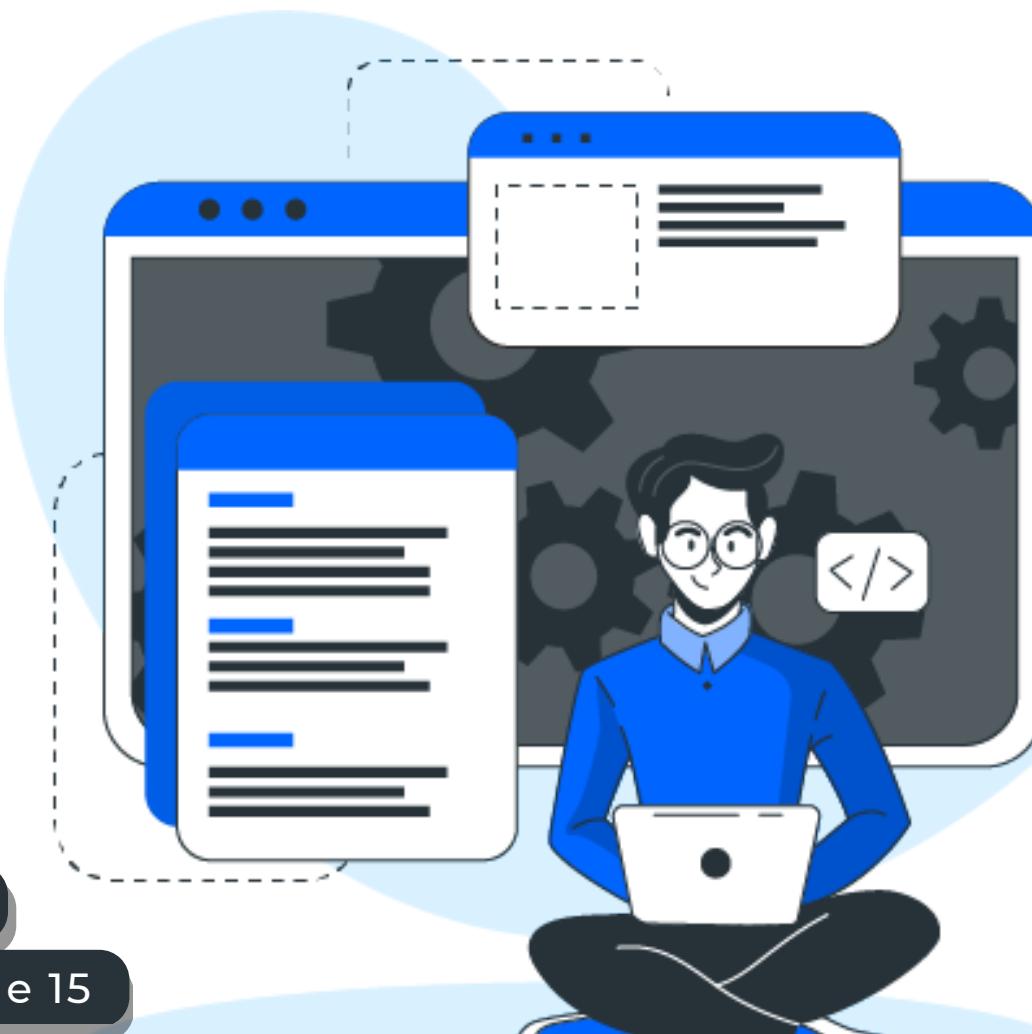




Most
Frequently
Asked

Python Interview Questions

w i t h a n s w e r s



Basic Level: Page 2

Advanced Level: Page 15



Join WhatsApp Group for Placement

Basic Level



1. What is the Difference Between a Shallow Copy and Deep Copy?

Answer:

Deepcopy creates a different object and populates it with the child objects of the original object. Therefore, changes in the original object are not reflected in the copy.

copy.deepcopy() creates a Deep Copy.

Shallow copy creates a different object and populates it with the references of the child objects within the original object. Therefore, changes in the original object are reflected in the copy.

copy.copy creates a Shallow Copy.

2. How Is Multithreading Achieved in Python?

Answer:

Multithreading usually implies that multiple threads are executed concurrently. The Python Global Interpreter Lock doesn't allow more than one thread to hold the Python interpreter at that particular point of time. So multithreading in python is achieved through context switching. It is quite different from multiprocessing which actually opens up multiple processes across multiple threads.

3. Discuss Django Architecture.

Answer:

Django is a web service used to build your web pages. Its architecture is as shown:

- Template: the front end of the web page
- Model: the back end where the data is stored
- View: It interacts with the model and template and maps it to the URL
- Django: serves the page to the user



Get free mentorship
from experts?



4. What Advantage Does the Numpy Array Have over a Nested List?

Answer:

Numpy is written in C so that all its complexities are backed into a simple to use a module. Lists, on the other hand, are dynamically typed. Therefore, Python must check the data type of each element every time it uses it. This makes Numpy arrays much faster than lists.

Numpy has a lot of additional functionality that list doesn't offer; for instance, a lot of things can be automated in Numpy.

5. What are Pickling and Unpickling?

Answer:

Pickling	Unpickling
Converting a Python object hierarchy to a byte stream is called pickling. Pickling is also referred to as serialization.	Converting a byte stream to a Python object hierarchy is called unpickling. Unpickling is also referred to as deserialization.

Free Pre - Placement Mock Test Series Aptitude & Technical

Why we are conducting Pre-Placement Test Series?

- To help students check their current level of Aptitude and Technical skills
- After knowing the current level it will become easy for a student to start their placement preparation

What does this Pre-Placement Test Series consist of?

- 25 Tests will be conducted on subjects C, C++, Java, Python, DSA, CN, OS, DBMS, Quant, Reasoning, and verbal ability.
- Topic-wise questions in every test will help students get strong and weak points





Follow us on Instagram



If you just created a neural network model, you can save that model to your hard drive, pickle it, and then unpickle to bring it back into another software program or to use it at a later time.

6. How is Memory managed in Python?

Answer:

Python has a private heap space that stores all the objects. The Python memory manager regulates various aspects of this heap, such as sharing, caching, segmentation, and allocation. The user has no control over the heap; only the Python interpreter has access.

7. Are Arguments in Python Passed by Value or by Reference?

Answer:

Arguments are passed in python by a reference. This means that any changes made within a function are reflected in the original object.

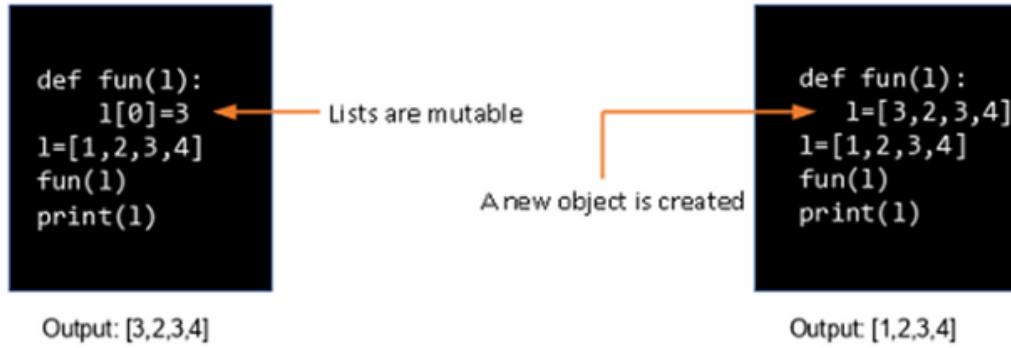
Consider two sets of code shown below:

Proud to be featured in more than **70 news articles**

The screenshot shows a news article from Business Standard. At the top, there's a navigation bar with a menu icon, a search bar, and links for 'SIGN IN' and 'SUBSCRIBE'. The main headline is 'Ed-tech platform Talent Battle crosses 3.5 Lakh registered students mark! The journey from 35 to 3.5 Lakh students!'. Below the headline, there's a timestamp 'February 13, 2023 19:00 IST | ANI Press Release' and a row of social media sharing icons. A large image of two men in blue shirts is displayed at the bottom.



Join WhatsApp Group for Placement



In the first example, we only assigned a value to one element of 'l', so the output is [3, 2, 3, 4].

In the second example, we have created a whole new object for 'l'. But, the values [3, 2, 3, 4] doesn't show up in the output as it is outside the definition of the function.

8. How Would You Generate Random Numbers in Python?

Answer:

To generate random numbers in Python, you must first import the random module.

The random() function generates a random float value between 0 & 1.

> random.random()

The randrange() function generates a random number within a given range.

Syntax: randrange(beginning, end, step)

Example - > random.randrange(1,10,2)

9. What Does the // Operator Do?

Answer:

In Python, the / operator performs division and returns the quotient in the float.

For example: 5 / 2 returns 2.5

The // operator, on the other hand, returns the quotient in integer.

For example: 5 // 2 returns 2

10. What Does the 'is' Operator Do?



Get free mentorship
from experts?

Answer:

The 'is' operator compares the id of the two objects.

```
list1=[1,2,3]
list2=[1,2,3]
list3=list1
list1 == list2  True
list1 is list2  False
list1 is list3  True
```

11. What Is the Purpose of the Pass Statement?

Answer:

The pass statement is used when there's a syntactic but not an operational requirement.

12. How Will You Check If All the Characters in a String are Alphanumeric?

Answer:

Python has an inbuilt method `isalnum()` which returns true if all characters in the string are alphanumeric.

Example -

```
>> "abcd123".isalnum()
Output: True
>>"abcd@123#".isalnum()
Output: False
Another way is to use regex as shown.
>>import re
>>bool(re.match('[A-Za-z0-9]+$', 'abcd123'))
Output: True
>> bool(re.match('[A-Za-z0-9]+$', 'abcd@123'))
Output: False
```

**Get a Free Mentorship from
experts for your Campus
Placement Preparation**

- Discuss your queries with experts
- Get a roadmap for your placement preparation



[Click to know more](#)



Follow us on Instagram



13. How Will You Merge Elements in a Sequence?

Answer:

There are three types of sequences in Python:

- Lists
- Tuples
- Strings

Example of Lists -

```
>>l1=[1,2,3]  
>>l2=[4,5,6]  
>>l1+l2
```

Output: [1,2,3,4,5,6]

Example of Tuples -

```
>>t1=(1,2,3)  
>>t2=(4,5,6)  
>>t1+t2
```

Output: (1,2,3,4,5,6)

Example of String -

```
>>s1="Talent"  
>>s2="Battle"  
>>s1+s2
```

Output: 'TalentBattle'

13. How Will You Merge Elements in a Sequence?

Answer:

There are three types of sequences in Python:

- Lists
- Tuples
- Strings

Example of Lists -

```
>>l1=[1,2,3]  
>>l2=[4,5,6]  
>>l1+l2
```

Output: [1,2,3,4,5,6]

Example of Tuples -

```
>>t1=(1,2,3)  
>>t2=(4,5,6)  
>>t1+t2
```

Output: (1,2,3,4,5,6)

Example of String -

```
>>s1="Talent"  
>>s2="Battle"  
>>s1+s2
```

Output: 'TalentBattle'



14. How Would You Remove All Leading Whitespace in a String?

Answer:

Python provides the inbuilt function lstrip() to remove all leading spaces from a string.

```
>>" Python".lstrip
```

Output: Python

15. How Would You Replace All Occurrences of a Substring with a New String?

Answer:

The replace() function can be used with strings for replacing a substring with a given string. Syntax:

```
str.replace(old, new, count)
```

replace() returns a new string without modifying the original string.

Example -

```
>>"Hey John. How are you, John?".replace("john","John",1)
```

Output: "Hey John. How are you, John?"

16. What Is the Difference Between Del and Remove() on Lists?

Answer:

The replace() function can be used with strings for replacing a substring with a given string. Syntax:

```
str.replace(old, new, count)
```

replace() returns a new string without modifying the original string.

Example -

```
>>"Hey John. How are you, John?".replace("john","John",1)
```

Output: "Hey John. How are you, John?"

del	remove()
<ul style="list-style-type: none">• del removes all elements of a list within a given range• Syntax: del list[start:end]	<ul style="list-style-type: none">• remove() removes the first occurrence of a particular character• Syntax: list.remove(element)



Get free mentorship
from experts?

Here is an example to understand the two statements -

```
>>lis=['a', 'b', 'c', 'd']
```

```
>>del lis[1:3]
```

```
>>lis
```

Output: ["a", "d"]

```
>>lis=['a', 'b', 'b', 'd']
```

```
>>lis.remove('b')
```

```
>>lis
```

Output: ['a', 'b', 'd']

Note that in the range 1:3, the elements are counted up to 2 and not 3.

17. How Do You Display the Contents of a Text File in Reverse Order?

Answer:

You can display the contents of a text file in reverse order using the following steps:

- Open the file using the open() function
- Store the contents of the file into a list
- Reverse the contents of the list
- Run a for loop to iterate through the list

Complete Masterclass Courses and Features

- | | | |
|---|---|--|
| <ul style="list-style-type: none">• Aptitude Cracker Course• C Programming• C++• Java• Python• Data Structures and Algorithms• Operating Systems• Computer Networks• DBMS• Topic-wise Mock Tests | <ul style="list-style-type: none">• Company Wise Mock Tests• Technical & Personal Interview Preparation• One to One Mock Interviews• Full Stack Development• Artificial Intelligence• Machine Learning• Data Analytics• Data Science• PowerBI• Tableau | <ul style="list-style-type: none">• Real-Time projects on AI, ML, & Data Science• Mini Projects based on C, C++, Java, Python• TCS iON Remote Internship (For 2 years course)• Certifications by Talent Battle and TCS iON• LIVE Lectures + Recorded Courses |
|---|---|--|

Complete Masterclass Courses and Features

TCS NQT | Accenture | Capgemini | Cognizant | Infosys | Wipro | Tech Mahindra | LTI | DXC

Hexaware | Persistent | Deloitte | Mindtree | Virtusa | Goldman Sachs | Bosch | Samsung Amazon | Nalsoft | Zoho Cisco and 10+ more companies preparation.



18. Differentiate Between `append()` and `extend()`.

Answer:

append()	extend()
<ul style="list-style-type: none">• <code>append()</code> adds an element to the end of the list• Example - <code>>>lst=[1,2,3]</code> <code>>>lst.append(4)</code> <code>>>lst</code> Output:[1,2,3,4]	<ul style="list-style-type: none">• <code>extend()</code> adds elements from an iterable to the end of the list• Example - <code>>>lst=[1,2,3]</code> <code>>>lst.extend([4,5,6])</code> <code>>>lst</code> Output:[1,2,3,4,5,6]

19. What Is the Output of the below Code? Justify Your

Answer:

```
>>def addToList(val, list=[]):  
>> list.append(val)  
>> return list  
>>list1 = addToList(1)  
>>list2 = addToList(123,[])  
>>list3 = addToList('a')  
>>print ("list1 = %s" % list1)  
>>print ("list2 = %s" % list2)  
>>print ("list3 = %s" % list3)
```

Output:

```
list1 = [1,'a']  
list2 = [123]  
list3 = [1,'a']
```



Join WhatsApp Group for Placement

Note that list1 and list3 are equal. When we passed the information to the addToList, we did it without a second value. If we don't have an empty list as the second value, it will start off with an empty list, which we then append. For list2, we appended the value to an empty list, so its value becomes [123].

For list3, we're adding 'a' to the list. Because we didn't designate the list, it is a shared value. It means the list doesn't reset and we get its value as [1, 'a'].

Remember that a default list is created only once during the function and not during its call number.

20. What Is the Difference Between a List and a Tuple?

Answer:

Lists are mutable while tuples are immutable.

Example:

List

```
>>lst = [1,2,3]
>>lst[2] = 4
>>lst
```

Output:[1,2,4]

Tuple

```
>>tpl = (1,2,3)
>>tpl[2] = 4
>>tpl
```

Output:TypeError: 'tuple' object does not support item assignment

There is an error because you can't change the tuple 1 2 3 into 1 2 4. You have to completely reassign tuple to a new value.

21. What Is Docstring in Python?

Answer:

Docstrings are used in providing documentation to various Python modules, classes, functions, and methods.

Example -

```
def add(a,b):
    """This function adds two numbers."""
    sum=a+b
    return sum
sum=add(10,20)
print("Accessing doctstring method 1:",add.__doc__)
print("Accessing doctstring method 2:",end="")
help(add)
```



Get free mentorship
from experts?



Output -

Accessing docstring method 1: This function adds two numbers.

Accessing docstring method 2: Help on function add-in module __main__:
add(a, b)

This function adds two numbers.

22. How Do You Use Print() Without the Newline?

Answer:

The solution to this depends on the Python version you are using.

Python v2

```
>>print("Hi. "),  
>>print("How are you?")
```

Output: Hi. How are you?

Python v3

```
>>print("Hi",end=" ")  
>>print("How are you?")
```

Output: Hi. How are you?

Some of our placed students from Complete Masterclass

Shrinija Kalluri
Placed in Oracle
9 LPA

Rohit Borse
Placed in Accenture
6.5 LPA



25. Write a Function Prototype That Takes a Variable Number of Arguments

Answer:

The function prototype is as follows:

```
def function_name(*list)
>>def fun(*var):
>> for i in var:
print(i)
>>fun(1)
>>fun(1,25,6)
```

In the above code, * indicates that there are multiple arguments of a variable.

26. What Are *args and *kwargs?

Answer:

- args

It is used in a function prototype to accept a varying number of arguments.

It's an iterable object.

Usage - def fun(*args)

- kwargs

It is used in a function prototype to accept the varying number of keyworded arguments.

· It's an iterable object

· Usage - def fun(**kwargs):
fun(colour="red".units=2)

27. “in Python, Functions Are First-class Objects.” What Do You Infer from This?

Answer:

It means that a function can be treated just like an object. You can assign them to variables, or pass them as arguments to other functions. You can even return them from other functions.



28. What Is the Output Of: Print(__name__)? Justify Your Answer.

`__name__` is a special variable that holds the name of the current module. Program execution starts from main or code with 0 indentations. Thus, `__name__` has a value `__main__` in the above case. If the file is imported from another module, `__name__` holds the name of this module.

29. What Is a Numpy Array?

Answer:

A numpy array is a grid of values, all of the same type, and is indexed by a tuple of non-negative integers. The number of dimensions determines the rank of the array. The shape of an array is a tuple of integers giving the size of the array along each dimension.

30. What Is the Difference Between Matrices and Arrays?

Answer:

Matrices	Arrays
<ul style="list-style-type: none">• A matrix comes from linear algebra and is a two-dimensional representation of data• It comes with a powerful set of mathematical operations that allow you to manipulate the data in interesting ways	<ul style="list-style-type: none">• An array is a sequence of objects of similar data type• An array within another array forms a matrix

Advanced Level



31. How Do You Get Indices of N Maximum Values in a Numpy Array?

Answer:

```
>>import numpy as np  
>>arr=np.array([1, 3, 2, 4, 5])  
>>print(arr.argsort() [ -N: ][: :-1])
```

32. How Would You Obtain the Res_set from the Train_set and the Test_set from Below?

```
>>train_set=np.array([1, 2, 3])  
>>test_set=np.array([[0, 1, 2], [1, 2, 3])  
Res_set [[1, 2, 3], [0, 1, 2], [1, 2, 3]]
```

Choose the correct option:

- 1.res_set = train_set.append(test_set)
- 2.res_set = np.concatenate([train_set, test_set]))
- 3.resulting_set = np.vstack([train_set, test_set])
- 4.None of these



Get free mentorship
from experts?

Talent Battle Masterclass - Placement Reports

Highest CTC

40 LPA

85% students
placed with CTC more
than 6 LPA

Average CTC

8.5 LPA

Referral Opportunities
in Top Companies & Startups
by Talent Battle

Average Offers

per student: 2.7

Students from
5000+ colleges
use Masterclass for
Placement Preparation

**Answer:**

Here, options a and b would both do horizontal stacking, but we want vertical stacking. So, option c is the right statement.

```
resulting_set = np.vstack([train_set, test_set])
```

33. How Would You Import a Decision Tree Classifier in Sklearn?**Choose the Correct Option.**

- 1.from sklearn.decision_tree import DecisionTreeClassifier
- 2.from sklearn.ensemble import DecisionTreeClassifier
- 3.from sklearn.tree import DecisionTreeClassifier
- 4.None of these

Answer:

```
3. from sklearn.tree import DecisionTreeClassifier
```

34. You Have Uploaded the Dataset in Csv Format on Google Spreadsheet and Shared It Publicly. How Can You Access This in Python?**Answer:**

We can use the following code:

```
>>link = https://docs.google.com/spreadsheets/d/...
>>source = StringIO.StringIO(requests.get(link).content)
>>data = pd.read_csv(source)
```

35. What Is the Difference Between the Two Data Series given Below?

df['Name'] and df.loc[:, 'Name'], where:

```
df = pd.DataFrame(['aa', 'bb', 'xx', 'uu'], [21, 16, 50, 33], columns = ['Name', 'Age'])
```

Choose the correct option:

- 1.1 is the view of original dataframe and 2 is a copy of original dataframe
- 2.2 is the view of original dataframe and 1 is a copy of original dataframe
- 3.Both are copies of original dataframe
- 4.Both are views of original dataframe

Answer:

3. Both are copies of the original dataframe.



Join WhatsApp Group for Placement

36. You Get the Error “temp.Csv” While Trying to Read a File Using Pandas. Which of the Following Could Correct It?

Error:

Traceback (most recent call last): File "<input>", line 1,
in<module> UnicodeEncodeError:
'ascii' codec can't encode character.

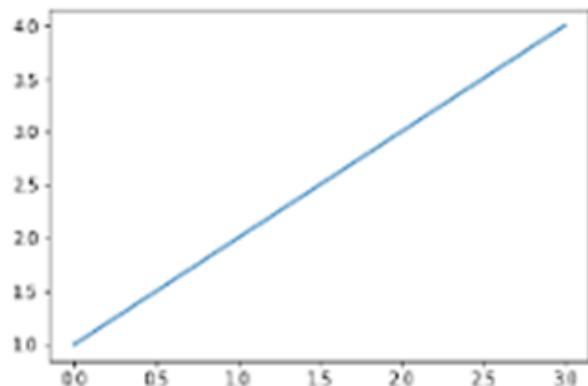
Choose the correct option:

- 1.pd.read_csv("temp.csv", compression='gzip')
- 2.pd.read_csv("temp.csv", dialect='str')
- 3.pd.read_csv("temp.csv", encoding='utf-8')
- 4.None of these

Answer:

The error relates to the difference between utf-8 coding and a Unicode. So option 3. pd.read_csv("temp.csv", encoding='utf-8') can correct it.

37. How Do You Set a Line Width in the Plot given Below?



```
>>import matplotlib.pyplot as plt  
>>plt.plot([1,2,3,4])  
>>plt.show()
```



Get free mentorship
from experts?



Choose the correct option:

- 1.pd.read_csv("temp.csv", compression='gzip')
- 2.pd.read_csv("temp.csv", dialect='str')
- 3.pd.read_csv("temp.csv", encoding='utf-8')
- 4.None of these

Answer:

3. In line two, write plt.plot([1,2,3,4], lw=3)

38. How Would You Reset the Index of a Dataframe to a given List?

Choose the Correct Option

- 1.df.reset_index(new_index,)
- 2.df.reindex(new_index,)
- 3.df.reindex_like(new_index,)
- 4.None of these

Answer:

3. df.reindex_like(new_index,)

39. How Can You Copy Objects in Python?

Answer:

The function used to copy objects in Python are:

copy.copy for shallow copy and

copy.deepcopy() for deep copy

Complete Placement Preparation LIVE Masterclass

Aptitude | Coding | Certifications & Upskilling | Mock Interviews & Interview Preparation

Click to know more



Page no: 18



40. What Is the Difference Between range() and xrange() Functions in Python?

Answer:

range()	xrange()
<ul style="list-style-type: none">• range returns a Python list object	<ul style="list-style-type: none">• xrange returns an xrange object

41. How Can You Check Whether a Pandas Dataframe Is Empty or Not?

Answer:

The attribute df.empty is used to check whether a pandas data frame is empty or not.

```
>>import pandas as pd  
>>df=pd.DataFrame({A:[]})  
>>df.empty  
Output: True
```

42. If You Split Your Data into Train/Test Splits, Is It Possible to over Fit Your Model?

Answer:

Yes. One common beginner mistake is re-tuning a model or training new models with different parameters after seeing its performance on the test set.

43. Which Python Library Is Built on Top of Matplotlib and Pandas to Ease Data Plotting?

Answer:

Seaborn is a Python library built on top of matplotlib and pandas to ease data plotting. It is a data visualization library in Python that provides a high-level interface for drawing statistical informative graphs.

44. What are the important features of Python?

Answer:

- Python is a scripting language. Python, unlike other programming languages like C and its derivatives, does not require compilation prior to execution.
- Python is dynamically typed, which means you don't have to specify the kinds of variables when declaring them or anything.
- Python is well suited to object-oriented programming since it supports class definition, composition, and inheritance.

45. What type of language is Python?

Answer:

Although Python can be used to write scripts, it is primarily used as a general-purpose programming language.

46. Explain how Python is an interpreted language.

Answer:

Any programming language that is not in machine-level code before runtime is called an interpreted language. Python is thus an interpreted language.

**Talent Battle is
associated with TCS iON**
for content partnership &
providing internships to
students across India.





Get free mentorship
from experts?



47. What is PEP 8?

Answer:

PEP denotes Python Enhancement Proposal. It's a collection of guidelines for formatting Python code for maximum readability.

48. Explain Python namespace.

Answer:

In Python, a namespace refers to the name that is assigned to each object.

49. What are decorators in Python?

Answer:

Decorators are used for changing the appearance of a function without changing its structure. Decorators are typically defined prior to the function they are enhancing.

50. How to use decorators in Python?

Answer:

Decorators are typically defined prior to the function they are enhancing. To use a decorator, we must first specify its function. Then we write the function to which it is applied, simply placing the decorator function above the function to which it must be applied.

51. Differentiate between .pyc and .py.

Answer:

The .py files are the source code files for Python. The bytecode of the python files are stored in .pyc files, which are created when code is imported from another source. The interpreter saves time by converting the source .py files to .pyc files.

52. What is slicing in Python?

Answer:

Slicing is a technique for gaining access to specific bits of sequences such as strings, tuples, and lists.



53. How to use the slicing operator in Python?

Answer:

Slicing is a technique for gaining access to specific bits of sequences such as lists, tuples, and strings. The slicing syntax is [start:end:step]. This step can also be skipped. [start:end] returns all sequence items from the start (inclusive) to the end-1 element. It means the ith element from the end of the start or end element is negative i. The step represents the jump or the number of components that must be skipped.

54. What are keywords in python?

Answer:

In Python, keywords are reserved words with a specific meaning. They are commonly used to specify the type of variables. Variable and function names cannot contain keywords. Following are the 33 keywords of Python:

- Yield
- For
- Else
- Elif
- If
- Not
- Or
- And
- Raise
- .Nonlocal
- None
- Is
- In
- Import
- Global
- From
- Finally
- Except
- Del
- Continue
- Class
- Assert
- With
- Try
- False
- True
- Return
- Pass
- Lambda
- Def
- As
- Break
- While



Join WhatsApp Group for Placement



55. How to combine dataframes in Pandas?

Answer:

The following are the ways through which the data frames in Pandas can be combined:

- Concatenating them by vertically stacking the two dataframes.
- Concatenating them by horizontally stacking the two dataframes.
- Putting them together in a single column.

56. What are the key features of the Python 3.9.0.0 version?

Answer:

- Zoneinfo and graphlib are two new modules.
- Improved modules such as asyncio and ast.
- Optimizations include improved idiom for assignment, signal handling, and Python built-ins.
- Removal of erroneous methods and functions.
- Instead of LL1, a new parser is based on PEG.
- Remove Prefixes and Suffixes with New String Methods.
- Generics with type hinting in standard collections.

Free Pre - Placement Mock Test Series - Aptitude & Technical



Why we are conducting Pre-Placement Test Series?

- To help students check their current level of Aptitude and Technical skills
- After knowing the current level it will become easy for a student to start their placement preparation

What does this Pre-Placement Test Series consist of?

- 25 Tests will be conducted on subjects C, C++, Java, Python, DSA, CN, OS, DBMS, Quant, Reasoning, and verbal ability.
- Topic-wise questions in every test will help students get strong and weak points



Get free mentorship
from experts?



57. In Python, how is memory managed?

Answer:

- Python's private heap space is in charge of memory management. A private heap holds all Python objects and data structures. This secret heap is not accessible to the programmer. Instead, the Python interpreter takes care of it.
- Python also includes a built-in garbage collector, which recycles all unused memory and makes it available to the heap space.
- Python's memory management is in charge of allocating heap space for Python objects. The core API allows programmers access to some programming tools.

58. Explain PYTHONPATH.

Answer:

It's an environment variable that is used when you import a module. When a module is imported, PYTHONPATH is checked to see if the imported modules are present in various folders. It is used by the interpreter to determine which module to load.

59. Explain global variables and local variables in Python.

Answer:

Local Variables:

A local variable is any variable declared within a function. This variable exists only in local space, not in global space.

Global Variables:

Global variables are variables declared outside of a function or in a global space. Any function in the program can access these variables.

60. Is Python case sensitive?

Answer:

Yes, Python is case sensitive.

61. How to install Python on Windows and set path variables?

Answer:

- Download Python from <https://www.python.org/downloads/>
- Install it on your computer. Using your command prompt, look for the location where PYTHON is installed on your computer by typing cmd python.
- Then, in advanced system settings, create a new variable called PYTHON_NAME and paste the copied path into it.
- Search the path variable, choose its value and select 'edit'.
- If the value doesn't have a semicolon at the end, add one, and then type %PYTHON HOME%.

62. Is it necessary to indent in Python?

Answer:

Indentation is required in Python. It designates a coding block. An indented block contains all of the code for loops, classes, functions, and so on. Typically, four space characters are used. Your code will not execute correctly if it is not indented, and it will also generate errors.

63. On Unix, how do you make a Python script executable?

Answer:

Script file should start with #!/usr/bin/env python.

64. What is the use of self in Python?

Answer:

Self is used to represent the class instance. In Python, you can access the class's attributes and methods with this keyword. It connects the attributes to the arguments. Self appears in a variety of contexts and is frequently mistaken for a term. Self is not a keyword in Python, unlike in C++.

65. What are the literals in Python?

Answer:

For primitive data types, a literal in Python source code indicates a fixed value.



Follow us on Instagram

66. What are the types of literals in Python?

Answer:

For primitive data types, a literal in Python source code indicates a fixed value. Following are the 5 types of literal in Python:

- String Literal: A string literal is formed by assigning some text to a variable that is contained in single or double-quotes. Assign the multiline text encased in triple quotes to produce multiline literals.
- Numeric Literal: They may contain numeric values that are floating-point values, integers, or complex numbers.
- Character Literal: It is made by putting a single character in double-quotes.
- Boolean Literal: True or False
- Literal Collections: There are four types of literals such as list collections, tuple literals, set literals, dictionary literals, and set literals.

67. What are Python modules? Name a few Python built-in modules that are often used.

Answer:

Python modules are files that contain Python code. Functions, classes, or variables can be used in this code. A Python module is a .py file that contains code that may be executed. The following are the commonly used built-in modules:

- JSON
- data time
- random
- math
- sys
- OS

68. What is `_init_`?

Answer:

`_init_` is a constructor or method in Python. This method is used to allocate memory when a new object is created.

**Get a Free Mentorship from
experts for your Campus
Placement Preparation**

- Discuss your queries with experts
- Get a roadmap for your placement preparation



Click to know more

69. What is the Lambda function?



Get free mentorship
from experts?

Answer:

A lambda function is a type of anonymous function. This function can take as many parameters as you want, but just one statement.

70. Why Lambda is used in Python?

Answer:

Lambda is typically utilized in instances where an anonymous function is required for a short period of time. Lambda functions can be applied in two different ways:

- Assigning Lambda functions to a variable
- Wrapping Lambda function into another function

71. How does continue, break, and pass work?

Answer:

Continue	When a specified condition is met, the control is moved to the beginning of the loop, allowing some parts of the loop to be transferred.
Break	When a condition is met, the loop is terminated and control is passed to the next statement.
Pass	When you need a piece of code syntactically but don't want to execute it, use this. This is a null operation.

72. What are Python iterators?

Answer:

Iterators are things that can be iterated through or traversed.



Follow us on Instagram



73. Differentiate between range and xrange.

Answer:

In terms of functionality, xrange and range are essentially the same. They both provide you the option of generating a list of integers to use whatever you want. The sole difference between range and xrange is that range produces a Python list object whereas xrange returns an xrange object. This is especially true if you are working with a machine that requires a lot of memory, such as a phone because range will utilize as much memory as it can to generate your array of numbers, which can cause a memory error and crash your program. It is a beast with a memory problem.

74. What are unpickling and pickling?

Answer:

The Pickle module takes any Python object and converts it to a string representation, which it then dumps into a file using the dump method. This is known as pickling. Unpickling is the process of recovering original Python objects from a stored text representation.

75. What are generators in Python?

Answer:

Functions which return an iterable set of items are known as generators.

76. How do you copy an object in Python?

Answer:

The assignment statement (= operator) in Python does not copy objects. Instead, it establishes a connection between the existing object and the name of the target variable. The copy module is used to make copies of an object in Python. Furthermore, the copy module provides two options for producing copies of a given object –

Deep Copy: Deep Copy recursively replicates all values from source to destination object, including the objects referenced by the source object.
from copy import copy, deepcopy



Join WhatsApp Group for Placement



```
from copy import copy, deepcopy
list_1 = [1, 2, [3, 5], 4]
## shallow copy
list_2 = copy(list_1)
list_2[3] = 7
list_2[2].append(6)
list_2 # output => [1, 2, [3, 5, 6], 7]
list_1 # output => [1, 2, [3, 5, 6], 4]
## deep copy
list_3 = deepcopy(list_1)
list_3[3] = 8
list_3[2].append(7)
list_3 # output => [1, 2, [3, 5, 6, 7], 8]
list_1 # output => [1, 2, [3, 5, 6], 4]
```

Shallow Copy: A bit-wise copy of an object is called a shallow copy. The values in the copied object are identical to those in the original object. If one of the values is a reference to another object, only its reference addresses are copied.

77. In Python, are arguments provided by value or reference?

Answer:

Pass by value: The actual item's copy is passed. Changing the value of the object's copy has no effect on the original object's value.

Complete Masterclass Courses and Features

- | | | |
|---|---|--|
| <ul style="list-style-type: none">Aptitude Cracker CourseC ProgrammingC++JavaPythonData Structures and AlgorithmsOperating SystemsComputer NetworksDBMSTopic-wise Mock Tests | <ul style="list-style-type: none">Company Wise Mock TestsTechnical & Personal Interview PreparationOne to One Mock InterviewsFull Stack DevelopmentArtificial IntelligenceMachine LearningData AnalyticsData SciencePowerBITableau | <ul style="list-style-type: none">Real-Time projects on AI, ML, & Data ScienceMini Projects based on C, C++, Java, PythonTCS iON Remote Internship (For 2 years course)Certifications by Talent Battle and TCS iONLIVE Lectures + Recorded Courses |
|---|---|--|

Complete Masterclass Courses and Features

TCS NQT | Accenture | Capgemini | Cognizant | Infosys | Wipro | Tech Mahindra | LTI | DXC
Hexaware | Persistent | Deloitte | Mindtree | Virtusa | Goldman Sachs | Bosch | Samsung Amazon |
Nalsoft | Zoho Cisco and 10+ more companies preparation.

Pass by reference: The actual object is passed as a reference. The value of the old object will change if the value of the new object is changed.

Arguments are passed by reference in Python.

```
def appendNumber(arr):
    arr.append(4)
arr = [1, 2, 3]
print(arr) #Output: => [1, 2, 3]
appendNumber(arr)
print(arr) #Output: => [1, 2, 3, 4]
```

78. How to delete a file in Python?

Answer:

Use command `os.remove(file_name)` to delete a file in Python.

79. Explain join() and split() functions in Python.

Answer:

The `join()` function can be used to combine a list of strings based on a delimiter into a single string.

The `split()` function can be used to split a string into a list of strings based on a delimiter.

```
string = "This is a string."
string_list = string.split(' ') #delimiter is 'space' character or ''
print(string_list) #output: ['This', 'is', 'a', 'string.']
print(''.join(string_list)) #output: This is a string.
```

80. What are negative indexes and why are they used?

Answer:

·The indexes from the end of the list, tuple, or string are called negative indexes.

·`Arr[-1]` denotes the array's last element. `Arr[]`

81. How will you capitalize the first letter of string?

Answer:

The `capitalize()` function in Python capitalizes a string's initial letter. It returns the original text if the string already contains a capital letter at the beginning.

82. What method will you use to convert a string to all lowercase?

Answer:

The lower() function can be used to convert a string to lowercase.

83. In Python, how do you remark numerous lines?

Answer:

Comments that involve multiple lines are known as multi-line comments. A # must prefix all lines that will be commented. You can also use a convenient shortcut to remark several lines. All you have to do is hold down the ctrl key and left-click anywhere you want a # character to appear, then input a # once. This will add a comment to every line where you put your cursor.

84. What is the purpose of 'not', 'is', and 'in' operators?

Answer:

Special functions are known as operators. They take one or more input values and output a result.

not- returns the boolean value's inverse

is- returns true when both operands are true

in- determines whether a certain element is present in a series

85. What are the functions help() and dir() used for in Python?

Answer:

Both help() and dir() are available from the Python interpreter and are used to provide a condensed list of built-in functions.

dir() function: The defined symbols are displayed using the dir() function.

help() function: The help() function displays the documentation string and also allows you to access help for modules, keywords, attributes, and other items.

86. Why isn't all the memory de-allocated when Python exits?

Answer:

- When Python quits, some Python modules, especially those with circular references to other objects or objects referenced from global namespaces, are not necessarily freed or deallocated.
- Python would try to de-allocate/destroy all other objects on exit because it has its own efficient cleanup mechanism.
- It is difficult to de-allocate memory that has been reserved by the C library.



Follow us on Instagram



Join WhatsApp Group for Placement

87. What is a dictionary in Python?

Answer:

Dictionary is one of Python's built-in datatypes. It establishes a one-to-one correspondence between keys and values. Dictionary keys and values are stored in pairs in dictionaries. Keys are used to index dictionaries.

88. In Python, how do you utilize ternary operators?

Answer:

The Ternary operator is the operator for displaying conditional statements. This is made of true or false values and a statement that must be evaluated.

89. Explain the split(), sub(), and subn() methods of the Python "re" module.

Answer:

Python's "re" module provides three ways for modifying strings. They are:

split (): a regex pattern is used to "separate" a string into a list

subn(): It works similarly to sub(), returning the new string as well as the number of replacements.

sub(): identifies all substrings that match the regex pattern and replaces them with a new string

90. Explain Python packages.

Answer:

Packages in Python are namespaces that contain numerous modules.

91. What are built-in types of Python?

Answer:

Given below are the built-in types of Python:

- Built in functions
- Boolean
- String
- Complex numbers
- Floating point
- Integers

Complete Placement Preparation LIVE Masterclass

Aptitude | Coding | Certifications & Upskilling | Mock Interviews & Interview Preparation

Click to know more

Page no: 32

92. What are the benefits of NumPy arrays over (nested) Python lists?

Answer:

- Lists in Python are useful general-purpose containers. They allow for (relatively) quick insertion, deletion, appending, and concatenation, and Python's list comprehensions make them simple to create and operate.
- They have some limitations: they don't enable "vectorized" operations like elementwise addition and multiplication, and because they can include objects of different types, Python must maintain type information for each element and execute type dispatching code while working on it.
- NumPy arrays are faster, and NumPy comes with a number of features, including histograms, algebra, linear, basic statistics, fast searching, convolutions, FFTs, and more.

93. What is the best way to add values to a Python array?

Answer:

The append(), extend(), and insert (i,x) procedures can be used to add elements to an array.

94. What is the best way to remove values from a Python array?

Answer:

The pop() and remove() methods can be used to remove elements from an array. The difference between these two functions is that one returns the removed value while the other does not.

95. Is there an object-oriented Programming (OOps) concept in Python?

Answer:

Python is a computer language that focuses on objects. This indicates that by simply constructing an object model, every program can be solved in Python. Python, on the other hand, may be used as both a procedural and structured language.





Follow us on Instagram



96. How multithreading is achieved in Python?

Answer:

- Although Python includes a multi-threading module, it is usually not a good idea to utilize it if you want to multi-thread to speed up your code.
- As this happens so quickly, it may appear to the human eye that your threads are running in parallel, but they are actually sharing the same CPU core.
- The Global Interpreter Lock is a Python concept (GIL). Only one of your 'threads' can execute at a moment, thanks to the GIL. A thread obtains the GIL, performs some work, and then passes the GIL to the following thread.

97. How are classes created in Python?

Answer:

The class keyword in Python is used to construct a class.

98. What is pandas dataframe?

Answer:

A dataframe is a 2D changeable and tabular structure for representing data with rows and columns labelled.

99. Explain monkey patching in Python.

Answer:

Monkey patches are solely used in Python to run-time dynamic updates to a class or module.

100. What is inheritance in Python?

Answer:

Inheritance allows one class to gain all of another class's members (for example, attributes and methods). Inheritance allows for code reuse, making it easier to develop and maintain applications.

101. What are the different types of inheritance in Python?

Answer:

The following are the various types of inheritance in Python:



Join WhatsApp Group for Placement

- Single inheritance: The members of a single super class are acquired by a derived class.
- Multiple inheritance: More than one base class is inherited by a derived class.
- Multi-level inheritance: D1 is a derived class inherited from base1 while D2 is inherited from base2.
- Hierarchical Inheritance: You can inherit any number of child classes from a single base class.

102. Is multiple inheritance possible in Python?

Answer:

A class can be inherited from multiple parent classes, which is known as multiple inheritance. In contrast to Java, Python allows multiple inheritance.

103. Explain polymorphism in Python.

Answer:

The ability to take various forms is known as polymorphism. For example, if the parent class has a method named ABC, the child class can likewise have a method named ABC with its own parameters and variables. Python makes polymorphism possible.

Some of our placed students from Complete Masterclass



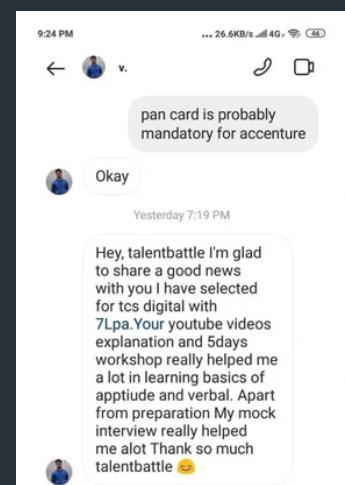
Placed in Cognizant GenC

10 LPA



Placed in BMC Software

12.50 LPA



Placed in TCS Digital

7 LPA

104. What is encapsulation in Python?

Answer:

Encapsulation refers to the joining of code and data. Encapsulation is demonstrated through a Python class.

105. In Python, how do you abstract data?

Answer:

Only the necessary details are provided, while the implementation is hidden from view. Interfaces and abstract classes can be used to do this in Python.

106. Is access specifiers used in Python?

Answer:

Access to an instance variable or function is not limited in Python. To imitate the behavior of protected and private access specifiers, Python introduces the idea of prefixing the name of the variable, function, or method with a single or double underscore.

107. How to create an empty class in Python?

Answer:

A class that has no code defined within its block is called an empty class. The pass keyword can be used to generate it. You can, however, create objects of this class outside of the class. When used in Python, the PASS command has no effect.

108. What does an object() do?

Answer:

It produces a featureless object that serves as the foundation for all classes. It also does not accept any parameters.

109. What is Flask and explain its benefits.

Answer:

Flask is a Python web microframework based on the BSD license. Two of its dependencies are Werkzeug and Jinja2. This means it will have few, if any, external library dependencies. It lightens the framework while reducing update dependencies and security vulnerabilities.

A session is just a way of remembering information from one request to the next. A session in a flask employs a signed cookie to allow the user to inspect and edit the contents of the session. If the user only has the secret key, he or she can change the session. Flask.secret key.

110. Is Django better as compared to Flask?

Answer:

Django and Flask map URLs or addresses entered into web browsers into Python functions.

Flask is easier to use than Django, but it doesn't do much for you, so you will have to specify the specifics, whereas Django does a lot for you and you won't have to do anything. Django has prewritten code that the user must examine, whereas Flask allows users to write their own code, making it easier to grasp. Both are technically excellent and have their own set of advantages and disadvantages.

111. Differentiate between Pyramid, Django, and Flask.

Answer:

- Pyramid is designed for larger apps. It gives developers flexibility and allows them to utilize the appropriate tools for their projects. The database, URL structure, templating style, and other options are all available to the developer. Pyramid can be easily customized.
- Flask is a "microframework" designed for small applications with straightforward needs. External libraries are required in a flask. The flask is now ready for use.
- Django, like Pyramid, may be used for larger applications. It has an ORM in it.

112. What is PIP?

Answer:

PIP denotes Python Installer Package. It is used to install various Python modules. It's a command-line utility that creates a unified interface for installing various Python modules. It searches the internet for the package and installs it into the working directory without requiring any user intervention.



Follow us on Instagram

113. What is the use of sessions in the Django framework?

Answer:

Django has a session feature that allows you to store and retrieve data for each site visitor. Django isolates the process of sending and receiving cookies by keeping all necessary data on the server-side and inserting a session ID cookie on the client-side.

114. Write a program that checks if all of the numbers in a sequence are unique.

Answer:

```
def check_distinct(data_list):
    if len(data_list) == len(set(data_list)):
        return True
    else:
        return False;
print(check_distinct([1,6,5,8])) #Prints True
print(check_distinct([2,2,5,5,7,8])) #Prints False
```



Join WhatsApp Group for Placement

Talent Battle Masterclass - Placement Reports

Highest CTC

40 LPA

85% students placed with CTC more than 6 LPA

Average CTC

8.5 LPA

Referral Opportunities
in Top Companies & Startups
by Talent Battle

Average Offers per student: 2.7

Students from
5000+ colleges
use Masterclass for
Placement Preparation



Get free mentorship
from experts?



115. What is an operator in Python?

Answer:

An operator is a symbol that is applied to a set of values to produce a result. An operator manipulates operands. Numeric literals or variables that hold values are known as operands. Unary, binary, and ternary operators are all possible.

The unary operator, which requires only one operand, the binary operator, which requires two operands, and the ternary operator, which requires three operands.

116. What are the various types of operators in Python?

Answer:

- Bitwise operators
- Identity operators
- Membership operators
- Logical operators
- Assignment operators
- Relational operators
- Arithmetic operators

Complete Placement Preparation LIVE Masterclass

Aptitude | Coding | Certifications & Upskilling | Mock Interviews & Interview Preparation

Click to know more



Page no: 39

Introducing...

Complete Placement

Preparatory Masterclass



What is included?



400+ Hours Foundation of LIVE + Recorded Training on **Quant, Reasoning, Verbal, C, C++, Java, Python, DSA, OS, CN, DBMS**



Interview preparation Training along with One-to-One Mock **Technical & Personal Interviews** by experts



Latest Technologies Certification Courses to get higher packages. 500+ hours of courses on **Full stack development, AI, ML, Data Science, AWS Cloud Computing, IoT, Robotics, Tkinter, Django Data Analytics, Tableau, PowerBI and much more**



Company-specific LIVE and Recorded training for **30+ service and product-based companies.**

TCS NQT | Accenture | Capgemini | Cognizant | Infosys | Persistent | Deloitte | Mindtree | Virtusa | Goldman Sachs | Bosch | Samsung Amazon | Nalsoft | Zoho Cisco and 15+ more companies preparation.



15+ Real Time Projects based on Latest Technologies and **10+ Mini Projects** based on C, C++, Java and Python to build your Profile



Get TCS NQT Paid Exam for Free

Our Placement Reports

97.6%

Selection Ratio
of Complete Masterclass Students

4.91 / 5



Overall Rating
out of 5

Highest CTC
40 LPA

85% students
placed with CTC more
than 6 LPA

Average CTC
8.5 LPA

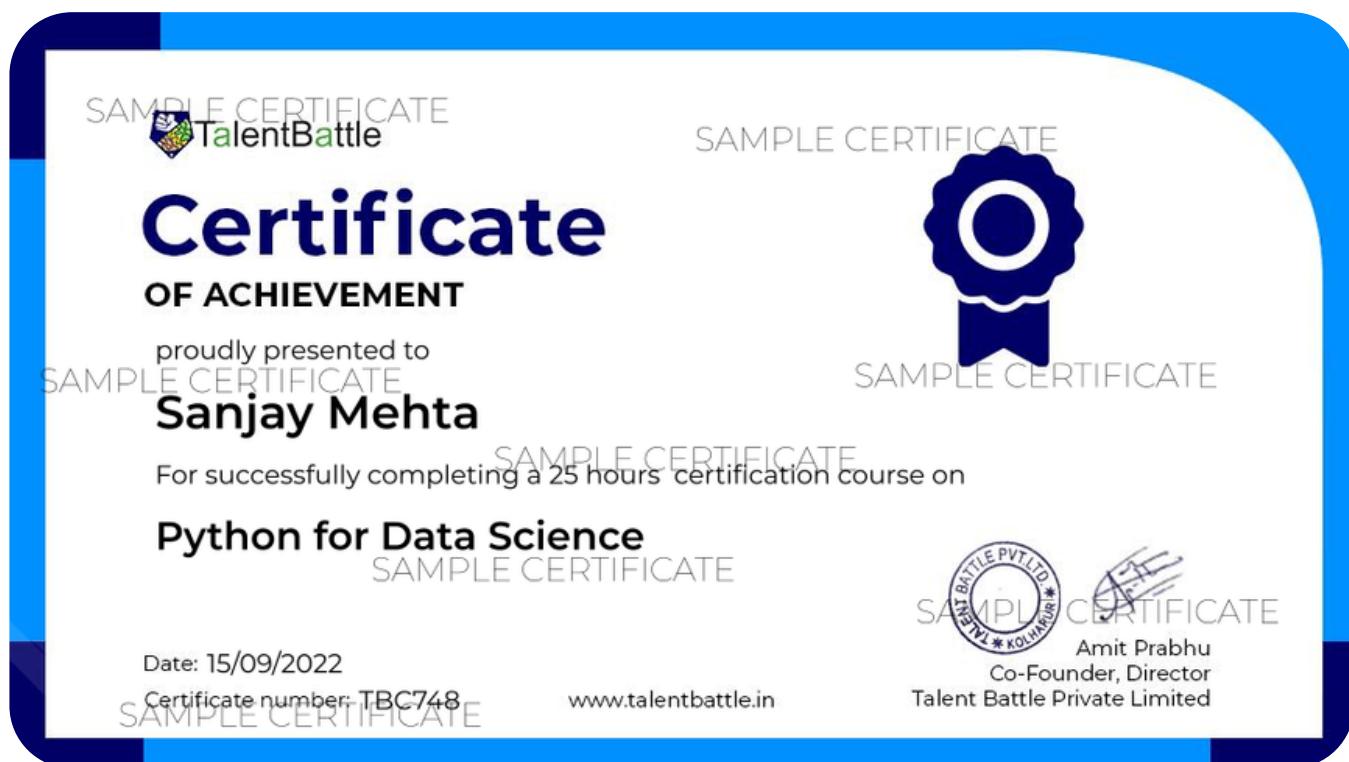
Referral Opportunities
in Top Companies & Startups
by Talent Battle

Average Offers
per student: 2.7

Students from
5000+ colleges
use Masterclass for
Placement Preparation

- **50000+ placed** students (in the last 10 years)
- Our Students placed in **600+ Companies**
- **10 Lakh+ Students** Prepare with Talent Battle Resources Every Year

Talent Battle Certifications



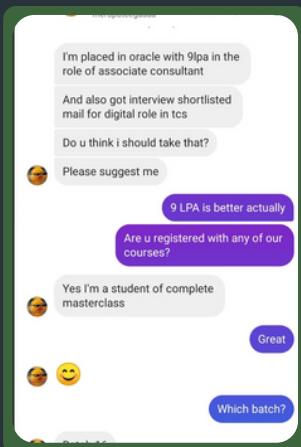
Get a Free Mentorship from experts for your Campus Placement Preparation

- Discuss your queries with experts
- Get a roadmap for your placement preparation

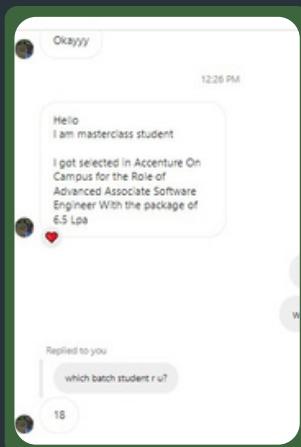
Click to know more



Some of our placed students



Srinija Kalluri
COMPLETE MASTERCLASS STUDENT
SELECTED AT ORACLE - 9 LPA



Rohit
COMPLETE MASTERCLASS STUDENT
SELECTED AT ACCENTURE - 6.5 LPA



Megha Ganguly
COMPLETE MASTERCLASS STUDENT
SELECTED COGNIZANT, WIPRO & BMC INDIA - 12.5 LPA



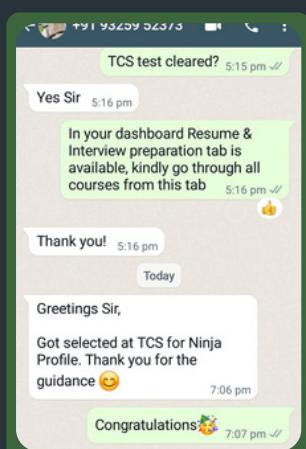
Aditya Kulestha
COMPLETE MASTERCLASS STUDENT
SELECTED AT COGNIZANT - 7.8 LPA



Shubham Verma
COMPLETE MASTERCLASS STUDENT
SELECTED AT CAPGEMINI - 7.5 LPA



Amardeep Prajapati
COMPLETE MASTERCLASS STUDENT
SELECTED AT HAPPIEST MIND TECHNOLOGY - 5.4 LPA



Rutuja Jangam
COMPLETE MASTERCLASS STUDENT
SELECTED AT TCS NINJA



Yogesh Lokhande
COMPLETE MASTERCLASS STUDENT
SELECTED AT HELLA ELECTRONICS - 5 LPA



Vikas Varak
COMPLETE MASTERCLASS STUDENT
SELECTED AT TCS NINJA

Tools & Technologies

covered in Placement Pro!



Our Team



Amit Prabhu

Co-founder

9+ years of experience in training students for Quantitative Aptitude, Verbal & Monitoring students for Campus Placement Preparation



Ajinkya Kulkarni

Co-founder

9+ years of experience in training students for Reasoning, Interview Training & Mentoring Students for Campus Placement Preparation



Rohit Bag
Lead Technical Trainer

10+ years of experience in training students for Programming Languages & Core Computer Science Subjects along with Company Specific Training



Vaishnavi K Dharan
Technical Trainer

5+ years of experience in training students on different Programming Languages and Data Structure. Master certified trainer in Robotic Process Automation in Automation Anywhere.



Poojitha Renati
Lead Aptitude Trainer

5+ years of experience in training students for Aptitude and Logical Reasoning. Trained more than 5000+ hours in various institutes including GITAM, Parul, KITS, JNTU and more



Chand Basha
Lead Aptitude Trainer

8+ years of experience in training students Quantitative Aptitude, Logical Reasoning & Verbal Ability for Campus Placements & Competitive Exams



Jasleen Chhabra
Mentor-Training and Placement

3+ years of experience in dealing with students and their counselling related to Academic problems as well as their placements.



Samradhni Wankhede
Graphic Designer

4+ years experience as a Creativity Expert, Graphic Designer, Teacher/Trainer and a social media enthusiast



Akshay Paricharak
Customer Service Manager

8+ years of experience in Customer service, Students counselling, Business development, Project co-ordination, Strategies Implementation, Planning and Execution.



Niranjan Kulkarni
Program Manager - Training and Placement

15 years of overall multi-functional experience in Industry and Academia, in managing diverse functions such as Training, and Human Resource Management.



Ruturaj Sankpal
Placement Mentor

2 years of experience in IT industry and currently mentoring students for campus placements.



Swapnil Wankhede
Marketing & Ops.

2.5+ years of experience in compassionate and ethical marketing. Striving to ensure students get the best opportunities and are prepared to make the most of it, learning and growing with Talent Battle.

Industry Mentors



Sandip Karekar
Industry Mentor

8 years of Industry experience and currently working with Mastercard. Decent understanding of core technologies in Computer Science & Information Technology and passionate about learning Cutting-Edge technologies.



Mayuresh Diwan
Industry Mentor

Placement Mentor: 4+ years of experience in automotive software development.



Swapnil Patil
Industry Mentor

Lead Engineer at John Deere
Over 4+ years of experience as a Software Developer. Having expertise in developing and maintaining web and desktop applications of different domains like Telecom, Simulations and Automations, ERP and CRM



Shadab Gada
Industry Mentor

Software developer with 3+ years of experience in design, developing, testing and maintenance of web based applications. Passionate about learning new technologies, always eager to share my knowledge, love interacting with new people.

FAQs

1 What is Talent Battle?

Talent battle is a group of mentors and educators who help students to prepare for their On and Off campus placement opportunities. We have trainers with an average of 10 years of experience in training students for their Tech company drives. We train students on Aptitude, Programming, communication skills, projects, advance technologies and all other necessary skills to get placed in their dream companies. If you want to get placed in any of your dream companies, then join our complete masterclass and fulfill your dream!

2 When and how to prepare for campus placements?

The best time to start preparing for your campus is in your third year of engineering. During this time you can start preparing for your Aptitude, Verbal and Programming skills.

Most of the companies have a similar testing pattern for selecting students. There are typically tests on aptitude, programming and communication skills. The short answer for this question is, **prepare with Talent Battle's Masterclass** as we cover all of the above mentioned topics in detail.

3 What is Complete Masterclass?

Complete Masterclass is a combination of Concept Clearing lectures for Aptitude, Coding, DSA + Company Specific Training for 30+ Companies + Interview Preparation with Mock Interviews. Foundational and Company Specific Training is conducted LIVE. Whenever companies launch their drives, we will be conducting company specific live sessions. Foundational training right from basic to advance level will also be available in recorded format.

Along with that we have 240+ hours of Full stack Development course, either of TCS ion internship or PAID NQT (for 2 years package) and 250+ hours of Advance certification courses like AI, ML, Data Science, etc will be available free of cost.

4 Why to chose Talent Battle?

We have structured and disciplined way of preparation. You don't need to seek outside information source. All the study material will be available on Talent Battle's dashboard. We provide end to end support to our students until they get placed. Talent Battle is one stop solution to prepare for placement drives.

Dont delay your placement preparation anymore!!

Learn from the experts!

Check out our social media to get regular placement related content & job drive updates

-  @talentbattle.in
-  @talentbattle_2023
-  @talentbattle_2024
-  @talentbattle_2025
-  @talentbattle_2026
-  WhatsApp Group
-  Free Mentorship
-  Talent Battle Facebook
-  Talent Battle YouTube
-  Talent Battle LinkedIn
-  <https://talentbattle.in/>

