PYTHON BASIC

What is Python? What are the benefits of using Python?

Python is a programming language with objects, modules, threads, exceptions, and automatic memory management. The benefits of pythons are that it is simple and easy, portable, extensible, build-in data structure, and it is open-source.

What is pickling and unpickling?

Pickle module accepts any Python object and converts it into a string representation and dumps it into a file by using dump function. This process is called pickling. While the process of retrieving original Python objects from the stored string representation is called unpickling.

How is Python interpreted?

Python language is an interpreted language. Python program runs directly from the source code. It converts the source code that is written by the programmer into an intermediate language, which is again translated into machine language that has to be executed.

How is memory managed in Python?

Python memory is managed by Python private heap space. All Python objects and data structures are located in a private heap. The programmer does not have an access to this private heap, and the interpreter takes care of this Python private heap.

The allocation of Python heap space for Python objects is done by the Python memory manager. The core API gives access to some tools for the programmer to code.

Python also has an inbuilt garbage collector, which recycles all the unused memory and frees the memory and makes it available to the heap space.

What are Python decorators?

A Python decorator is a specific change that we make in Python syntax to alter functions easily.

What is the difference between list and tuple?

The difference between list and tuple is that list is mutable while tuple is not. Tuple can be hashed, for example., as a key for dictionaries.

How are arguments passed by value or by reference?

Everything in Python is an object, and all variables hold references to the objects. The reference values are according to the functions. Therefore, you cannot change the value of the

references. However, you can change the objects if it is mutable.

What is Dict and List comprehensions are?

They are syntax constructions to ease the creation of a Dictionary or List based on existing iterable.

What are built-in type does python provides?

Python provides two built-in types: 1) Mutable and 2) Immutable.

Mutable built-in types are:

List Sets Dictionaries

Immutable built-in types

Strings Tuples Numbers

Explain namespace in Python

In Python, every name introduced has a place where it lives and can be hooked for. This is known as a namespace. It is like a box where a variable name is mapped to the object placed. Whenever the variable is searched out, this box will be searched to get the corresponding object.

What is lambda in Python?Why lambda forms in python do not have statements?

It is a single expression anonymous function often used as inline function.

A lambda form in python does not have statements as it is used to make new function object and then return them at runtime.

Explain slicing in Python?

A mechanism to select a range of items from sequence types like list, tuple, strings etc., is known as slicing.

What is docstring in Python?

A Python documentation string is known as docstring, it is a way of documenting Python functions, modules, and classes.

What is negative index in Python?

Python sequences can be index in positive and negative numbers. For positive index, 0 is the first index, 1 is the second index, and so forth. For the negative index, (-1) is the last index, and

(-2) is the second last index, and so forth.

How can you convert a number to a string?

In order to convert a number into a string, use the inbuilt function str(). If you want a octal or hexadecimal representation, use the inbuilt function oct() or hex().

What is the difference between xrange and range?

Xrange returns the xrange object while range returns the list and uses the same memory and no matter what the range size is.

What is module and package in Python?

In Python, module is the way to structure a program. Each Python program file is a module, which imports other modules like objects and attributes. The folder of Python program is a package of modules. A package can have modules or subfolders.

What are the rules for local and global variables in Python?

Here are the rules for local and global variables in Python:

Local variables: If a variable is assigned a new value anywhere within the function's body, it's assumed to be local.

Global variables: Those variables that are only referenced inside a function are implicitly global.

How can you share global variables across modules?

To share global variables across modules within a single program, create a special module. Import the config module in all modules of your application. The module will be available as a global variable across modules.

How can you access a module written in Python from C?

Module = PyImport_ImportModule("<modulename>");

What is the use of // operator in Python?

It is a Floor Divisionoperator, which is used for dividing two operands with the result as a quotient showing only digits before the decimal point. For instance, 10//5 = 2 and 10.0//5.0 = 2.0.

Mention five benefits of using Python

Mention the use of the split function in Python

The use of the split function in Python is that it breaks a string into shorter strings using the defined separator. It gives a list of all words present in the string.

Explain Flask and its benefits

Flask is a web micro framework for Python based on “Werkzeug, Jinja 2 and good intentions” BSD licensed. Werkzeug and jingja are two of its dependencies.

Flask is part of the micro-framework. Which means it will have little to no dependencies on external libraries. It makes the framework light while there is a little dependency to update and less security bugs.

What is the difference between Django, Pyramid, and Flask?

Flask is a “microframework” primarily build for a small application with simpler requirements. In a flask, you don’t have to use external libraries. Flask is ready to use.

Pyramids are built for larger applications. It provides flexibility and lets the developer use the right tools for their project. The developer can choose the database, URL structure, templating style, and more. Like Pyramid, Django can also be used for larger applications. It includes an ORM.

Explain how you can access sessions in Flask?

A session basically allows you to remember information from one request to another. In a flask, it uses a signed cookie so the user can look at the session contents and modify. The user can modify the session if only it has the secret key Flask.secret_key.

Explain database connection in Python Flask?

Flask supports database-powered applications (RDBS). Such a system requires creating a schema, which requires piping the shema.sql file into a sqlite3 command. So you need to install sqlite3 command in order to create or initiate the database in Flask.

Flask allows to request database in three ways

before_request(): It is called before a request and pass no arguments

after_request(): It is called after a request and pass the response that will be sent to the client

teardown_request(): It is called in a situation when exception is raised, and response is not guaranteed. They are called after the response has been constructed. They are not allowed to modify the request, and their values are ignored.

If you have multiple Memcache servers, and one of them fails that contain data, will it try to

get them?

The data in the failed server won't get removed, but there is a provision for auto-failure, which you can configure for multiple nodes. Fail-over can be triggered during any kind of socket or Memcached server level errors and not during normal client errors like adding an existing key, etc.

Explain how you can minimize the Memcached server outages in your Python Development?

When one instance fails, several of them goes down, this will put a larger load on the database server when lost data is reloaded as the client make a request. To avoid this, if your code has been written to minimize cache stampedes, then it will leave a minimal impact

Another way is to bring up an instance of memcached on a new machine using the lost machine's IP address

Code is another option to minimize server outages as it gives you the liberty to change the Memcached server list with minimal work

Setting timeout value is another option that some Memcached clients implement for Memcached server outage. When your Memcached server goes down, the client will keep trying to send a request till the time-out limit is reached.

Explain what is Dogpile effect? How can you prevent this effect?

Dogpile effect is referred to the event when cache expires, and websites are hit by the multiple requests made by the client at the same time. This effect can be prevented by using a semaphore lock. In this system, when the value expires, the first process acquires the lock and starts generating a new value.

Explain how memcached should not be used in your Python project?

Here are the ways you should not use memcached in your Python project:

Memcached common misuse is to use it as a data store and not as a cache

Never use Memcached as the only source of the information you need to run your application. Data should always be available through another source as well

Memcached is just a key or value store and cannot perform query over the data or iterate over the contents to extract information.

Memcached does not offer any form of security either in encryption or authentication.

Explain While loop in Python with example

While loop does the exact same thing what “if statement” does, but instead of running the code block once, they jump back to the point where it began the code and repeat the whole process again.

The syntax of while loop is as follows:

while expression

Statement

The example of while loop is as follows:

```
x=0

#define a while loop

while(x <4):

    print(x)

    x = x+1
```

What is enumerate() in Python?

Enumerate() in Python is a built-in function used for assigning an index to each item of the iterable object. It adds a loop on the iterable objects while keeping track of the current item and returns the object in an enumerable form. This object can be used in a for loop to convert it into a list by using list() method.

What is Tuple Matching in Python?

Tuple Matching in Python is a method of grouping the tuples by matching the second element in the tuples. It is achieved by using a dictionary by checking the second element in each tuple in python programming. However, we can make new tuples by taking portions of existing tuples.

Syntax:

```
Tup = ('Jan','feb','march')
```

To write an empty tuple, you need to write as two parentheses containing nothing-

tup1 = ();

Explain Dictionary in Python with example

A Dictionary in Python is the unordered and changeable collection of data values that holds key-value pairs. Each key-value pair in the dictionary maps the key to its associated value making it more optimized. A Dictionary in python is declared by enclosing a comma-separated list of key-value pairs using curly braces({}). Python Dictionary is classified into two elements: Keys and Values.

Syntax for Python Dictionary:

Dict = { ' Tim': 18, xyz,.. }

Example:

Dict = {'Tim': 18,'Charlie':12,'Tiffany':22,'Robert':25}

print((Dict['Tiffany']))

How can you copy the entire dictionary to a new dictionary?

You can also copy the entire dictionary to a new dictionary. For example, here we have copied our original dictionary to the new dictionary name “Boys” and “Girls”.

Dict = {'Tim': 18,'Charlie':12,'Tiffany':22,'Robert':25}

Boys = {'Tim': 18,'Charlie':12,'Robert':25}

Girls = {'Tiffany':22}

studentX=Boys.copy()

studentY=Girls.copy()

print(studentX)

print(studentY)

How can you sort elements in Python dictionary?

In the dictionary, you can easily sort the elements. For example, if we want to print the name of the elements of our dictionary alphabetically, we have to use for loop. It will sort each element of the dictionary accordingly.

Dict = {'Tim': 18,'Charlie':12,'Tiffany':22,'Robert':25}

```
Boys = {'Tim': 18,'Charlie':12,'Robert':25}

Girls = {'Tiffany':22}

Students = list(Dict.keys())

Students.sort()

for S in Students:

    print(":".join((S,str(Dict[S]))))
```

What are all dictionary methods:

copy() update() items() sort() len() cmp() Str()

Explain Arithmetic operators

Arithmetic Operators perform various arithmetic calculations like addition, subtraction, multiplication, division, %modulus, exponent, etc. There are various methods for arithmetic calculation in Python, like you can use the eval function, declare variable & calculate, or call functions.

How can you reverse array in Python?

You can use reverse() to reverse array in Python.

Explain Inheritance

Inheritance is a feature used in object-oriented programming; it refers to defining a new class with less or no modification to an existing class. The new class is called the derived class, and from one which it inherits is called the base. Python supports inheritance; it also supports multiple inheritances. A class can inherit attributes and behavior methods from another class called subclass or heir class.

Give example of Python constructors

```
class User:

  name = ""

  def __init__(self, name):

    self.name = name

  def sayHello(self):
```

```
    print("Welcome, " + self.name)

User1 = User("Alex")

User1.sayHello()
```

What are the common examples of exceptions in Python?

The common examples of exceptions in Python are:

Division by Zero

Accessing a file that does not exist.

Addition of two incompatible types

Trying to access a nonexistent index of a sequence

Removing the table from the disconnected database server.

ATM withdrawal of more than the available amount

Explain important Python errors

The important Python errors are 1) ArithmeticError, 2) ImportError, and 3) IndexError.

ArithmeticError: ArithmeticError act as a base class for all arithmetic exceptions. It is raised for errors in arithmetic operations.

ImportError: ImportError is raised when you are trying to import a module which does not present. This kind of exception occurs if you have made a typing mistake in the module name or the module which is not present in the standard path.

IndexError: An IndexError is raised when you try to refer a sequence which is out of range.