Q1. What is constructor in python.

Ans. A constructor is a special type of method (function) which is used to initialize the instance members of the class.

In C++ or Java, the constructor has the same name as its class, but it treats constructor differently in Python. It is used to create an object.

Constructors can be of two types.

Parameterized Constructor

Non-parameterized Constructor

Constructor definition is executed when we create the object of this class. Constructors also verify that there are enough resources for the object to perform any start-up task.

In Python, the method the **\_\_init\_\_()** simulates the constructor of the class. This method is called when the class is instantiated. It accepts the **self**-keyword as a first argument which allows accessing the attributes or method of the class.

We can pass any number of arguments at the time of creating the class object, depending upon the **\_\_init\_\_()** definition. It is mostly used to initialize the class attributes. Every class must have a constructor, even if it simply relies on the default constructor.

Q2. What is destructor in python.

Ans. The users call Destructor for destroying the object. In Python, developers might not need destructors as much it is needed in the C++ language. This is because Python has a garbage collector whose function is handling memory management automatically.

In this article, we will discuss how the destructors in Python works and when the users can use them.

The **\_\_del\_\_()** function is used as the destructor function in [Python](https://www.javatpoint.com/python-tutorial). The user can call the **\_\_del\_\_()** function when all the references of the object have been deleted, and it becomes garbage collected.

the garbage collector does not know the order in which the object should be destroyed, so it leaves them. So, if the instances of the users are involved in this circular reference, they will remain stored in the memory for as long as the application will run.

Q3. Explain the magic method.

Ans. To add "magic" to the class we create, we can define special methods called "magic methods." For example, the magic methods \_\_init\_\_ and \_\_str\_\_are always wrapped by double underscores from both sides. By granting us accessibility to Python's built-in syntax tools, magic methods can improve the structure of our classes.

We can integrate Python's built-in classes with our classes. The class which has inherited from the built-in class is known as a child class. A child class has access to all of the attributes of the parent class, including its methods. By utilizing the essential built-in features, we can customize some of the tasks of our class by using magic methods.

In Python, magic methods (also known as "dunder" methods, short for "double underscore")

are special methods that have double underscores at the beginning and end of their names.

These methods are used to define the behavior of objects in Python, and they are called

automatically in response to specific events or operations.

Some of the most commonly used magic methods include:

\_\_init\_\_: A constructor method that is called when an object of the class is created.

It is used to initialize the object's state.

\_\_str\_\_: A method that returns a string representation of the object.

It is used to define what a string representation of the object should look like.

\_\_len\_\_: A method that returns the length of the object.

It is used to define the length of objects that have a notion of length.

\_\_del\_\_: A destructor method that is called when an object is about to be destroyed.

It is used to clean up any resources used by the object.

These are just a few examples of magic methods in Python.

There are many other magic methods available, and they can be used to define a wide variety of

behaviors for objects in Python. When writing classes, you can use magic methods to define custom

behavior for your objects, making it easier to work with them in a consistent and intuitive way.

Q4. Differentiate between old class style and new class style.

Ans. In Python, classes can be divided into two types: old-style and new-style. The main difference between the two is that new-style classes inherit from a built-in base class, **object**, whereas old-style classes do not. This means that new-style classes have access to certain features, such as the **\_\_class\_\_** attribute and the **super()** function, that are not available in old-style classes.

In Python, the old-style classes and new-style classes refer to the two different styles of class

definition that were used in different versions of Python.

Old-style classes, also known as classic classes, were the only type of classes available in

Python versions prior to 2.2. They do not inherit from the built-in object class and do not

support some of the advanced features that are available in new-style classes.

New-style classes were introduced in Python 2.2 and are the recommended way to define classes

in Python. They inherit from the object class and provide additional features such as better

support for subclassing, improved method resolution order, and additional magic methods.

In summary, new-style classes are the recommended way to define classes in Python, as they provide

additional features and better support for subclassing compared to old-style classes. If you are

using a recent version of Python, it is recommended to use new-style classes in your code.

Q5. Differentiate between list and tuples (terms of memory).

Ans.

|  |  |  |
| --- | --- | --- |
| Python Lists | Python | Tuples |
| 1 | List are mutable | Tuples are immutable | Tuples are immutable |
| 2 | Iterations are time-consuming | Iterations are comparatively Faster | Iterations are comparatively Faster |
| 3 | Inserting and deleting items is easier with a list. | Accessing the elements is best accomplished with a tuple data type. | Accessing the elements is best accomplished with a tuple data type. |
| 4 | Lists consume more memory | Tuple consumes less than the list | Tuple consumes less than the list |
| 5 | Lists have several built-in methods. | A tuple does not have many built-in methods because of immutability | A tuple does not have many built-in methods because of immutability |
| 6 | A unexpected change or error is more likely to occur in a list. | In a tuple, changes and errors don't usually occur because of immutability. | In a tuple, changes and errors don't usually occur because of immutability. |

Q6. Write a note on session vs cookies.

Ans.

|  |  |
| --- | --- |
| **Session** | **Cookies** |
| A session stores the variables and their values within a file in a temporary directory on the server. | Cookies are stored on the user's computer as a text file. |
| The session ends when the user logout from the application or closes his web browser. | Cookies end on the lifetime set by the user. |
| It can store an unlimited amount of data. | It can store only limited data. |
| We can store as much data as we want within a session, but there is a maximum memory limit, which a script can use at one time, and it is 128 MB. | The maximum size of the browser's cookies is 4 KB. |
| We need to call the session\_start() function to start the session. | We don't need to call a function to start a cookie as it is stored within the local computer. |
| In PHP, to set a session data, the $\_SESSION global variable is used. | In PHP, to get the data from cookies, the $\_COOKIE global variable is used. |
| In PHP, to destroy or remove the data stored within a session, we can use the session\_destroy() function, and to unset a specific variable, we can use the unset() function. | We can set an expiration date to delete the cookie's data. It will automatically delete the data at that specific time. There is no particular function to remove the data. |
| Sessions are more secured compared to cookies, as they save data in encrypted form. | Cookies are not secure, as data is stored in a text file, and if any unauthorized user gets access to our system, he can temper the data. |

Q7. Where is a cookie stored in the browser.

Ans. They are stored in **random access memory** and are never written to the hard drive.

Q8. Write the main difference between https and http. A cookie is stored in the browser's local file system on the client's computer.

It is stored as a small text file that is associated with a specific website or domain.

The browser uses the information in the cookie to send the cookie data back to the server

with each subsequent request to the website. This allows the server to maintain state and

persist data across multiple requests from the same client. The location of the cookie files

depends on the specific browser and operating system being used, but they are typically stored in a

dedicated directory or folder within the browser's local file system.

Ans.

|  |  |
| --- | --- |
| **HTTP** | **HTTPS** |
| The full form of HTTP is the Hypertext Transfer Protocol. | The full form of HTTPS is Hypertext Transfer Protocol Secure. |
| It is written in the address bar as http://. | It is written in the address bar as https://. |
| The HTTP transmits the data over port number 80. | The HTTPS transmits the data over port number 443. |
| It is unsecured as the plain text is sent, which can be accessible by the hackers. | It is secure as it sends the encrypted data which hackers cannot understand. |
| It is mainly used for those websites that provide information like blog writing. | It is a secure protocol, so it is used for those websites that require to transmit the bank account details or credit card numbers. |
| It is an application layer protocol. | It is a transport layer protocol. |
| It does not use SSL. | It uses SSL that provides the encryption of the data. |
| Google does not give the preference to the HTTP websites. | Google gives preferences to the HTTPS as HTTPS websites are secure websites. |
| The page loading speed is fast. | The page loading speed is slow as compared to HTTP because of the additional feature that it supports, i.e., security. |