mostly used to copy key: value items from one dictionary to another.

#### Does Python have switch statements?

No, Python doesn’t have switch statements.

#### How do you implement the functionality of switch statements in Python?

We can implement the functionality of switch statements using if and elif statements.

*>>> if a == 1:*

*... print(...)*

*... elif a == 2:*

*... print(....)*

#### What are break and continue statements?

break – the break statement is used to terminate the running loop. The execution of the code will jump to the outside of the break loop.

continue – the continue statement is used to skip the execution of the remaining code. The code after the continue statement doesn’t execute in the current iteration, and the execution goes to the next iteration.

#### When is the code in else executed with while and for loops?

The code inside the else block with while and for loops is executed after executing all iterations. And the code inside the else block doesn’t execute when we break the loops.

#### What are list and dictionary comprehensions?

List and dictionary comprehensions are syntactic sugar for the for-loops.

*>>> a = [i for i in range(10)]*

*>>> a*

*[0, 1, 2, 3, 4, 5, 6, 7, 8, 9]*

*>>> a = {i: i + 1 for i in range(10)}*

*>>> a*

*{0: 1, 1: 2, 2: 3, 3: 4, 4: 5, 5: 6, 6: 7, 7: 8, 8: 9, 9: 10}*

*>>>*

#### How does the range function work?

The range function returns the sequence of numbers between the start to stop with a step increment. The syntax of the range function is range(start, stop[, step]).

The stop argument is mandatory. The arguments start and step are optional. The default value of start and step are 0 and 1, respectively.

#### What are the parameters and arguments?

Parameters are the names listed in the function definition.

Arguments are the values passed to the function while invoking.

#### What are the different types of arguments in Python?

There are mainly four types of arguments. They are positional arguments, default arguments, keyword arguments, and arbitrary arguments.

Positional Arguments: the normal arguments that we define in user-defined functions are called positional arguments. All positional arguments are required while invoking the function.

#### What is the lambda function?

Lambda functions are small anonymous functions in Python. It has single expressions and accepts multiples arguments.

#### What’s the difference between normal function and lambda function?

The functionality of both normal functions and lambda functions are similar. But, we need to write some extra code in normal functions compared to lambda functions for the same functionality.

Lambda functions come in handy when there is a single expression.

#### What is the pass keyword used for?

The pass keyword is used to mention an empty block in the code. Python doesn’t allow us to leave the blocks without any code. So, the pass statement allows us to define empty blocks (when we decide to fill the code later).

#### What is a recursive function?

The function calling itself is called a recursive function.

#### What are packing operators in Python? How to use them?

The packing operators are used to collect multiple arguments in functions. They are known as arbitrary arguments.

Note: you can refer to [this article](https://geekflare.com/python-unpacking-operators/) for more info on packing operators in Python.