**Python Basic Interview Questions**

**1) What is Python?**

**Ans:** Python is a high-level and object-oriented programming language with unified semantics designed primarily for developing apps and web. It is the core language in the field of Rapid Application Development (RAD) as it offers options such as dynamic binding and dynamic typing.

**2) What are the benefits of Python?**

**Ans:** The benefits of Python are as follows:

* **Speed and Productivity:** Utilizing the productivity and speed of Python will enhance the process control capabilities and possesses strong integration.
* **Extensive Support for Libraries:** Python provides a large standard library that includes areas such as operating system interfaces, web service tools, internet protocols, and string protocols. Most of the programming tasks are already been scripted in the standard library which reduces effort and time.
* **User-friendly Data Structures:** Python has an in-built dictionary of data structures that are used to build fast user-friendly data structures.
* **Existence of Third Party Modules:** The presence of third party modules in the [Python Package Index](https://pypi.org/) (PyPI) will make Python capable to interact with other platforms and languages.
* **Easy Learning:** Python provides excellent readability and simple syntaxes to make it easy for beginners to learn.

**3) What are the key features of Python?**

**Ans:** The following are the significant features of Python, and they are:

* **Interpreted Language:** Python is an interpreted language that is used to execute the code line by line at a time. This makes debugging easy.
* **Highly Portable:** As Python can run on different platforms such as Unix, Macintosh, Linux, Windows, and so on. So, we can say that it is a highly portable language.
* **Extensible:** It ensures that the Python code can be compiled on various other languages such as C, C++ and so on.
* **GUI programming Support:** It implies that the Python provides support to develop graphical user interfaces

**4) What type of language is Python? Programming or Scripting?**

**Ans:** Python is suitable for scripting, but in general it is considered as a general-purpose programming language.

**5) What are the applications of Python?**

**Ans:** The applications of Python are as follows:

* GUI based desktop applications
* Image processing applications
* Business and Enterprise applications
* Prototyping
* Web and web framework applications

**6) What is the difference between list and tuple in Python?**

**Ans:** The difference between tuple and list are as follows:

|  |  |
| --- | --- |
| **List** | **Tuple** |
| The list is mutable (can be changed) | A tuple is immutable (remains constant) |
| These lists performance is slower | Tuple performance is faster when compared to lists |
| Syntax: list\_1 = [20, ‘Mindmajix’, 30] | Syntax: tup\_1 = (20, ‘Mindmajix’, 30) |

**7) What are the global and local variables in Python?**

**Ans: Global Variables in Python**: The variables that are declared outside the function are called global variables. These variables can be accessed or invoked by any function in the program.

**Example:**

|  |  |
| --- | --- |
| 1  2  3  4 | def v() :  print g  g = "welcome to mindmajix"  v() |

**Output:**

|  |  |
| --- | --- |
| 1 | Welcome to mindmajix |

**Local Variables in Python:** The variables that are declared inside a function are called local variables. These type of variables can be accessed only inside the function.

**8) Define PYTHON PATH?**

**Ans:** PYTHONPATH is an environmental variable that is used when we import a module. Suppose at any time we import a module, PYTHONPATH is used to check the presence of the modules that are imported in different directories. Loading of the module will be determined by interpreters.

**9) What are the two major loop statements?**

**Ans:**for and while

**10) What do you understand by the term PEP 8?**

**Ans:** PEP 8 is the Python latest coding convention and it is abbreviated as Python Enhancement Proposal. It is all about how to format your Python code for maximum readability.

**11) How memory management is done in Python?**

**Ans:**

* In Python memory management is done using private heap space. The private heap is the storage area for all the data structures and objects. The interpreter has access to the private heap and the programmer cannot access this private heap.
* The storage allocation for the data structures and objects in Python is done by the memory manager. The access for some tools is provided by core API for programmers to code.
* The built-in garbage collector in Python is used to recycle all the unused memory so that it can be available for heap storage area.

**12) Java vs Python**

**Ans:** The major difference between Java and Python are as follows:

|  |  |  |
| --- | --- | --- |
| **Function** | **Java** | **Python** |
| **Coding** | In Java, we need to write a long code to print something. | In Python coding is simple and smaller when compared to Java |
| **Syntax** | In Java we need to put a semicolon at the end of the statement and also code must be placed in curly braces. | Whereas, in Python indentation is mandatory as it improves the readability of the code. |
| **Dynamic** | In Java, we need to declare the type for each variable | In this case, codes are dynamically typed and this is also known as duck typing |
| **Easy to use** | Java is not easy to use because of its larger coding | In Python, it is very easy to code and perform very easily. |
| **Databases** | Java Database Connectivity (JDBC) is more popular and used most commonly. | In Python database access layers are weaker when compared to Java. |

**13) Define modules in Python?**

**Ans:** Module is defined as a file that includes a set of various functions and Python statements that we want to add in our application.

**Example of creating` a module:**

In order to create a module first, we need to save the code that we want in a file with .py extension.

Save the module with **module.py**

|  |  |
| --- | --- |
| 1  2 | def wishes(name):  Print("Hi, " + name) |

**14) What are the built-in types available in Python?**

**Ans:** The built-in types in Python are as follows:

* Integer
* Complex numbers
* Floating-point numbers
* Strings
* Built-in functions

**15) What are Python Decorators?**

**Ans:** Decorator is the most useful tool in Python as it allows programmers to alter the changes in the behavior of class or function.

**Example for Python Decorator is:**

|  |  |
| --- | --- |
| 1  2  3 | @gfg\_decorator  def hi\_decorator():      print("Gfg") |

**16) How do we find bugs and statistical problems in Python?**

**Ans:**We can detect bugs in python source code using a static analysis tool named PyChecker. Moreover, there is another tool called PyLint that checks whether the Python modules meet their coding standards or not.

**17) What is the difference between .py and .pyc files?**

**Ans:**.py files are Python source files. .pyc files are the compiled bytecode files that are generated by the Python compiler

**18) How do you invoke the Python interpreter for interactive use?**

**Ans:**By using python or pythonx.y we can invoke Python interpreter. where x.y is the version of the Python interpreter.

**19) Define String in Python?**

**Ans:**String in Python is formed using a sequence of characters. Value once assigned to a string cannot be modified because they are immutable objects. String literals in Python can be declared using double quotes or single quotes.

**Example:**

|  |  |
| --- | --- |
| 1  2 | print("Hi")  print('Hi') |

**20) What do you understand by the term namespace in Python?**

**Ans:** A namespace in Python can be defined as a system that is designed to provide a unique name for every object in python. Types of namespaces that are present in Python are:

* Local namespace
* Global namespace
* Built-in namespace

**Scope of an object in Python:**

Scope refers to the availability and accessibility of an object in the coding region.

**21) How do you create a Python function?**

**Ans:**Functions are defined using the def statement.

An example might be **def foo(bar):**

**22) Define iterators in Python?**

**Ans:** In Python, an iterator can be defined as an object that can be iterated or traversed upon. In another way, it is mainly used to iterate a group of containers, elements, the same as a list.

**23) How does a function return values?**

**Ans:**Functions return values using the return statement.

**24) Define slicing in Python?**

**Ans:** Slicing is a procedure used to select a particular range of items from sequence types such as Strings, lists, and so on.

**25) How can Python be an interpreted language?**

**Ans:** As in Python the code which we write is not machine-level code before runtime so, this is the reason why Python is called as an interpreted language.

**26) What happens when a function doesn’t have a return statement? Is this valid?**

**Ans:**Yes, this is valid. The function will then return a None object. The end of a function is defined by the block of code is executed (i.e., the indenting) not by any explicit keyword.

**27) Define package in Python?**

**Ans:** In Python packages are defined as the collection of different modules.

**28) How can we make a Python script executable on Unix?**

**Ans:** In order to make a Python script executable on Unix, we need to perform two things. They are:

Script file mode must be executable and

The first line should always begin with #.

**29) Which command is used to delete files in Python?**

**Ans:** OS.unlink(filename) or OS.remove(filename) are the commands used to delete files in Python Programming.

**Example:**

|  |  |
| --- | --- |
| 1  2 | import OS  OS.remove("abc.txt") |

**30) Define pickling and unpickling in Python?**

**Ans:**

**Pickling in Python:** The process in which the pickle module accepts various Python objects and converts into a string representation and dumps the file accordingly using dump function is called pickling.

**Unpickling in Python:** The process of retrieving actual Python objects from the stored string representation is called unpickling.

[Check Out Python Tutorials](https://mindmajix.com/python-tutorial)

**31) Explain the difference between local and global namespaces?**

**Ans:**Local namespaces are created within a function when that function is called. Global namespaces are created when the program starts.

**32) What is a boolean in Python?**

**Ans:** Boolean is one of the built-in data types in Python, it mainly contains two values, and they are true and false.

Python bool() is the method used to convert a value to a boolean value.

|  |  |
| --- | --- |
| 1 | Syntax for bool() method: bool([a]) |

**33) What is Python String format and Python String replace?**

**Ans:**

**Subscribe to our youtube channel to get new updates..!**

**Python String Format:** The String format() method in Python is mainly used to format the given string into an accurate output or result.

**Syntax for String format() method:**

|  |  |
| --- | --- |
| 1 | template.format(p0, p1, ..., k0=v0, k1=v1, ...) |

**Python String Replace**: This method is mainly used to return a copy of the string in which all the occurrence of the substring is replaced by another substring.

**Syntax for String replace() method:**

|  |  |
| --- | --- |
| 1 | str.replace(old, new [, count]) |

**34Q. Name some of the built-in modules in Python?**

**Ans:** The built-in modules in Python are:

* sys module
* OS module
* random module
* collection module
* JSON
* Math module

**35Q. What are the functions in Python?**

**Ans:** In Python, functions are defined as a block of code that is executable only when it is called. The def keyword is used to define a function in Python.

**Example:**

|  |  |
| --- | --- |
| 1  2  3 | def Func():  print("Hello, Welcome toMindmajix")  Func(); #calling the function |

**Output:** Hello, Welcome to Mindmajix

**36) What are Dict and List comprehensions in Python?**

**Ans:**These are mostly used as syntax constructions to ease the creation of list and dictionaries based on existing iterable.

**37Q. Define the term lambda?**

**Ans:**Lambda is the small anonymous function in Python that is often used as an inline function.

**38Q. When would you use triple quotes as a delimiter?**

**Ans:**Triple quotes ‘’”” or ‘“ are string delimiters that can span multiple lines in Python. Triple quotes are usually used when spanning multiple lines, or enclosing a string that has a mix of single and double quotes contained therein.

**39Q. Define self in Python?**

**Ans:** In Python self is defined as an object or an instance of a class. This self is explicitly considered as the first parameter in Python. Moreover, we can also access all the methods and attributes of the classes in Python programming using self keyword.

In the case of the init method, self refers to the newer creation of the object. Whereas in the case of other methods self refers to the object whose method was called.

**40Q. What is \_init\_?**

**Ans:** The \_init\_ is a special type of method in Python that is called automatically when the memory is allocated for a new object. The main role of \_init\_ is to initialize the values of instance members for objects.

**Example:**

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11 | class  Student:  def \_init\_ (self, name, age, marks):  self.name = name  self.age = age  self.marks = 950  S1 = Student("ABC", 22, 950)  # S1 is the instance of class Student.  # \_init allocates memory for S1.  print(S1.name)  print(S1.age)  print(S1.marks) |

**Output:**

|  |  |
| --- | --- |
| 1  2  3 | ABC  22  950 |

**41Q. Define generators in Python?**

**Ans:** The way of implementing an effective representation of iterators is known as generators. It is only the normal function that yields expression in the function.

**42Q. Define docstring in Python?**

**Ans:** The docstring in Python is also called a documentation string, it provides a way to document the Python classes, functions, and modules.

**43Q. How do we convert the string to lowercase?**

**Ans:** lower() function is used to convert string to lowercase.

**Example:**

|  |  |
| --- | --- |
| 1  2 | str = 'XYZ'  print(str.lower()) |

**Output:**

|  |  |
| --- | --- |
| 1 | xyz |

**44Q. How to remove values from a Python array?**

**Ans:** The elements can be removed from a Python array using remove() or pop() function. The difference between pop() and remove() will be explained in the below example.

**Example:**

|  |  |
| --- | --- |
| 1  2  3  4  5 | x = arr.array('d',  [ 1.0, 2.2, 3.4, 4.8, 5.2, 6.6, 7.3])  print(x.pop())  print(x.pop(3))  x.remove(1.0)  print(a) |

**Output:**

|  |  |
| --- | --- |
| 1  2  3 | 7.3  4.8  array(‘d’, [2.2, 3.4, 5.2, 6.6]) |

**45Q. What is Try Block?**

A block which is preceded by the try keyword is known as a try block

Syntax:

|  |  |
| --- | --- |
| 1  2  3 | try{     //statements that may cause an exception  } |

**46Q. Why do we use the split method in Python?**

**Ans:** split() method in Python is mainly used to separate a given string.

**Example:**

|  |  |
| --- | --- |
| 1  2 | x = "Mindmajix Online Training"  print(a.split()) |

**Output:**

|  |  |
| --- | --- |
| 1 | [‘Mindmajix’, ‘Online’, ‘Training’] |

**47Q. How can we access a module written in Python from C?**

**Ans:**We can access the module written in Python from C by using the following method.

|  |  |
| --- | --- |
| 1 | Module == PyImport\_ImportModule("<modulename>"); |

**48Q. How do you copy an object in Python?**

**Ans:**To copy objects in Python we can use methods called **copy.copy()** or **copy.deepcopy(**).

**49Q. How do we reverse a list in Python?**

**Ans:** By using list.reverse(): we can reverse the objects of the list in Python.

**Python Programming Interview Questions and Answers**

**50Q. How can we debug a Python program?**

**Ans:**  By using the following command we can debug the Python program

|  |  |
| --- | --- |
| 1 | $ python -m pdb python-script.py |

**51Q. Write a program to count the number of capital letters in a file?**

**Ans:**

|  |  |
| --- | --- |
| 1  2  3  4  5  6 | with open(SOME\_LARGE\_FILE) as countletter:  count = 0  text = countletter.read()  for character in text:  if character.isupper():  count += 1 |

**52Q. Write a program to display the Fibonacci sequence in Python?**

**Ans:**

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21 | # Displaying Fibonacci sequence  n = 10  # first two terms  n0 = 0  n1 = 1  #Count  x = 0  # check if the number of terms is valid  if n <= 0:     print("Enter positive integer")  elif n == 1:     print("Numbers in Fibonacci sequence upto",n,":")     print(n0)  else:     print("Numbers in Fibonacci sequence upto",n,":")     while x < n:         print(n0,end=', ')         nth = n0 + n1         n0 = n1         n1 = nth         x += 1 |

**Output:**

|  |  |
| --- | --- |
| 1 | 0, 1, 1, 2, 3, 5, 8, 13, 21, 34, |

**53Q. Write a program in Python to produce Star triangle?**

**Ans:** The code to produce star triangle is as follows:

|  |  |
| --- | --- |
| 1  2  3  4 | def pyfun(r):  for a in range(r):  print(' '\*(r-x-1)+'\*'\*(2\*x+1))  pyfun(9) |

**Output:**

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9 | \*         \*\*\*        \*\*\*\*\*       \*\*\*\*\*\*\*      \*\*\*\*\*\*\*\*\*     \*\*\*\*\*\*\*\*\*\*\*    \*\*\*\*\*\*\*\*\*\*\*\*\*   \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* |

**54Q. Write a program to check whether the given number is prime or not?**

**Ans:** The code to check prime number is as follows:

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17 | # program to check the number is prime or not  n1 = 409  # num1 = int(input("Enter any one number: "))  # prime number is greater than 1  if n1 > 1:  # check the following factors  for x is in range of(2,num1):  if (n1 % x) == 0:  print(n1,"is not a prime number")  print(x,"times",n1//x,"is",num)  break  else:  print(n1,"is a prime number")  # if input number is smaller than  # or equal to the value 1, then it is not prime number  else:  print(n1,"is not a prime number") |

**Output:**

|  |  |
| --- | --- |
| 1 | 409 is a prime number |

**55Q. Write Python code to check the given sequence is a palindrome or not?**

**Ans:**

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11 | # Python code to check a given sequence  #  is palindrome or not  my\_string1 = 'MOM'  My\_string1 = my\_string1.casefold()  # reverse the given string  rev\_string1  = reversed(my\_string1)  # check whether the string is equal to the reverse of it or not  if list(my\_string1) == list(rev\_string1):  print("It is a palindrome")  else:  print("It is not a palindrome") |

**Output:**

|  |  |
| --- | --- |
| 1 | it is a palindrome |

**56Q. Write Python code to sort a numerical dataset?**

**Ans:** The code to sort a numerical dataset is as follows:

|  |  |
| --- | --- |
| 1  2  3  4 | list = [ "13", "16", "1", "5" , "8"]  list = [int(x) for x in the list]  list.sort()  print(list) |

**Output:**

|  |  |
| --- | --- |
| 1 | 1, 5, 8, 13, 16 |

**57Q. What is the output of the following code?**

|  |  |
| --- | --- |
| 1  2 | x = ['ab','cd']  print(list(map(list, x))) |

**Ans:** The output of the following code is

|  |  |
| --- | --- |
| 1 | [ [‘a’, ‘b’], [‘c’, ‘d’] |

**Python Coding Interview Questions & Examples**

**58Q. What is the procedure to install Python on Windows and set path variable?**

**Ans:** We need to implement the following steps to install Python on Windows, and they are:

* First you need to install Python from https://www.python.org/downloads/
* After installing Python on your PC, find the place where it is located in your PC using the cmd python command.
* Then visit advanced system settings on your PC and add new variable. Name the new variable as PYTHON\_NAME then copy the path and paste it.
* Search for the path variable and select one of the values for it and click on ‘edit’.
* Finally we need to add a semicolon at the end of the value and if the semicolon is not present then type %PYTHON\_NAME%.

**59Q. Differentiate between SciPy and NumPy?**

**Ans:** The difference between SciPy and NumPy is as follows:

|  |  |
| --- | --- |
| **NumPy** | **SciPy** |
| Numerical Python is called NumPy | Scientific Python is called SciPy |
| It is used for performing general and efficient computations on numerical data which is saved in arrays. For example indexing, reshaping, sorting, and so on | This is an entire collection of tools in Python mainly used to perform operations like differentiation, integration and many more. |
| There are some of the linear algebraic functions present in this module but they are not fully fledged. | For performing algebraic computations this module contain some of the fully fledged operations |

**60Q. How do Python arrays and lists differ from each other?**

**Ans:** The difference between Python array and Python list are as follows:

|  |  |
| --- | --- |
| **Arrays** | **Lists** |
| Array is defined as a linear structure that is used to store only homogeneous data. | List are used to store arbitrary and heterogenous data |
| Since array stores only similar type of data so it occupies less amount of memory when compared to list. | List stores different types of data so it requires huge amount of memory |
| Length of the array is fixed at the time of designing and no more elements can be added in the middle. | Length of the list is no fixed, and adding items in the middle is possible in lists. |

**61Q. Can we make multi-line comments in Python?**

**Ans:** In python there is no specific syntax to display multi-line comments like other languages. In order to display mitli-line comments in Python, programmers use triple-quoted (docstrings) strings. If the docstring is not used as the first statement in the present method, it will not be considered by Python parser.

Explore Python Sample Resumes! Download & Edit, Get Noticed by Top Employers![**Download Now!**](https://mindmajix.com/python-sample-resumes)

**62Q. What is the difference between range and xrange?**

**Ans:** These both the methods are mainly used in Python to iterate the for loop for a fixed number of times. They differ only when we talk regarding Python versions.

The difference between range and xrange are as follows:

|  |  |
| --- | --- |
| **Range() method** | **Xrange() method** |
| The xrange() method is not supported in Python3 so that the range() method is used for iteration in for loops. | The xrange() method is used only in the Python version 2 for the iteration in for loops. |
| List is returned by this range() method | It only returns the generator object because it doesn’t produce a static list during run time. |
| It occupies a huge amount of memory as it stores the complete list of iterating numbers in memory. | It occupies less memory because it only stores one number at the time in memory. |

**Django Interview Questions**

**63Q. What is Django?**

**Ans:**Django is an advanced python web framework that supports agile growth and clean pragmatic design, built through experienced developers, this cares much about the trouble of web development, so you can concentrate on writing your app without wanting to reinvent that wheel.

**64Q. List the features of Django?**

**Ans:**

* Excellent documentation
* Python web framework
* SEO optimised
* High scalability
* Versatile in nature
* Offers high security
* Thoroughly tested
* Provides rapid Development

**65Q. Which level framework does Django belong to?**

**Ans:**Django is a high-level Python web framework which was developed for realistic design, clean, rapid development.

**66Q. What are the advantages of Django?**

**Ans:**

* One of the important advantages of Django is it is a framework of python language which is very simple to learn
* Django is a multifaceted framework
* When it comes to security Django is the best framework
* Scalability is added advantage of Django

**67Q. Why should we use Django framework?**

**Ans:**The main goal to design Django is to make it simple to its users, to do this Django uses:

* The principles concerning rapid development, which implies developers can complete more than one iteration at a time without beginning the full schedule from scratch;
* DRY philosophy — Do not Replicate Yourself — that means developers can reuse surviving code and also focus on the individual one.

**68Q. List the common security issues that can be avoided by using Django?**

**Ans:**Few common security issues that can be avoided by using Django are:

* Clickjacking
* Cross-site scripting and
* SQL injection

**69Q. List a few of the famous companies that are using Django?**

**Ans:**Few well-known companies that are using the Django framework are

* Instagram
* Spotify
* Mozilla
* Dropbox
* NASA

**70Q. What can we do with the Django framework?**

**Ans:**Here is an exciting reality: Django has first created to power a web application as a newspaper publisher, the Lawrence Journal-World. You all can demand it to be very good about handling projects by volumes from the text files, media, including extremely high traffic or else something that operates as a web publication.

**71Q. List steps for setting up static files in Django?**

**Ans:**There are only three main steps for setting up Django static files

* Firstly set STATIC\_ROOT in settings.py
* Run manage.py collect static
* Setting up a static file entry pythonAnywhere tab

**72Q. Is Django stable?**

**Ans:**Yes, Django is used by many famous companies because it is quite stable.

**73Q. Differentiate Django reusability code with other frameworks?**

**Ans:**Django web framework is operated and also maintained by an autonomous and non-profit organization designated as Django Software Foundation (DSF). The initial foundation goal is to promote, support, and advance this Django Web framework.

**74Q. How can we handle URLs in Django?**

**Ans:**

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9 | from django.contrib import admin    from django.urls import path      urlpatterns = [    path('appmajix/', appmajix.site.urls),    ] |

**75Q. List the mandatory files if Django project?**

**Ans:**

* manage.py
* settings.py
* \_\_init\_\_.py
* urls.py
* wsgi.py

**76Q. Explain about Django session?**

**Ans:**A session comprises a mechanism to store information on specific server-side at the interaction by the web application. By default, session reserves in the database and allows file-based and cache-based sessions.

**77Q. Why do we use a cookie in Django?**

**Ans:**A cookie is a piece of information that is stored in a client browser for a specific time. When the specific time is completed cookie gets automatically removed from the client browser.

**78Q. Mentions the methods used for setting and getting cookie values?**

**Ans:**The two methods to set and get cookie values are

* Set\_cookie this method is used to set the values of the cookie
* Get\_cookie this method is used to get the values of the cookie

**79Q. What is the use of Django-admin.py?**

**Ans:**Django-admin.py is a command-line argument which is utilised for administrative tasks

**80Q. What is the use of manage.py?**

**Ans:**It is an automatically built file inside each Django project. It is a flat wrapper encompassing the Django-admin.py. It possesses the following usage:

* It establishes your project's package on sys.path.
* It fixes the DJANGO\_SETTING\_MODULE environment variable to point to your project's setting.py file.

**81Q. Why is Django loosely packed?**

**Ans:**Django has described as a loosely coupled framework because concerning the MTV architecture it’s based upon. Django’s architecture means a variant of MVC architecture and also MTV is helpful because this completely separates the server code of the client’s machine.

**82Q. List the ways we add view functions to urls.py?**

**Ans:**

* Adding a function view
* Adding a class-based view

**83Q. Explain how can we build or set up the database in Django?**

**Ans:** we can make use of edit mysite/setting.py command, it is a simple Python module consists of levels for presenting or displaying Django settings.

By default Django uses SQLite; this also makes easy for Django users in case of any other type of installations. For example, if your database choice is different then you need to follow certain keys in the DATABASE like default items to match database connection settings.

* **Engines:** By these engines you change the database by using commands such as ‘django.db.backends.postgresql\_psycopg2’, ‘django.db.backends.sqlite3’, ‘django.db.backends.oracle’, ‘django.db.backends.mysql’, and so on.
* **Name:** This represents the name of your own database. If you are familiar with using SQLite as your database, in such case database is available as a file on your particular system. Moreover, the name should be as a fully absolute or exact path along with the file name of the particular file.
* Suppose if we are not using SQlite as your database then additional settings such password, user, the host must be added.

Django mainly uses SQLite as its default database to store entire information in a single file of the filesystem. If you want to use different database servers rather than SQLite, then make use of database administration tools to create a new database for the Django project.  Another way is by using your own database in that place, and remaining is to explain Django about how to use it. This is the place in which Python’s project settings.py file comes into the picture.

**We need to add the below code to the setting.py file:**

|  |  |
| --- | --- |
| 1  2  3  4  5  6 | DATABASE  = {  ‘Default’ : {  ‘ENGINE’ :  ‘django.db.backends.sqlite3’,  ‘NAME’ : os.path.join(BASE\_DIR, ‘db.sqlite3’),  }  } |

**84Q. List out the inheritance styles in Django?**

**Ans:** There are three possible inheritance styles in Django, and they are:

* **Proxy models:** This style is mainly used for those who want to modify the Python level behaviour of the model, without modifying the model’s fields.
* Abstract Base Classes: This inheritance style is used only when we want to make parent class hold the data which they don’t want to repeat it again in the child class.
* **Multi-table Inheritance:** This inheritance style is used only when we want to subclass an existing model and there must a database table designed for each model on its own.

**85Q. How to save an image locally using Python in which we already know the URL address?**

Ans: The following code is used to save the image locally from the URL address which we know.

|  |  |
| --- | --- |
| 1  2 | import urllib.request  urllib.request.urlretrieve("URL", "local-filename.jpg") |

**86Q. How can we access sessions in flask?**

**Ans:** A session will basically allow us to remember information from one request to another. In a flask, a signed cookie is used to make the user look at the session contents and modify them. Moreover, the user can change the session only when the secret key named Flask.secret\_key is present.

**87Q. Is flask an MVC model? If yes, justify your answer by showing an example of your application with the help of MVC pattern?**

**[Upcoming Batches - Python Training!](https://mindmajix.com/python-training" \l "training_dates" \t "_blank)**

**[15](https://mindmajix.com/python-training" \l "training_dates" \t "_blank)**

**[MARCH](https://mindmajix.com/python-training" \l "training_dates" \t "_blank)**

[Sunday  
7:00 AM IST](https://mindmajix.com/python-training" \l "training_dates" \t "_blank)

**[19](https://mindmajix.com/python-training" \l "training_dates" \t "_blank)**

**[MARCH](https://mindmajix.com/python-training" \l "training_dates" \t "_blank)**

[Thursday  
6:30 AM IST](https://mindmajix.com/python-training" \l "training_dates" \t "_blank)

**[28](https://mindmajix.com/python-training" \l "training_dates" \t "_blank)**

**[MARCH](https://mindmajix.com/python-training" \l "training_dates" \t "_blank)**

[Saturday  
6:30 AM IST](https://mindmajix.com/python-training" \l "training_dates" \t "_blank)

**[05](https://mindmajix.com/python-training" \l "training_dates" \t "_blank)**

**[APRIL](https://mindmajix.com/python-training" \l "training_dates" \t "_blank)**

[Sunday  
6:30 AM IST](https://mindmajix.com/python-training" \l "training_dates" \t "_blank)

[More Batches](https://mindmajix.com/python-training" \l "training_dates" \t "_blank)

**Ans:** Basically, flask is a minimalistic framework that behaves the same as MVC framework. So MVC will be perfectly suitable for the flask and we will consider the MVC pattern in the below example.

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7 | from flask import Flask  In this code your,  app = Flask(\_name\_)  @app.route(“/”)  Def hey():  return “Welcome to Appmajix”  app.run(debug = True) |

The following code can be fragmented into

Configuration part will be,

|  |  |
| --- | --- |
| 1  2 | In this code your,  app = Flask(\_name\_) |

View part will be,

|  |  |
| --- | --- |
| 1  2  3 | @app.route(“/”)  Def hey():  return “Welcome to Appmajix” |

While your main part will be,

|  |  |
| --- | --- |
| 1 | app.run(debug = True) |

**88Q. What are the database connections in Python Flask, explain?**

**Ans:** Database powered applications are supported by flask. The relational database systems need to create a schema that requires piping the schema.sql file into a SQLite3 command. So, in this case you need to install SQLite3 command on your system to initiate and create the database in the flask.

We can request a database using flask in three ways, and they are:

* **before\_request():** Using this we can request database before only without passing arguments.
* **after\_request():** This method is called after requesting the database and also send the response to client.
* **teardown\_request():** This method is called in the cases where the responses are not guaranteed and the exception is raised. They have no access to modify the request.

**89Q. Explain the procedure to minimize or lower the outages of Memcached server in your Python development?**

**Ans:** The following are the steps used to minimize the outages of the Memcached server in your Python development, and they are.

* When a single instance fails, this will impact on larger load of the database server. The client makes the request when the data is reloaded. In order to avoid this, the code that you have written must be used to lower cache stampedes then it will used to leave a minimal impact.
* The other way is to bring out the instance of the memcached on a new machine by using the IP address of the lost machine.
* Another important option is to lower the server outages is code. This code provides you the liberty to modify the memcached server list with minimal work
* Another way is by setting timeout value that will be one of the options for memcac

|  |  |
| --- | --- |
| 1  2  3  4  5 | Class Student:  def \_\_init\_\_(self, name):  self.name = name  S1=Student("XYZ")  print(S1.name) |

* hed clients to implement the memcached server outage. When the performance of the server goes down, the client keeps on sending a request until the timeout limit is reached.

**90Q. What is Dogpile effect?**

**Ans:** This is defined as an occurrence of event when the cache expires and also when the websites are hit with more number of requests by the client at a time. This dogpile effect can be averted by the use of a semaphore lock. If in the particular system the value expires then, first of all, the particular process receives the lock and begin generating new value.

**Python OOPS Concepts**

**91Q. What are the OOPS concepts available in Python?**

**Ans:** Python is also object-oriented programming language like other programming languages. It also contains different OOPS concepts, and they are

* Object
* Class
* Method
* Encapsulation
* Abstraction
* Inheritance
* Polymorphism

**92Q. Define object in Python?**

**Ans:** An object in Python is defined as an instance that has both state and behaviour. Everything in Python is made of objects.

**93Q. What is a class in Python?**

Ans: Class is defined as a logical entity that is a huge collection of objects and it also contains both methods and attributes.

**94Q. How to create a class in Python?**

**Ans:** In Python programming, the class is created using a class keyword. The syntax for creating a class is as follows:

|  |  |
| --- | --- |
| 1  2 | class ClassName:  #code (statement-suite) |

Example of creating a class in Python.

**Output:**

|  |  |
| --- | --- |
| 1 | XYZ |

**95Q. What is the syntax for creating an instance of a class in Python?**

**Ans:** The syntax for creating an instance of a class is as follows:

|  |  |
| --- | --- |
| 1 | <object-name> = <class-name>(<arguments>) |

**96Q. Define what is “Method” in Python programming?**

**Ans:** The Method is defined as the function associated with a particular object. The method which we define should not be unique as a class instance. Any type of objects can have methods.

**97Q. Does multiple inheritances is supported in Python?**

**Ans:** Multiple inheritance is supported in python. It is a process that provides flexibility to inherit multiple base classes in a child class.

Example of multiple inheritance in Python is as follows:

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13 | class Calculus:  def Sum(self,a,b):  return a+b;  class Calculus1:  def Mul(self,a,b):  return a\*b;  class Derived(Calculus,Calculus1):  def Div(self,a,b):  return a/b;  d = Derived()  print(d.Sum(10,30))  print(d.Mul(10,30))  print(d.Div(10,30)) |

**Output:**

|  |  |
| --- | --- |
| 1  2  3 | 40  300  0.3333 |

**98Q. What is data abstraction in Python?**

**Ans:** In simple words, abstraction can be defined as hiding of unnecessary data and showing or executing necessary data. In technical terms, abstraction can be defined as hiding internal process and showing only the functionality. In Python abstraction can be achieved using encapsulation.

**99Q. Define encapsulation in Python?**

**Ans:** Encapsulation is one of the most important aspects of object-oriented programming. Binding or wrapping of code and data together into a single cell is called encapsulation. Encapsulation in Python is mainly used to restrict access to methods and variables.

[**Python Training in Bangalore**](https://mindmajix.com/python-training-in-bangalore)

**100Q. What is polymorphism in Python?**

**Ans:** By using polymorphism in Python we will understand how to perform a single task in different ways. For example, designing a shape is the task and various possible ways in shapes are a triangle, rectangle, circle, and so on.

**101Q. Does Python make use of access specifiers?**

**Ans:** Python does not make use of access specifiers and also it does not provide a way to access an instance variable. Python introduced a concept of prefixing the name of the method, function, or variable by using a double or single underscore to act like the behaviour of private and protected access specifiers.

**102Q. How can we create an empty class in Python?**

**Ans:** Empty class in Python is defined as a class that does not contain any code defined within the block. It can be created using pass keyword and object to this class can be created outside the class itself.

**Example:**

|  |  |
| --- | --- |
| 1  2  3  4  5 | class x:  &nbsp; pass  obj=x()  obj.id="123"  print("Id = ",obj.id) |

**Output:**

|  |  |
| --- | --- |
| 1 | 123 |

**103Q. Define Constructor in Python?**

**Ans:** Constructor is a special type of method with a block of code to initialize the state of instance members of the class. A constructor is called only when the instance of the object is created. It is also used to verify that they are sufficient resources for objects to perform a specific task.

There are two types of constructors in Python, and they are:

* Parameterized constructor
* Non-parameterized constructor

**104Q. How can we create a constructor in Python programming?**

**Ans:** The \_init\_ method in Python stimulates the constructor of the class. Creating a constructor in Python can be explained clearly in the below example.

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12 | class Student:  def \_\_init\_\_(self,name,id):  self.id = id;  self.name = name;  def display (self):  print("ID: %d nName: %s"%(self.id,self.name))  stu1 =Student("nirvi",105)  stu2 = Student("tanvi",106)  #accessing display() method to print employee 1 information  stu1.display();  #accessing display() method to print employee 2 information  stu2.display(); |

**Output:**

|  |  |
| --- | --- |
| 1  2  3  4 | ID: 1  Name: nirvi  ID: 106  Name: Tanvi |

**105Q. Define Inheritance in Python?**

**Ans:** When an object of child class has the ability to acquire the properties of a parent class then it is called inheritance. It is mainly used to acquire runtime polymorphism and also it provides code reusability.