1) 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.

2) What is PEP 8?

PEP 8 is a coding convention, a set of recommendation, about how to write your Python code more readable.

3) 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.

4) 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.

5) 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.

6) What are the tools that help to find bugs or perform the static analysis?

PyChecker is a static analysis tool that detects the bugs in Python source code and warns about the style and complexity of the bug. Pylint is another tool that verifies whether the module meets the coding standard.

7) What are Python decorators?

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

8) 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.

9) 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.

10) What is Dict and List comprehensions are?

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

11) 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

Immutable built-in types are:

- Strings
- Tuples
- Numbers

12) 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.

13) What is lambda in Python?

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

14) Why lambda forms in python do not have statements?

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

15) Explain pass in Python

Pass means no-operation Python statement, or in other words, it is a place holder in a compound statement, where there should be a blank left, and nothing has to be written there.

16) In Python what are iterators?

In Python, iterators are used to iterate a group of elements, containers like a list.

17) What is the unit test in Python?

A unit testing framework in Python is known as unittest. It supports sharing of setups, automation testing, shutdown code for tests, aggregation of tests into collections, etc.

18) Explain slicing in Python?

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

19) What are generators in Python?

The way of implementing iterators are known as generators. It is a normal function except that it yields expression in the function.

20) What is docstring in Python?

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

21) How can you copy an object in Python?

To copy an object in Python, you can try a copy.copy () or copy.deepcopy() for the general case. You cannot copy all objects but most of them.

22) 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.

23) 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().

24) 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.

25) 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.

26) 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.

27) 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.

28) Explain how can you make a Python Script executable on Unix?

To make a Python Script executable on Unix, you need to do two things,

Script file's mode must be executable, and the first line must begin with # (#!/usr/local/bin/python)

29) Explain how to delete a file in Python?

By using a command os.remove (filename) or os.unlink(filename)

30) Explain how can you generate random numbers in Python?

To generate random numbers in Python, you need to import command as

```
import random
random.random()
```

This returns a random floating-point number in the range [0,1)

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

You can access a module written in Python from C by following method,

```
Module = PyImport_ImportModule("<modulename>");
```

32) 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.

33) Mention five benefits of using Python

Here are the five benefits of using Python:

- Python comprises of a huge standard library for most Internet platforms like Email, HTML, etc.
- Python does not require explicit memory management as the interpreter itself allocates the memory to new variables and free them automatically
- Provide easy readability due to use of square brackets
- Easy-to-learn for beginners
- Having the built-in data types saves programming time and effort from declaring variables

34) 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.

35) 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.

36) 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.

37) What is Flask-WTF and what are their features?

Flask-WTF offers simple integration with WTForms. Features include for Flask WTF are:

- Integration with WTFforms
- Secure form with CSRF token
- Global CSRF protection
- Internationalization integration
- Recaptcha supporting
- File upload that works with Flask Uploads

38) Explain what is the common way for the Flask script to work?

The common way for the flask script to work is:

- Either it should be the import path for your application

- Or the path to a Python file

39) 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.

40) Is Flask an MVC model, and if yes give an example showing MVC pattern for your application?

Basically, Flask is a minimalistic framework that behaves same as MVC framework. So MVC is a perfect fit for Flask, and the pattern for MVC we will consider for the following example

from flask import Flask
app = Flask(_name_)
@app.route(“/”)
Def hello():
return “Hello World”
app.run(debug = True)
	In this code your,
	• Configuration part will be
	from flask import Flask
	app = Flask(_name_)
	• View part will be
	@app.route(“/”)
	Def hello():
	return “Hello World”
	• While you model or main part will be
	app.run(debug = True)

41) 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.

42) 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.

43) 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.

44) 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.

45) 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.

46) What is Python If Statement?

Python if Statement is used for decision-making operations. It contains a body of code that runs only when the condition given in the if statement is true. If the condition is false, then the optional else statement runs, which contains some code for the else condition.

When you want to justify one condition while the other condition is not true, then you use Python if-else statement.

Python if Statement Syntax:

```
if expression
 Statement
else
 Statement
```

Let’s see an example of Python if else Statement:

```
def main():
    x,y =2,8

    if(x < y):
        st= "x is less than y"
    print(st)

if __name__ == "__main__":
    main()
```

47) 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
```

48) 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.

51) 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.

55) 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.

57) What are all dictionary methods:

Here is the list of dictionary methods:

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

58) Explain Arithmetic operators with example

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.

62) Explain arrays in Pythons

A Python Array is a collection of a common type of data structures having elements with the same data type. It is used to store collections of data. In Python programming, arrays are handled by the “array” module. If you create arrays using the array module, elements of the array must be of the same numeric type.

70) 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.

72) How can you access values in string?

Python does not support a character type, these are treated as strings of length one, also considered as a substring.

75) What is timmer method in Python?

Timer is a method available with Threading, and it helps to get the same functionality as Python time sleep.

76) What is decorator in python?

Python decorators are functions or other callable objects that can be used to add functionalities to another function without modifying its source code. A decorator in python accepts a function as an input argument, adds some functionalities to it and returns a new function with the modified functionalities. Implementation of decorators in python requires knowledge of different concepts such as first class objects and nested functions.

In Python, **first class objects** are those objects that

- can be passed to a function as a parameter.
- can be returned from a function.
- can be assigned to a variable.

Nested functions are the functions defined inside another function.

78) 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

79) 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.

86) Explain slicing of matrix with example

Slicing will return you the elements from the matrix based on the start /end index given.

The syntax for slicing is:

[start:end]

87) Explain what AWS is?

AWS stands for Amazon Web Service; it is a collection of remote computing services also known as a cloud computing platform.

- **Simple Storage Device or (S3):** It is a storage device and the most widely used AWS service. S3 stands for Simple Storage Service. You can use the S3 interface to store and retrieve any amount of data, at any time and from anywhere on the web.
- **Elastic Compute Cloud (EC2):** It provides on-demand computing resources for hosting applications. It is handy in case of unpredictable workloads.

88) Mention what the difference between Amazon S3 and EC2 is?

- The difference between EC2 and Amazon S3 is that

EC2	S3

• It is a cloud web service used for hosting your application	• It is a data storage system where any amount of data can be stored
• It is like a huge computer machine which can run either Linux or Windows and can handle applications like PHP, Python, Apache, or any databases	• It has a REST interface and uses secure HMAC-SHA1 authentication keys

89) What is AMI?

AMI stands for Amazon Machine Image. It's a template that provides the information (an operating system, an application server, and applications) required to launch an instance, which is a copy of the AMI running as a virtual server in the cloud. You can launch instances from as many different AMIs as you need.

90) What is AWS Lambda?

Lambda is an Amazon compute service which allows you to run code in the AWS Cloud without managing servers.

91) What is Jenkins and why use it?

Ans: Jenkins is one of the leading open-source continuous integration tools. The main functionality of this tool is to keep track of the version control system and monitor the build system and provide notifications and reports to alert. It enables you to deliver software by integrating with a large number of testing and deployment technologies.

The following are the reasons to use Jenkins:

- It possesses an installer package for major operating systems.

- Integrates individual projects for a larger purpose
- To keep your team in sync
- Troubleshoot and audit past jobs effortlessly
- Provides accurate data support for project management

92) What is continuous integration?

Ans: Continuous integration is a process of continuously checking the developer's code into a version control system several times a day and automating the build to check and detect bugs in the written code. Continuous Integration includes the following:

- Development and Compilation
- Database Integration
- Unit Testing
- Production Deployment
- Code Labeling
- Functional Testing
- Generating and Analyzing Reports

93) Name some of the plugins in Jenkin?

Ans: Some of the important plugins in Jenkin includes:

- Maven 2 project
- Amazon EC2
- HTML publisher
- Copy Artifact
- Join
- Green Balls
- Git plugin
- Multi job plugin
- Test Results Analyzer
- Metrics

94) What is a Jenkins Pipeline?

Ans: Jenkins Pipeline is a set of features of Jenkins, which are installed as plugins to enable continuous delivery pipeline implementation. These are the automated processes to get the software from source control through deployment to end-users.

95) what is version control?

- VCS or version control is a management system that tracks changes in a computer file.
- It is a software tool that records and manages changes to the source code over a period of time.

- VCS or version control is a management system that tracks changes in a computer file.

96) git and github

Git is a version control system that lets you manage and keep track of your source code and history whereas GitHub is a cloud-based hosting service that lets you manage git repositories.

GitHub is a hosting service for Git repositories. So they are not the same thing: Git is the tool, GitHub is the service for projects that use Git.

A unit test is a way to test a unit, the smallest code in a system that can logically be isolated. This is a function, a subroutine, a procedure, or a property in most programming languages.

Q1). Tell me about Unit Testing in brief.

Ans:-Unit Testing is used to check the independent modules of a software app during the development phase. An independent module can be anything like procedure, function, etc. Unit testing is done by developers and testers together before the integration testing. They have to write unit test cases as well if needed.

Q2). What is the total number of phases in a Unit Test Case?
Ans:-The working of a unit test case can be divided into 3 phases. At the first stage, it will initialize the specific module of a software app that you want to test. In the second stage, it will execute the test case. In the end, it will analyze the final output.

Q3). What are the various types of Unit Testing for a software app?
Ans:-

- *State-based Unit Testing*
- *Interaction-based Unit Testing*

Q4). What do you know about state-based Unit Testing?
Ans:-If you want to check if the final output is right or not, then it becomes state-based.

Q5). Tell me about interaction-based unit testing in brief.
Ans:-If you want to check the behavior of functions or procedures, whether they are invoked in the right way or not then it is interaction-based.

Q6). Have you ever used or worked on Unit test frameworks? If yes, name them.
Ans:-Yes, I do have practical knowledge of unit test frameworks, Junit, and TestNG.

Q7). Tell me about the Junit testing framework.
Ans:-Several test cases need to be executed repeatedly. If you need test cases for repeated execution then the Junit framework can help you.

Q8). Who can perform Unit Testing?
Ans:-Unit Testing is generally done at the development phase so that it can be performed by developers. At the same time, if developers are occupied with other development tasks then unit testing is generally performed by automation engineers and QA experts.

Q9). What do you know about the term Refactoring?

Ans:-If you want to revamp any existing code, then this technique is used. It is generally done in small steps where only the code is changed, not the functionality or the logic. It helps in bug fixing too.

Q10). How is unit testing different from Integration Testing?

Ans:-Users are generally confused between unit testing and integration testing. Here is a quick comparison between the two for your reference.

Q16). What is Faking?

Ans:- It is a class that is good for implementing interfaces without any logic. It just returns good or bad based on the implementation findings.

Q17). What is Mocking?

Ans:- It is a class that is suitable for exceptional handling, and it will give you a detailed idea of when a particular method was called. In case a method was not called by this class then you will be notified for the same.

Q18). What is Stubbing?

Ans:- Stubs can set dynamic values when exceptions are thrown by methods. It works similar to mock classes but does not give any idea of whether either methods were called or not.

98) Day to day tasks of a Python Developer

- Write new code to create a new application
- Create scripts to execute highly repeatable jobs
- Debug code which has errors
- Rewrite old code to optimise it with modern procedures