

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=c4be012e-e120-44a0-a3c3-dfae363411e

NXT WAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

Introduction to Python Reading Material 25 mins

Conditional Statements and Loops Reading Material 15 mins

List and Functions Reading Material 25 mins

Tuples, Sets and Dictionaries Reading Material 25 mins

OOPs Reading Material 20 mins

Miscellaneous Topics Reading Material 20 mins

1. What is software?

Software is a set of instructions for the hardware.

2. What is Python and what are its applications?

Python is an object-oriented programming language that is easy to learn and simple to implement.

Applications of Python

Python is a versatile language that has applications in almost every field

- Artificial intelligence (AI)
- Machine Learning (ML)
- Big Data
- Smart Devices/Internet of Things (IoT)
- Cyber Security
- Game Development
- Backend Development, etc.

Complete the MCQs to proceed Continue

74°F Haze

Search

Windows Start

Google Chrome Microsoft Edge Microsoft Word Microsoft Excel LinkedIn Microsoft Teams Spotify Xbox

ENG IN

10:35 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=c4be012e-e120-44a0-a3c3-dfae363411e

NXT WAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

3. What are the features of Python?

Features of Python:

- Easy to learn & code
- Open Source Programming Language
- Object-Oriented Language
- Dynamic Typed Language
- Large Standard Library

4. Is Python case-sensitive?

Yes, Python is case-sensitive. The **username**, **UserName**, and **userName** are three different variables, and using these names interchangeably causes an error.

Code

```
1 username = "Rahul"
2 print(username)
3 print(userNmae)
```

PYTHON

Output

```
Rahul
NameError: name 'userNmae' is not defined
```

Complete the MCQs to proceed Continue

Upcoming Earnings 1

Search

10:36 13-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=c4be012e-e120-44a0-a3c3-dfae363411e

NXT WAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

NameError: name 'userName' is not defined

Q. Is Python a dynamically typed programming language?

Companies Asked: Translytics Business Services

Yes, Python is a dynamically typed language. This means that in Python the type checking of a variable is done only as code runs, and the type of a variable is allowed to change over its lifetime. There is no need to declare the type of the variable.

While programming languages like C, Java, C++, etc are statically typed languages where we cannot change the data type of a variable during the execution of the program.

Code

```
1 x = 6
2 print(type(x))
3 x = 'Rahul'
4 print(type(x))
```

PYTHON

Output

```
<class 'int'>
<class 'str'>
```

Complete the MCQs to proceed Continue

Upcoming Earnings 1

Search

Windows Start, Task View, File Explorer, Google Chrome, Microsoft Edge, Microsoft Word, Microsoft Excel, Microsoft Powerpoint, Microsoft Teams, Spotify, Xbox

ENG IN 10:36 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=c4be012e-e120-44a0-a3c3-dfae3634111e

NXT WAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Output

```
<class 'int'>
<class 'str'>
```

Question 1 of 4

What is the code to check the type of a variable in Python?

type(variable) typeof(variable) check_type(variable) variable.type()

Next

Complete the MCQs to proceed Continue

74°F Haze

Search

10:37 13-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=c4be012e-e120-44a0-a3c3-dfae363411e

NXT WAVE™

CONTENTS

Interview Questions (2 hours, 10 mins)

- Introduction to Python (25 mins)
- Conditional Statements and Loops (15 mins)
- List and Functions (25 mins)
- Tuples, Sets and Dictionaries (25 mins)
- OOPs (20 mins)
- Miscellaneous Topics (20 mins)

Float

List

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

6. What are the advantages of Python over Java?

Basis of Comparison	Python	Java
Learning curve	Easy to learn	Compared to Python, it's difficult to learn
Typing	Dynamically-typed	Statically-typed
Syntax	Easy to read and remember	Difficult to read and remember
Applications	Artificial Intelligence, Data Science and Machine Learning applications	Enterprise, Embedded and Cross-platform applications
Code Length	Fewer lines of code compared to Java	More lines of code compared to Python
Example Program	print("Hello World")	public class Simple { public static void main(String args[]){ System.out.println("Hello World"); } }

Continue

74°F Haze

Search

10:38 13-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=c4be012e-e120-44a0-a3c3-dfae363411e

Example Program

```
print("Hello World")
```

```
public class Simple {  
    public static void main(String args[]){  
        System.out.println("Hello World");  
    }  
}
```

Q. What are the specifications, benefits of Python compared to C.

Features	Python	C
Syntax	Python has a clean and readable syntax, making it easy for beginners to learn.	C has a more complex syntax compared to Python.
Formatting	It uses indentation to define blocks of code.	It uses curly braces {} to define blocks of code.
Typing	Python is dynamically typed, meaning variable types can be changed over time, offering flexibility.	C is statically typed, and variable types must be declared before use.
Memory Management	Python has automatic memory management through a garbage collector.	C requires manual memory management using functions like malloc() and free().
Execution	Python is an interpreted language, executing code line by line at runtime.	C is a compiled language, and the entire program is compiled before execution.
Performance	Python is generally slower than C due to its interpreted nature.	C is known for its high performance and efficiency.
Community	Python has a large and active community with extensive libraries and frameworks.	C has a well-established community, particularly in systems programming and embedded systems, but it may lack in modern web and application development libraries.

← Question 1 of 3

What is a benefit of using Python over C?

Python has a more complex syntax compared to C.

74°F Haze

Search

Google Chrome Microsoft Edge Microsoft Word Microsoft Excel Microsoft Powerpoint Microsoft OneDrive LinkedIn Microsoft Teams Spotify Xbox

ENG IN 10:39 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=c4be012e-e120-44a0-a3c3-dfae3634111e

	Management	Execution	Performance	Community
Management	Python uses a garbage collector.			C requires manual memory management using functions like malloc() and free().
Execution	Python is an interpreted language, executing code line by line at runtime.			C is a compiled language, and the entire program is compiled before execution.
Performance	Python is generally slower than C due to its interpreted nature.			C is known for its high performance and efficiency.
Community	Python has a large and active community with extensive libraries and frameworks.			C has a well-established community, particularly in systems programming and embedded systems, but it may lack in modern web and application development libraries.

← Question 2 of 3

Which language is known for manual memory management using functions like malloc() and free()?

- Python
- C
- Both Python and C
- None of the above

Next

74°F Haze

Search

Gmail

Google Chrome

Microsoft Edge

Microsoft Store

File Explorer

LinkedIn

Skype

Spotify

Xbox

10:40 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=c4be012e-e120-44a0-a3c3-dfae363411e

Execution	runtime.	C is a compiled language, and the entire program is compiled before execution.
Performance	Python is generally slower than C due to its interpreted nature.	C is known for its high performance and efficiency.
Community	Python has a large and active community with extensive libraries and frameworks.	C has a well-established community, particularly in systems programming and embedded systems, but it may lack in modern web and application development libraries.

◀ Question 3 of 3

Which programming language uses indentation to define code blocks?

- Python
- C
- Both Python and C
- Neither Python nor C

74°F Haze

Search

Windows Start

Icons: 🎭, 📱, 💬, 🚗, 🌐, 📁, 📲, 📺, 🎵, 🎮

Eng IN

10:40 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=c4be012e-e120-44a0-a3c3-dfae363411e

NXT
wAVG™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

7. How to perform the arithmetic operations using Python?

Addition

The addition is denoted by `+` sign.
It gives the sum of two numbers.

Code

```
1 print(2 + 5)
2 print(1 + 1.5)
```

PYTHON

Output

```
7
2.5
```

Subtraction

The subtraction is denoted by `-` sign.
It gives the difference between the two numbers.

Code

```
1 print(5 - 2)
```

PYTHON

Output

```
3
```

Continue

74°F Haze

Search

Windows Start, Task View, File Explorer, Google Chrome, Microsoft Edge, Microsoft Word, Microsoft Excel, Microsoft Powerpoint, Microsoft Teams, Spotify, Xbox

ENG IN

10:41 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=c4be012e-e120-44a0-a3c3-dfae363411e

NXT
WAVES™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Interview Questions > Introduction to Python Reading Material

Multiplication

The multiplication is denoted by * sign.

Code

```
1 print(2 * 5)
2 print(5 * 0.5)
```

PYTHON

Output

```
10
2.5
```

Division

The division is denoted by / sign.

Code

```
1 print(5 / 2)
2 print(4 / 2)
```

PYTHON

Output

```
2.5
2.0
```

Modulus

To find the remainder between two numbers, we use the Modulus operator %

74°F Haze

Search

L

10:42 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=c4be012e-e120-44a0-a3c3-dfae363411e

NXT
wAVe™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions (2 hours, 10 mins)

- Introduction to Python Reading Material (25 mins)
- Conditional Statements and Loops Reading Material (15 mins)
- List and Functions Reading Material (25 mins)
- Tuples, Sets and Dictionaries Reading Material (25 mins)
- OOPs Reading Material (20 mins)
- Miscellaneous Topics Reading Material (20 mins)
- More Interview Questions (20 mins)

Interview Questions > Introduction to Python Reading Material

Multiplication

The multiplication is denoted by `*` sign.

Code

```
1 print(2 * 5)
2 print(5 * 0.5)
```

PYTHON

Output

```
10
2.5
```

Division

The division is denoted by `/` sign.

Code

```
1 print(5 / 2)
2 print(4 / 2)
```

PYTHON

Output

```
2.5
2.0
```

Modulus

74°F Haze

Search

Windows Start

Google Chrome Microsoft Edge Microsoft Word Microsoft Excel Microsoft Powerpoint Microsoft Teams Microsoft OneDrive LinkedIn Microsoft Store Spotify Xbox

ENG IN

10:52 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=c4be012e-e120-44a0-a3c3-dfae363411e

NXT
wAVe™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Interview Questions > Introduction to Python Reading Material

Modulus

To find the remainder between two numbers, we use the Modulus operator %

Code

```
1 print(6 % 3)
```

PYTHON

Output

```
0
```

Exponent

To calculate a power b, we use Exponent Operator **

Code

```
1 print(2 ** 3)
```

PYTHON

Output

```
8
```

74°F Haze

Search

Windows Start

Google Chrome Microsoft Edge Microsoft Word Microsoft Excel Microsoft Powerpoint Microsoft OneDrive LinkedIn Microsoft Teams Spotify Xbox

ENG IN

10:53 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=c4be012e-e120-44a0-a3c3-dfae363411e

NXT
wAVe™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions (2 hours, 10 mins)

- Introduction to Python (25 mins)
 - Reading Material
- Conditional Statements and Loops (15 mins)
 - Reading Material
- List and Functions (25 mins)
 - Reading Material
- Tuples, Sets and Dictionaries (25 mins)
 - Reading Material
- OOPs (20 mins)
 - Reading Material
- Miscellaneous Topics (20 mins)
 - Reading Material
- More Interview Questions (20 mins)

8

Question 4 of 4

Which of the following is a common use case for the modulus operator in Python?

- To raise a number to a power
- To check if a number is odd or even
- To concatenate strings
- To perform integer division

Continue

Breaking news Unfolding now

Search

10:55 13-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=c4be012e-e120-44a0-a3c3-dfae363411e

NXT
wAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Exponent

To calculate a power **b**, we use Exponent Operator `**`

Code

```
1 print(2 ** 3)
```

PYTHON

Output

```
8
```

Question 4 of 4

Which of the following is a common use case for the modulus operator in Python?

- To raise a number to a power
- To check if a number is odd or even
- To concatenate strings
- To perform integer division

Complete the MCQs to proceed

Continue

Breaking news Unfolding now

Search

10:56 13-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=c4be012e-e120-44a0-a3c3-dfae3634111e

NXT
wAVG™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions (2 hours, 10 mins)

- Introduction to Python Reading Material (25 mins)
- Conditional Statements and Loops Reading Material (15 mins)
- List and Functions Reading Material (25 mins)
- Tuples, Sets and Dictionaries Reading Material (25 mins)
- OOPs Reading Material (20 mins)
- Miscellaneous Topics Reading Material (20 mins)
- More Interview Questions (20 mins)

Q. What is the output of `1 % 4` and `2 % 4` ?

- The output for `1 % 4` is:

```
1 print(1 % 4) # Output: 1
```

- The output for `2 % 4` is:

```
1 print(2 % 4) # Output: 2
```

← Question 1 of 3

What will be the output for the following code?

```
1 print(5%)
```

0
 5
 1
 3

Show Answer Submit

Complete the MCQs to proceed Continue

Breaking news Unfolding now

Search

Windows Start, Task View, File Explorer, Google Chrome, Microsoft Edge, Microsoft Word, Microsoft Excel, Microsoft Powerpoint, Microsoft Teams, Spotify, Xbox

ENG IN 10:56 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=c4be012e-e120-44a0-a3c3-dfae363411e

NXT
wAVe™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions (2 hours, 10 mins)

- Introduction to Python Reading Material (25 mins)
- Conditional Statements and Loops Reading Material (15 mins)
- List and Functions Reading Material (25 mins)
- Tuples, Sets and Dictionaries Reading Material (25 mins)
- OOPs Reading Material (20 mins)
- Miscellaneous Topics Reading Material (20 mins)
- More Interview Questions (20 mins)

9. What is Operator Precedence in Python?

The operator precedence determines which operator is executed first if there is more than one operator in an expression.

The operator precedence in Python is listed in the following table. It is in descending order (the upper group has higher precedence than the lower ones).

Operators	Meaning
()	Parentheses
**	Exponent
+X , -X , ~X	Unary plus, Unary minus, Bitwise NOT
*, /, //, %	Multiplication, Division, Floor division, Modulus
+, -	Addition, Subtraction
<<, >>	Bitwise shift operators
&	Bitwise AND
^	Bitwise XOR
	Bitwise OR
==, !=, >, >=, <, <=, is , is not , in , not in	Comparisons, Identity, Membership operators
not	Logical NOT
and	Logical AND
or	Logical OR

BODMAS

The standard order of evaluating an expression

- Brackets (B)
- Orders (O)
- Division (D)
- Multiplication (M)

Continue

Breaking news Unfolding now

Search

10:57 13-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=c4be012e-e120-44a0-a3c3-dfae363411e

NXT
wAVG™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions

Introduction to Python Reading Material

Conditional Statements and Loops Reading Material

List and Functions Reading Material

Tuples, Sets and Dictionaries Reading Material

OOPs Reading Material

Miscellaneous Topics Reading Material

More Interview Questions

BODMAS

The standard order of operations:

- Brackets (B)
- Orders (O)
- Division (D)
- Multiplication (M)
- Addition (A)
- Subtraction (S)

Expression: $(5 * 2) + (3 * 4 + (10) + (12 + 2)) - (10) + (14)$

Step by Step Explanation:

$(5 * 2) + (3 * 4 + (10) + (12 + 2)) - (10) + (14)$

Code

```
1 print((5 * 2) + (3 * 4 + (10) + (12 + 2)) - (10) + (14))
```

Output

24

Welcome to use Lenovo Smart Key!

Try Smart key to get quick access to Lenovo applications and services.

PYTHON

Next

Continue

Breaking news Unfolding now

Search

10:57 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=c4be012e-e120-44a0-a3c3-dfae363411e

NXT
wAVe™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions

Introduction to Python Reading Material

Conditional Statements and Loops Reading Material

List and Functions Reading Material

Tuples, Sets and Dictionaries Reading Material

OOPs Reading Material

Miscellaneous Topics Reading Material

More Interview Questions

BODMAS

The standard order of evaluating an expression

- Brackets (B)
- Orders (O)
- Division (D)
- Multiplication (M)
- Addition (A)
- Subtraction (S)

Expression: $(5 * 2) + (3 * 4 + 4 / 2)$

Step by Step Explanation

$(5 * 2) + (3 * 4 + 4 / 2)$
 $(10) + (3 * 4 + 2)$
 $(10) + (12 + 2)$
 $(10) + (14)$
24

Code

```
1 print((5 * 2) + (3 * 4 + 4 / 2))
```

PYTHON

Output

24

Continue

Breaking news
Unfolding now

Search

Windows Start

Google Chrome

Microsoft Edge

Microsoft Word

Microsoft Excel

Microsoft Powerpoint

Microsoft OneDrive

LinkedIn

Facebook

Twitter

Instagram

Spotify

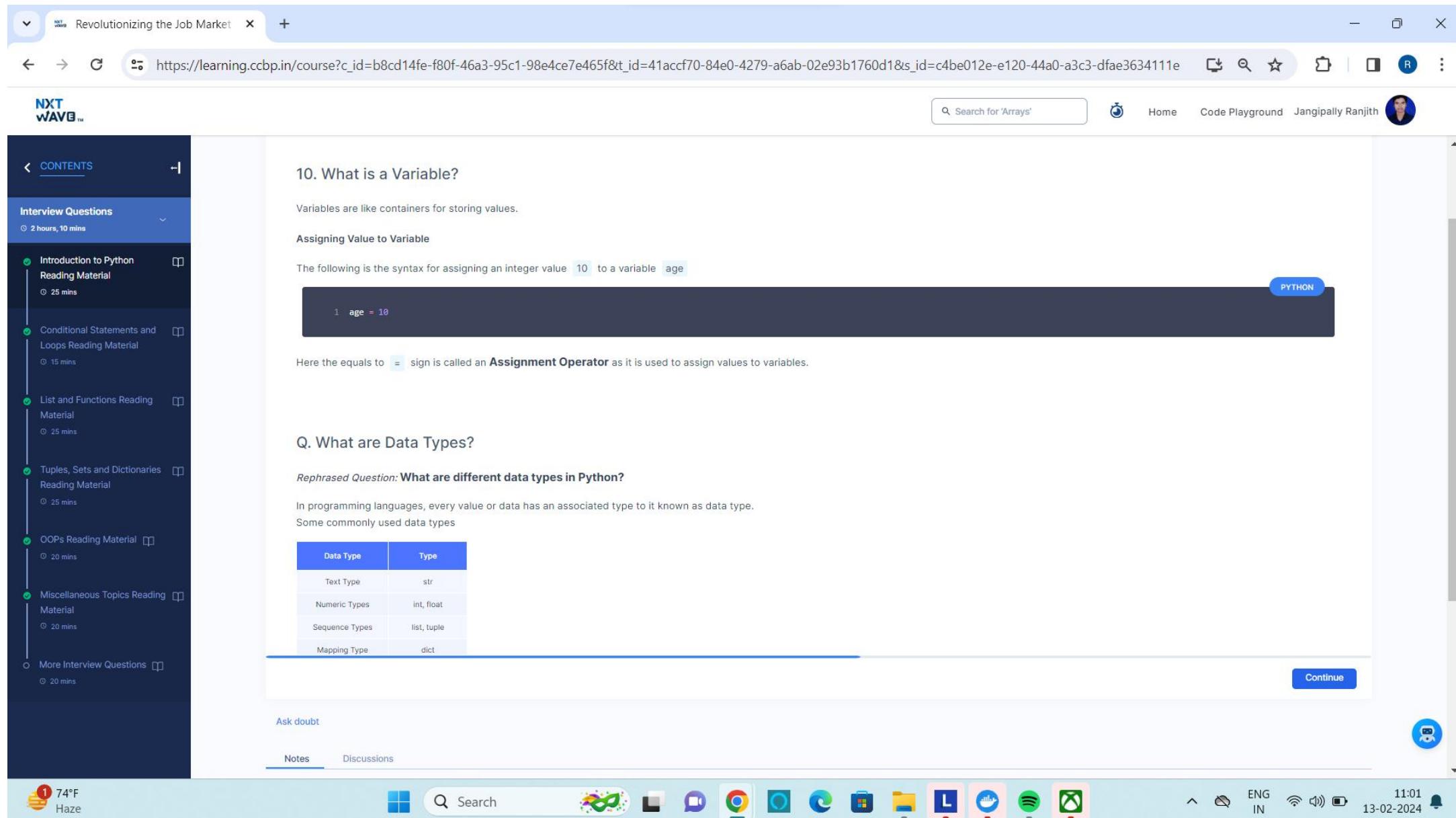
Xbox

Wi-Fi

Eng IN

10:58

13-02-2024



Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=c4be012e-e120-44a0-a3c3-dfae363411e

NXT
wAVe™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions (2 hours, 10 mins)

- Introduction to Python (25 mins) Reading Material
- Conditional Statements and Loops (15 mins) Reading Material
- List and Functions (25 mins) Reading Material
- Tuples, Sets and Dictionaries (25 mins) Reading Material
- OOPs (20 mins) Reading Material
- Miscellaneous Topics (20 mins) Reading Material
- More Interview Questions (20 mins) Reading Material

Q. What are Data Types?

Rephrased Question: What are different data types in Python?

In programming languages, every value or data has an associated type to it known as data type. Some commonly used data types

Data Type	Type
Text Type	str
Numeric Types	int, float
Sequence Types	list, tuple
Mapping Type	dict
Set Type	set
Boolean Type	bool

This data type determines how the value or data can be used in the program. For example, mathematical operations can be done on Integer and Float types of data.

▼ More Information

- Text Types:** The Text Data Type holds sequence of characters.
- Numeric Types:** The Numeric Data Type holds numeric values.
- Sequence Types:** The Sequence Data Type holds collection of items.
- Mapping Type:** The Mapping Data Type holds data in key-value pair form.
- Set Type:** The Set Data Type hold collection of unique items.
- Boolean Type:** The Boolean Data Types holds either `True` or `False`.

Continue

Ask doubt

Notes Discussions

74°F Haze

Search

Windows Start, Mask, Photo, Chat, Google Chrome, Microsoft Edge, Microsoft Store, File Explorer, LinkedIn, Microsoft Teams, Spotify, Xbox

ENG IN 11:01 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=c4be012e-e120-44a0-a3c3-dfae363411e

NXT
wAVG™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions (2 hours, 10 mins)

- Introduction to Python Reading Material (25 mins)
- Conditional Statements and Loops Reading Material (15 mins)
- List and Functions Reading Material (25 mins)
- Tuples, Sets and Dictionaries Reading Material (25 mins)
- OOPs Reading Material (20 mins)
- Miscellaneous Topics Reading Material (20 mins)
- More Interview Questions (20 mins)

Data Type	Type
Text Type	str
Numeric Types	int, float
Sequence Types	list, tuple
Mapping Type	dict
Set Type	set
Boolean Type	bool

This data type determines how the value or data can be used in the program. For example, mathematical operations can be done on Integer and Float types of data.

More Information

- Text Types:** The Text Data Type holds sequence of characters.
- Numeric Types:** The Numeric Data Type holds numeric values.
- Sequence Types:** The Sequence Data Type holds collection of items.
- Mapping Type:** The Mapping Data Type holds data in key-value pair form.
- Set Type:** The Set Data Type hold collection of unique items.
- Boolean Type:** The Boolean Data Types holds either True or False .

Ask doubt

Notes Discussions

+ NEW NOTE

74°F Haze

Search

11:01 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=c4be012e-e120-44a0-a3c3-dfae363411e

NXT
wAVe™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

12. What are the numeric data types in Python?

The Numeric Data Types in Python are:

- Integers
- Float
- Complex Numbers

Code

```
1 a = 10
2 print("Type of a: ", type(a))
3
4 b = 10.0
5 print("Type of b: ", type(b))
6
7 c = 10 + 20j
8 print("Type of c: ", type(c))
```

PYTHON

Output

```
Type of a: <class 'int'>
Type of b: <class 'float'>
Type of c: <class 'complex'>
```

Continue

74°F Haze

Search

Windows Start

Icons for various applications: Mask, Photo, Chat, Google Chrome, Microsoft Edge, Microsoft Word, File Explorer, LinkedIn, Microsoft Teams, Spotify, Xbox

Network, Battery, Volume, and Notifications icons

ENG IN 11:02 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=c4be012e-e120-44a0-a3c3-dfae363411e

NXT
wAVe™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions (2 hours, 10 mins)

- Introduction to Python Reading Material (25 mins)
- Conditional Statements and Loops Reading Material (15 mins)
- List and Functions Reading Material (25 mins)
- Tuples, Sets and Dictionaries Reading Material (25 mins)
- OOPs Reading Material (20 mins)
- Miscellaneous Topics Reading Material (20 mins)
- More Interview Questions (20 mins)

13. What is meant by mutability? Name some mutable data types?

Mutable means capable of being changed. In Python, objects whose value can be changed are said to be mutable.

Some of the mutable data types in Python are list, dictionary, set and user-defined classes.

14. What is meant by immutability? Name some immutable data types?

Immutable means capable of not being changed. In Python, objects whose value cannot be changed are said to be immutable.

Some of the immutable data types in Python are tuple, integer, boolean, string, etc.

← Question 1 of 2

What does it mean for a data type to be `immutable` ?

- It can be modified after creation
- It cannot be changed after creation
- It is always a string
- None of the above

Next

Complete the MCQs to proceed Continue

74°F Haze

Search

11:03 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=c4be012e-e120-44a0-a3c3-dfae3634111e

NXT
wAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions (2 hours, 10 mins)

- Introduction to Python Reading Material (25 mins)
- Conditional Statements and Loops Reading Material (15 mins)
- List and Functions Reading Material (25 mins)
- Tuples, Sets and Dictionaries Reading Material (25 mins)
- OOPs Reading Material (20 mins)
- Miscellaneous Topics Reading Material (20 mins)
- More Interview Questions (20 mins)

Q. What are the differences between the mutable and immutable data types?

Rephrased Question: What is immutability and mutability?

Mutable and **Immutable** are terms primarily used to refer to whether an object can be changed (modified) after it's created.

Immutable Data Types	Mutable Data Types
Cannot be changed after creation	Can be modified after creation
Can be used as dictionary keys	Cannot be used as dictionary keys
Reduces potential bugs from unintended side effects	More prone to bugs if not handled carefully
Examples: int, float, str, tuple	Examples: list, dict, set
Creating a new instance doesn't affect the original object	A shallow copy references the same objects in the original object; a deep copy creates an independent object
For some operations, immutable data types might require the creation of many temporary objects, which can be inefficient	Adding an element to a list or modifying a dict in-place can be efficient

Question 1 of 5

Which of the following data types cannot be used as dictionary keys in Python?

- str
- tuple
- list
- int

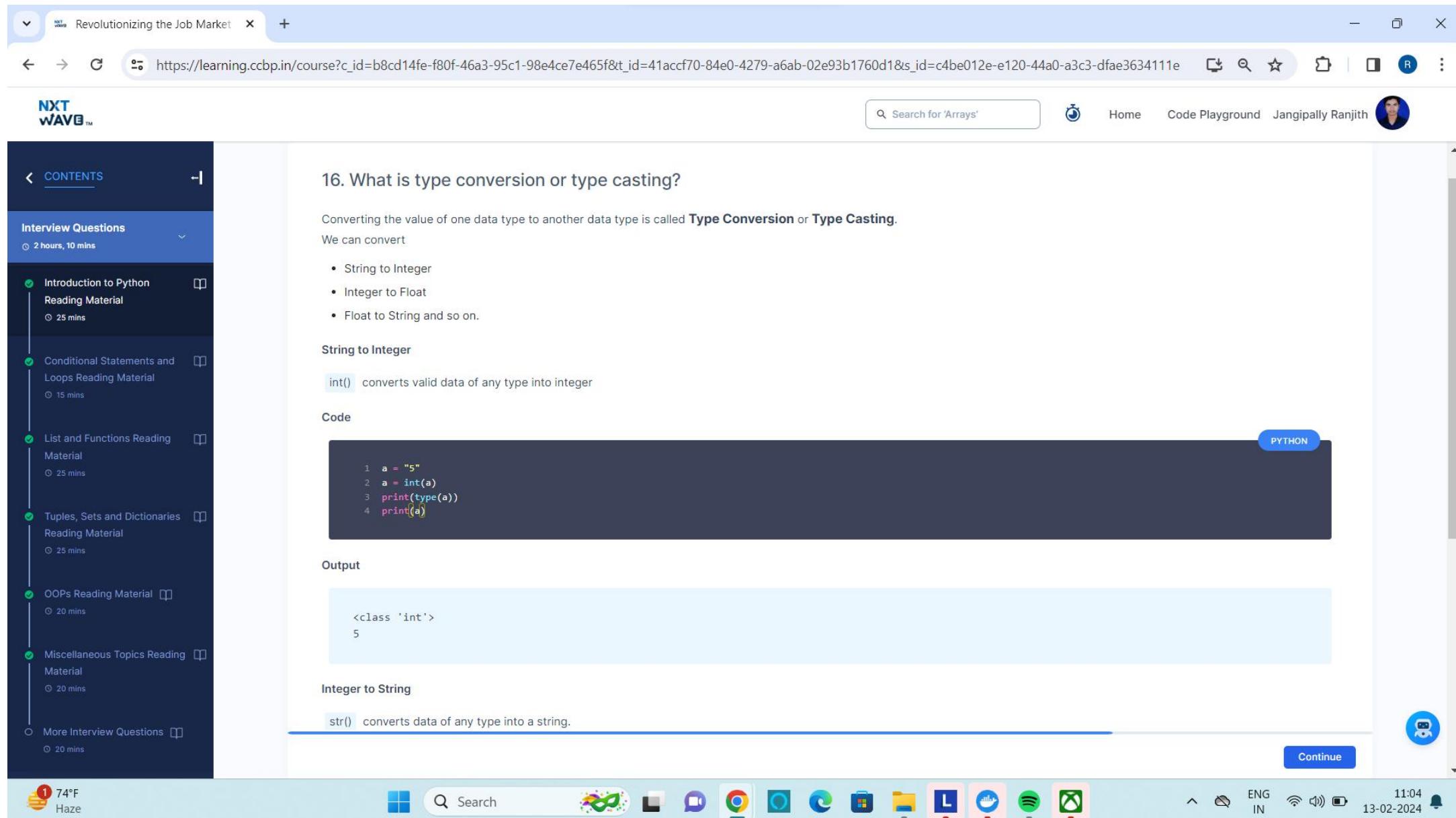
Next

Complete the MCQs to proceed Continue

74°F Haze

Search

11:04 13-02-2024 ENG IN



Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=c4be012e-e120-44a0-a3c3-dfae363411e

NXT
wAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Interview Questions > Introduction to Python Reading Material

Integer to String

`str()` converts data of any type into a string.

Code

PYTHON

```
1 a = input()
2 a = int(a)
3 b = input()
4 b = int(b)
5 result = a + b
6 print("Sum: " + str(result))
```

Input

```
2
3
```

Output

```
Sum: 5
```

Similarly,

- `float()` → Converts to a float data type
- `bool()` → Converts to a boolean data type

74°F Haze

Search

11:05 13-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=c4be012e-e120-44a0-a3c3-dfae363411e

NXT
wAVe™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

17. What is a String?

A String is a stream of characters enclosed within quotes.

Stream of Characters

- Capital Letters (A – Z)
- Small Letters (a – z)
- Digits (0 – 9)
- Special Characters (~ ! @ # \$ % ^ . ? ,)
- Space

Some examples:

- "Hello, World!"
- "some@example.com"
- "1234"

18. What is String Slicing?

Obtaining a part of a string is called String Slicing.

Syntax:

```
variable_name[start_index:end_index]
```

- end_index is not included in the slice.

Code

PYTHON

Continue

74°F Haze

Search

13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=c4be012e-e120-44a0-a3c3-dfae3634111e

NXT
wAVe™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

18. What is String Slicing?

Obtaining a part of a string is called String Slicing.

Syntax:

```
variable_name[start_index:end_index]
```

- `end_index` is not included in the slice.

Code

```
1 message = "Hi Ravi"
2 part = message[3:7]
3 print(part)
```

PYTHON

Output

```
Ravi
```

Slicing to End

If the end index is not specified, slicing stops at the end of the string.

Code

```
1 message = "Hi Ravi"
2 part = message[3:]
3 print(part)
```

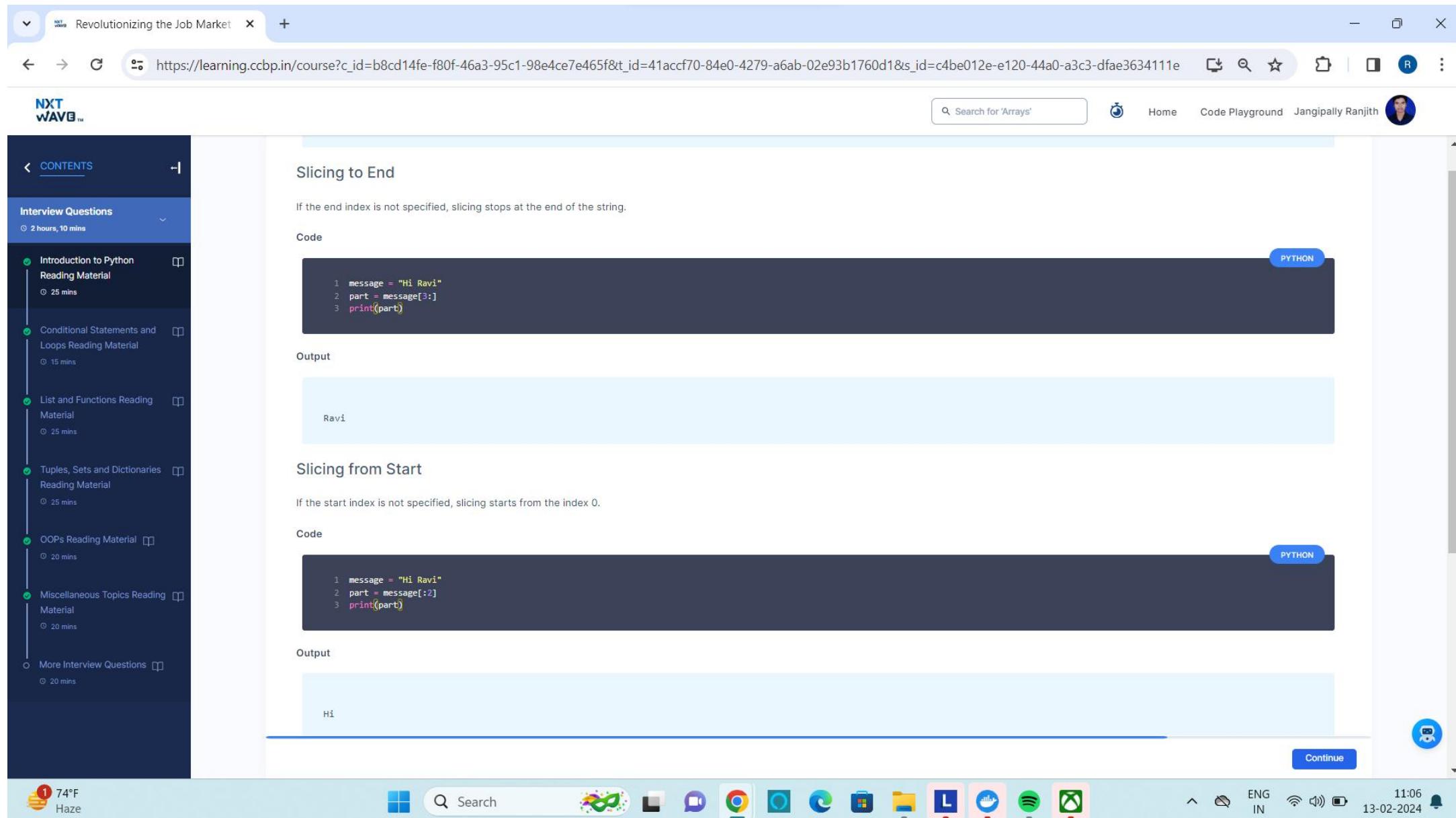
PYTHON

Continue

74°F Haze

Search

11:06 13-02-2024 ENG IN



Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=c4be012e-e120-44a0-a3c3-dfae363411e

NXT
wAVe™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPS Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Extended Slicing

Syntax: variable[start_index:end_index:step]

[1:8:2]

Step determines the increment between each index for slicing.

Code

PYTHON

```
1 a = "Waterfall"
2 part = a[1:8:2]
3 print(part)
```

Output

aefl

Continue

74°F Haze

Search

11:06 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=c4be012e-e120-44a0-a3c3-dfae363411e

NXT
WAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

19. How to reverse a string?

A string can be reversed using extended slicing.

Syntax:

```
variable[start:end:negative_step]
```

-1 for step will reverse the order of the characters.

Code

```
1 string_1 = "Program"
2 string_2 = string_1[::-1]
3 print(string_2)
```

PYTHON

Output

```
margorP
```

20. What is string `capitalize()` in Python?

The `capitalize()` method converts the first character of a string to an uppercase letter and all other alphabets to lowercase.

Code

```
[REDACTED]
```

PYTHON

Continue

74°F Haze

Search

11:06 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=c4be012e-e120-44a0-a3c3-dfae363411e

NXT
wAVe™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

20. What is string `capitalize()` in Python?

The `capitalize()` method converts the first character of a string to an uppercase letter and all other alphabets to lowercase.

Code

PYTHON

```
1 sentence = "proGraMmiNg"
2 capitalized_string = sentence.capitalize()
3 print(capitalized_string)
```

Output

Programming

21. What is string `replace()` in Python?

The `replace()` returns a new string after replacing all the occurrences of the old substring with the new substring.

Syntax: `str_var.replace(old, new)`

Code

PYTHON

```
1 sentence = "teh cat and teh dog"
2 sentence = sentence.replace("teh", "the")
3 print(sentence)
```

Output

Continue

74°F Haze

Search

11:07 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=c4be012e-e120-44a0-a3c3-dfae363411e

NXT
wAVe™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPS Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Output

Programming

21. What is string `replace()` in Python?

The `replace()` returns a new string after replacing all the occurrences of the old substring with the new substring.

Syntax: `str_var.replace(old, new)`

Code

```
1 sentence = "teh cat and teh dog"
2 sentence = sentence.replace("teh", "the")
3 print(sentence)
```

PYTHON

Output

the cat and the dog

Snipping Tool

Screenshot copied to clipboard and saved Select here to mark up and share the image

74°F Haze

Search

11:07 13-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=c4be012e-e120-44a0-a3c3-dfae363411e

NXT
wAVG™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions (2 hours, 10 mins)

- Introduction to Python Reading Material (25 mins)
- Conditional Statements and Loops Reading Material (15 mins)
- List and Functions Reading Material (25 mins)
- Tuples, Sets and Dictionaries Reading Material (25 mins)
- OOPs Reading Material (20 mins)
- Miscellaneous Topics Reading Material (20 mins)
- More Interview Questions (20 mins)

22. What is `round()` function?

Rounds the float value to the given number of decimal digits.

Syntax:

```
round(number, digits(optional))
```

`digits` → defines the number of decimal digits to be considered for rounding.

- When `digits` not specified, the default value is **0**.

Code

```
1 a = round(3.14159, 2)
2 print(a)
3 a = round(5.6777)
4 print(a)
```

PYTHON

Output

```
3.14
6
```

23. How to write comments in Python?

A comment starts with a hash `#`

It can be written in its own line next to a statement of code.

Continue

74°F Haze

Search

Icons: Windows, Search, Mask, Photo, Chat, Google Chrome, Microsoft Edge, Microsoft Word, File, LinkedIn, Microsoft Teams, Spotify, Xbox

Eng IN 11:08 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=c4be012e-e120-44a0-a3c3-dfae363411e

NXT
WAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

23. How to write comments in Python?

A comment starts with a hash #

It can be written in its own line next to a statement of code.

Code

```
1 n = 5
2 # Finding if Even
3 even = (n % 2 == 0)
4 print(even) # prints boolean value
```

PYTHON

Output

False

Snipping Tool

Screenshot copied to clipboard and saved
Select here to mark up and share the image

74°F Haze

Search

11:08 13-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=c4be012e-e120-44a0-a3c3-dfae363411e

NXT
wAVe™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. What are Iterators in Python?

An iterator is a special object in Python that allows us to traverse through a collection of items one at a time.

In Python, we implement two special methods known as iterators methods:

Method	Description
<code>__iter__()</code>	Called to initialize the iterator. It must return an iterator object.
<code>__next__()</code>	Called to iterate over the iterator. It must return the next value in the data stream.

► More Information

Question 1 of 5

What is the purpose of the `__iter__()` method in Python iterators?

- It initializes the iterator.
- It returns the next value in the data stream.
- It creates the iterator object.
- It defines the collection of items to be iterated.

Next

74°F Haze

Search

Windows Start

Google Chrome Microsoft Edge Microsoft Word Microsoft Excel Microsoft Powerpoint Microsoft OneDrive LinkedIn Microsoft Teams Spotify Xbox

ENG IN 11:08 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=5b8e7625-f223-46e0-b16a-996a6132fbf2

NXT
WAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions (2 hours, 10 mins)

- Introduction to Python Reading Material (25 mins)
- Conditional Statements and Loops Reading Material (15 mins)
- List and Functions Reading Material (25 mins)
- Tuples, Sets and Dictionaries Reading Material (25 mins)
- OOPs Reading Material (20 mins)
- Miscellaneous Topics Reading Material (20 mins)
- More Interview Questions (20 mins)

Interview Questions

Recently Asked Questions and Answers

1. What are logical operators?

The logical operators are used to perform logical operations on Boolean values. Gives `True` or `False` as a result.

Following are the logical operators

- `and`
- `or`
- `not`

Logical AND Operator

Gives `True` if both the booleans are true, else, it gives `False`

Code

```
1 print(True and True)
```

PYTHON

Output

```
True
```

Continue

74°F Haze

Search

Windows Start

Google Chrome

Microsoft Edge

Microsoft Word

Microsoft Excel

Microsoft Powerpoint

Microsoft OneDrive

LinkedIn

Facebook

Twitter

Spotify

Xbox

Wi-Fi

Bluetooth

Screen Resolution

Volume

Battery

11:09

ENG IN

13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=5b8e7625-f223-46e0-b16a-996a6132fbf2

NXT
WAVES™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions (2 hours, 10 mins)

- Introduction to Python Reading Material (25 mins)
- Conditional Statements and Loops Reading Material (15 mins)
- List and Functions Reading Material (25 mins)
- Tuples, Sets and Dictionaries Reading Material (25 mins)
- OOPs Reading Material (20 mins)
- Miscellaneous Topics Reading Material (20 mins)
- More Interview Questions (20 mins)

Example

Code

```
1 print((2 < 3) and (1 < 2))
```

PYTHON

Step by Step Explanation

```
(2 < 3) and (1 < 2)
True and (1 < 2)
True and True
```

Output

```
True
```

Logical OR Operator

Gives `True` if any one of the booleans is true, else, it gives `False`

Code

```
1 print(False or False)
```

PYTHON

Output

```
False
```

Continue

74°F Haze

Search

Windows Start

Google Chrome

Microsoft Edge

Microsoft Word

Microsoft Excel

Microsoft Powerpoint

LinkedIn

Facebook

Twitter

Spotify

Xbox

IN 11:09 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=5b8e7625-f223-46e0-b16a-996a6132fbf2

NXT
wAVe™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Example

Code

```
1 print((2 > 3) or (2 < 1))
```

PYTHON

Step by Step Explanation

```
(2 > 3) or (2 < 1)
False or (2 < 1)
False or False
```

Output

```
False
```

Logical NOT Operator

Gives the opposite value of the given boolean.

Code

```
1 print(not(False))
```

Output

```
True
```

Snipping Tool

Screenshot copied to clipboard and saved
Select here to mark up and share the image

74°F Haze

Search

L

11:09 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=5b8e7625-f223-46e0-b16a-996a6132fbf2

NXT
wAVe™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Logical OR Operator

Gives `True` if any one of the booleans is true, else, it gives `False`

Code

```
1 print(False or False)
```

PYTHON

Output

```
False
```

Example

Code

```
1 print((2 > 3) or (2 < 1))
```

PYTHON

Step by Step Explanation

```
(2 > 3) or (2 < 1)
False or (2 < 1)
False or False
```

Output

Continue

74°F Haze

Search

Windows Start

Google Chrome Microsoft Edge Microsoft Word Microsoft Excel Microsoft Powerpoint Microsoft Teams Microsoft OneDrive LinkedIn Microsoft Office 365 Spotify Xbox

ENG IN

11:10 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=5b8e7625-f223-46e0-b16a-996a6132fbf2

NXT
wAVe™

CONTENTS

Interview Questions (2 hours, 10 mins)

- Introduction to Python (Reading Material, 25 mins)
- Conditional Statements and Loops (Reading Material, 15 mins)
- List and Functions (Reading Material, 25 mins)
- Tuples, Sets and Dictionaries (Reading Material, 25 mins)
- OOPs (Reading Material, 20 mins)
- Miscellaneous Topics (Reading Material, 20 mins)
- More Interview Questions (20 mins)

Output

```
False
```

Logical NOT Operator

Gives the opposite value of the given boolean.

Code

```
1 print(not(False))
```

PYTHON

Output

```
True
```

Example

Code

```
1 print(not(2 < 3))
```

PYTHON

Step by Step Explanation

```
not(2 < 3)
not(True)
False
```

Continue

74°F Haze

Search

Windows Start

Google Chrome

Microsoft Edge

Microsoft Word

Microsoft Excel

Microsoft Powerpoint

Microsoft Teams

Microsoft Word

Microsoft Excel

Microsoft Powerpoint

Microsoft Teams

Spotify

Xbox

ENG IN

11:10

13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=5b8e7625-f223-46e0-b16a-996a6132fbf2

NXT
wAVe™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Gives the opposite value of the given boolean.

Code

```
1 print(not(False))
```

PYTHON

Output

True

Example

Code

```
1 print(not(2 < 3))
```

PYTHON

Step by Step Explanation

```
not(2 < 3)
not(True)
False
```

Output

False

Snipping Tool

Screenshot copied to clipboard and saved Select here to mark up and share the image

74°F Haze

Search

L

11:10 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=5b8e7625-f223-46e0-b16a-996a6132fbf2

NXT
wAVe™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPS Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

2. What are conditional statements?

The Conditional Statement allows you to execute a block of code based on a condition.

If statement

The Conditional Statement allows you to execute a block of code only when a specific condition is `True`.

`if condition:`

code

```
1 if True:
2     print("If Block")
3     print("Inside If")
```

Output

```
If Block
Inside If
```

If-Else statement

Snipping Tool

Screenshot copied to clipboard and saved Select here to mark up and share the image

74°F Haze

Search

L

11:11 ENG IN

13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=5b8e7625-f223-46e0-b16a-996a6132fbf2

Home Code Playground Jangipally Ranjith

CONTENTS

If-Else statement

When If-Else conditional statement is used, the Else block of code executes if the condition is False .

if condition:

Block of code

else:

Indentation ← Block of code → Executes if the condition is **False**

Using If-Else

Code

```
1 a = int(input())
2 if a > 0:
3     print("Positive")
4 else:
5     print("Not Positive")
6 print("End")
```

PYTHON

Input

2

Output

Positive
End

Snipping Tool

Screenshot copied to clipboard and saved
Select here to mark up and share the image

74°F Haze

Search

11:11 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=5b8e7625-f223-46e0-b16a-996a6132fbf2

NXT
WAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

For Loop

4. What is the use of `while` loop?

While loop allows us to execute a block of code several times as long as the condition is `True`.

Initialization

while termination_condition:

Block of Code

Updation

Example

The following code snippet prints the next three consecutive numbers after a given number.

Code

```
1 a = int(input())
2 counter = 0
3 while counter < 3:
4     a = a + 1
5     print(a)
6     counter = counter + 1
```

PYTHON

Continue

74°F Haze

Search

11:12 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=5b8e7625-f223-46e0-b16a-996a6132fbf2

NXT
WAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Example

The following code snippet prints the next three consecutive numbers after a given number.

Code

PYTHON

```
1 a = int(input())
2 counter = 0
3 while counter < 3:
4     a = a + 1
5     print(a)
6     counter = counter + 1
```

Input

4

Output

5
6
7

5. How to create a do-while loop in python?

Generally, in other programming languages, we will have another loop called the do-while loop.

In do-while loops, the code in the loop runs at least once before checking if the condition is True.

Snipping Tool

Screenshot copied to clipboard and saved
Select here to mark up and share the image

74°F Haze

Search

L

11:12 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=5b8e7625-f223-46e0-b16a-996a6132fbf2

NXT
WAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions (2 hours, 10 mins)

- Introduction to Python Reading Material (25 mins)
- Conditional Statements and Loops Reading Material (15 mins)
- List and Functions Reading Material (25 mins)
- Tuples, Sets and Dictionaries Reading Material (25 mins)
- OOPs Reading Material (20 mins)
- Miscellaneous Topics Reading Material (20 mins)
- More Interview Questions (20 mins)

5. How to create a do-while loop in python?

Generally, in other programming languages, we will have another loop called the `do-while` loop.

In `do-while` loops, the code in the loop runs at least once before checking if the condition is **True**.

The `do-while` loop is not directly available in Python.

Example: do-while loop in c language

code

```
1 #include <stdio.h>
2
3 int main () {
4     int a = 1;
5
6     do {
7         printf("%d\n", a);
8         a = a + 1;
9     }while( a < 4 );
10
```

output

```
1
2
3
```

If the condition checked evaluates to **True**, the loop continues. For the cases where you would want your code to run at least one time then the `do-while` loops will come in handy.

Continue

74°F Haze

Search

11:12 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=5b8e7625-f223-46e0-b16a-996a6132fbf2

NXT
WAVES™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

output

```
1
2
3
```

If the condition checked evaluates to **True**, the loop continues. For the cases where you would want your code to run at least one time then the `do-while` loops will come in handy.

do-while loop in python:

In Python, we can create a `do-while` loop by using the `while` loop to achieve similar behavior.

Code

PYTHON

```
1 i = 1
2
3 while True:
4     print(i)
5     i = i + 1
6     if(i > 3):
7         break
```

Output

```
1
2
3
```

6. What is the use of `for` loop?

Continue

Breaking news Unfolding now

Search

11:32 13-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=5b8e7625-f223-46e0-b16a-996a6132fbf2

NXT
wAVe™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

6. What is the use of `for` loop?

The `for` loop is used to execute a block of code a known number of times. The `for` statement iterates over each item of a sequence.

for each_item in sequence:

Examples of sequences:

- The sequence of Characters (string)
- The sequence of numbers, etc.

Code

```
1 word = "Python"
2 for each_char in word:
3     print(each_char)
```

PYTHON

Output

```
P
y
t
h
o
n
```

Continue

Breaking news Unfolding now

Search

11:32 13-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=5b8e7625-f223-46e0-b16a-996a6132fbf2

NXT
WAVES™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

7. What are `range()` and `xrange()` functions?

`range():`

The `range()` function generates a sequence of integers starting from 0 to n(n is not included) and returns it.

Syntax: `range(n)`

Code

```
1 - for number in range(3):
2     print(number)
```

PYTHON

Output

```
0
1
2
```

Range with Start and End

Syntax: `range(start, end)`

Generates a sequence of numbers starting from `start` to `end` (`end` is not included).

Code

```
1 - for number in range(5, 8):
2     print(number)
```

PYTHON

Continue

Breaking news Unfolding now

Search

11:33 13-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=5b8e7625-f223-46e0-b16a-996a6132fbf2

NXT
WAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Range with Start and End

Syntax: range(start, end)

Generates a sequence of numbers starting from `start` to `end` (`end` is not included).

Code

PYTHON

```
1 - for number in range(5, 8):
2 -     print(number)
```

Output

```
5
6
7
```

xrange():

The `xrange()` function generates a sequence of integers starting from 0 similar to `range()` function.

Code

PYTHON

```
1 - for number in xrange(4):
2 -     print(number)
```

Output

```
0
1
2
3
```

Continue

Breaking news Unfolding now

Search

11:33 13-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=5b8e7625-f223-46e0-b16a-996a6132fbf2

NXT
wAVG™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Note: The `xrange()` function is deprecated from Python3. You cannot run `xrange()` function in our code playground.

Q. What is the range function in Python?

The `range()` function returns a sequence of numbers, starting from 0 by default, and increment the step by 1 (by default), and stops before a specified number.

Syntax:

```
range([start, stop, step])
```

```
1 x = range(3, 9, 2)
2 for n in x:
3     print(n) #Output: 3 5 7
```

PYTHON

Question 1 of 6

By default, what is the starting number when using `range()` in Python?

- 1
- 0
- 1
- It starts at the end of the list.

Next

Complete the MCQs to proceed Continue

Breaking news Unfolding now

Search

11:34 13-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=5b8e7625-f223-46e0-b16a-996a6132fbf2

NXT
WAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

8. When does an infinite loop occurs?

An infinite loop occurs when the condition always evaluates to True i.e. incorrect termination condition.

Example

Code

PYTHON

```
1 a = 10
2 while a > 3:
3     a = a + 1
4     print(a)
```

In the above code the while loop will run infinite times as the condition always evaluates to True .

9. What are Nested Loops?

An inner loop within the repeating block of an outer loop is called Nested Loop.

The **Inner Loop** will be executed one time for each iteration of the **Outer Loop**.

for item in sequence A: Outer Loop

for item in sequence B: Inner Loop

Block 1

Block 2

Block 3

Continue

Breaking news Unfolding now

Search

L 11:34 13-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=5b8e7625-f223-46e0-b16a-996a6132fbf2

NXT WAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Interview Questions > Conditional Statements and Loops Reading Material

9. What are Nested Loops?

An inner loop within the repeating block of an outer loop is called Nested Loop.

The **Inner Loop** will be executed one time for each iteration of the **Outer Loop**.

for item in sequence A: Outer Loop

for item in sequence B: Inner Loop

Code

PYTHON

```
1 - for i in range(2):
2 -     print("Outer: " + str(i))
3 -     for j in range(2):
4 -         print("  Inner: " + str(j))
```

Output

```
Outer: 0
  Inner: 0
  Inner: 1
Outer: 1
  Inner: 0
  Inner: 1
```

74°F Haze

Search

11:35 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=5b8e7625-f223-46e0-b16a-996a6132fbf2

NXT wAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. How does Inner loop work in nested loops in Python?

In Python, the term **inner loop** typically refers to a loop that is nested (inside) within another loop.

Example:

```
1 - for i in range(3): # Outer loop
2 -     print("Outer loop iteration:", i)
3 -
4 -     for j in range(2): # Inner loop
5 -         print("Inner loop iteration:", j)
```

Explanation:

We have two loops here, first one is the `Outer loop` and the nested one is the `Inner loop`. For each iteration of the outer loop, the inner loop runs completely from start to finish. So, for each value of `i` in the outer loop, the inner loop executes its iterations.

Question 1 of 2

How many times the `print` statement is going to execute in the given Python code?

```
1 - for i in range(4): # Outer loop
2 -     for j in range(3): # Inner loop
3 -         print("Executing inner loop iteration:", j)
```

4 times
 3 times
 12 times

Complete the MCQs to proceed Continue

74°F Haze

Search

11:35 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=5b8e7625-f223-46e0-b16a-996a6132fbf2

NXT
wAVe™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

10. What is a break statement?/ How to exit from a loop?

Break statement makes the program exit a loop early.

```
while condition:  
    [loop body]  
    break
```

Using Break

Generally, `break` is used to exit a loop when a condition is satisfied.

```
for item in Sequence:  
    [loop body]  
    if condition B:  
        break
```

In the below example, when the variable `i` value equals to `3` then the `break` statement gets executed and stops the execution of the loop further.

Code

PYTHON

Continue

Upcoming Earnings 1

Search

11:36 13-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=5b8e7625-f223-46e0-b16a-996a6132fbf2

NXT
WAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

In the below example, when the variable `i` value equals to `3` then the `break` statement gets executed and stops the execution of the loop further.

Code

PYTHON

```
1 - for i in range(5):
2 -     if i == 3:
3 -         break
4 -     print(i)
5 - print("END")
```

Output

```
0
1
2
END
```

Break in Nested Loop

The `break` in the inner loop stops the execution of the inner loop.

for item in Sequence A:
 for item in Sequence B:
 break

Code

Continue

Upcoming Earnings 1

Search

L

11:36 13-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=5b8e7625-f223-46e0-b16a-996a6132fbf2

NXT
wAVe™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

The `break` in the inner loop stops the execution of the inner loop.

for item in Sequence A:

```
for item in Sequence A:  
    for item in Sequence B:  
        break
```

Code

PYTHON

```
1 - for i in range(4):  
2 -     for j in range(4):  
3 -         if i * j > 10:  
4 -             break  
5 -         if (i > 0) and (j > 0):  
6 -             print(i * j)
```

Output

```
3  
6  
9
```

Continue

Ask doubt

Notes Discussions

Upcoming Earnings + NEW NOTE

11:36 ENG IN 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=5b8e7625-f223-46e0-b16a-996a6132fbf2

NXT
WAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

11. What is `continue` in loops?

The `continue` statement makes the program skip the remaining statements in the current iteration and begin the next iteration.

```
while condition A:
    [loop body]
    continue
[loop body]
```

Using `continue`

Generally, `continue` is used to skip the remaining statements in the current iteration when a condition is satisfied.

```
for item in Sequence:
    [loop body]
    if condition:
        continue
    [loop body]
```

In the below example, when the variable `i` value equals to `3` then the next statements in the loop body are skipped.

Code

```
1. for i in range(5):
2.     if i == 3:
```

PYTHON Continue

Miramal Tank Ro... Construction

Search

11:37 13-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=5b8e7625-f223-46e0-b16a-996a6132fbf2

NXT
wAVG™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

if condition:
continue

In the below example, when the variable `i` value equals to `3` then the next statements in the loop body are skipped.

Code

```
1 for i in range(5):
2     if i == 3:
3         continue
4     print(i)
5 print("END")
```

PYTHON

Output

```
0
1
2
4
END
```

Continue

Ask doubt

Notes Discussions

Miramal Tank Ro... Construction

Search

11:37 ENG IN 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=5b8e7625-f223-46e0-b16a-996a6132fbf2

NXT
WAVES™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

END

12. What is `pass` in Python?

The `pass` statement is a syntactic placeholder. When it is executed, nothing happens. Generally it is used when we have to test the code before writing the complete code.

```
if condition A:  
    Block 1  
elif condition B:  
    pass  
else:  
    Block 3
```

Empty Loops

We can use `pass` statements to test code written so far, before writing loop logic.

```
while condition A:    for item in Sequence:  
    pass            pass
```

Continue

Miramal Tank Ro... Construction

Search

11:37 13-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=5b8e7625-f223-46e0-b16a-996a6132fbf2

NXT
wAVe™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions (2 hours, 10 mins)

- Introduction to Python (25 mins)
- Conditional Statements and Loops (15 mins)
- List and Functions (25 mins)
- Tuples, Sets and Dictionaries (25 mins)
- OOPs (20 mins)
- Miscellaneous Topics (20 mins)
- More Interview Questions (20 mins)

Q. Write a Python program to print from 1 to 100.

PYTHON

```
1 - for i in range(1,101):
2     print(i)
```

Explanation:

The code uses a `for` loop to iterate over a range of numbers. `range(1, 101)` generates a sequence of numbers starting from 1 up to 101, but not including 101. So, the loop will iterate through the numbers 1 to 100.

Question 1 of 2

Which of the following generate the numbers 5 to 15 in Python?

- for i in range(5, 16):
- for i in range(1, 11):
- for i in range(5, 15):
- for i in range(15, 5):

Next

Complete the MCQs to proceed Continue

1 74°F Haze

Search

11:38 13-02-2024 ENG IN

Revolutionizing the Job Market

Q. Write a Python program that reverses a number using mathematical operations.

```
1 number = int(input())
2
3 reversed_number = 0
4
5 while number > 0:
6     digit = number % 10
7     reversed_number = (reversed_number * 10) + digit
8     number = number // 10
9
10 print(reversed_number)
```

PYTHON

Explanation:

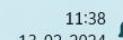
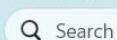
The Python code takes an integer number as input. The while loop repeatedly extracts the last digit of the original number, adds it to the reversed number, and removes it from the original number until all digits are processed.

Question 1 of 2

What is the purpose of the `digit = number % 10` line in the given Python code?

```
1 number = int(input())
2
3 reversed_number = 0
4
5 while number > 0:
6     digit = number % 10
7     reversed_number = (reversed_number * 10) + digit
```

PYTHON



Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=5b8e7625-f223-46e0-b16a-996a6132fbf2

NXT
wAVe™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. Write a program to print the following pattern:

- First Line: Odd Numbers
- Second Line: Even Numbers

Example

```
135  
246
```

Intent of the interviewer: The interviewer expects you to print the odd numbers on the first line and the even numbers on the second line, for numbers ranging from 1 to N.

PYTHON

```
1 number = int(input())  
2  
3 odd_numbers_string = ""  
4 even_numbers_string = ""  
5  
6 for num in range(1,number+1):  
7     if num % 2 != 0:  
8         odd_numbers_string += str(num)  
9     else:  
10        even_numbers_string += str(num)
```

Iterate through the range of numbers:

- If the number is odd, add it to odd_numbers_string
- If the number is even, add it to even_numbers_string

Continue

Ask doubt

74°F Haze

Search

11:39 13-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=5b8e7625-f223-46e0-b16a-996a6132fbf2

NXT
WAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPS Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. Write a program to print the following pattern:

- First Line: Odd Numbers
- Second Line: Even Numbers

Example

```
135
246
```

Intent of the interviewer: The interviewer expects you to print the odd numbers on the first line and the even numbers on the second line, for numbers ranging from 1 to N.

```
4 even_numbers_string = ""
5
6 for num in range(1,number+1):
7     if num % 2 != 0:
8         odd_numbers_string += str(num)
9     else:
10        even_numbers_string += str(num)
11
12 print(odd_numbers_string)
13 print(even_numbers_string)
```

PYTHON

Iterate through the range of numbers:

- If the number is odd, add it to odd_numbers_string
- If the number is even, add it to even_numbers_string

Ask doubt

Snipping Tool

Screenshot copied to clipboard and saved
Select here to mark up and share the image

74°F Haze

Search

11:40 13-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=5b8e7625-f223-46e0-b16a-996a6132fbf2

NXT
wAVe™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions (2 hours, 10 mins)

- Introduction to Python (25 mins)
- Conditional Statements and Loops (15 mins)
- List and Functions (25 mins)
- Tuples, Sets and Dictionaries (25 mins)
- OOPs (20 mins)
- Miscellaneous Topics (20 mins)
- More Interview Questions (20 mins)

Q. Write a program to print the following pattern.

```
1
1 3
1 3 5
```

Intent of the interviewer: The interviewer expects you to print the odd numbers as a right angled triangle of N rows

PYTHON

```
1 rows = int(input())
2
3 odd_number = 1
4 for i in range(1, rows + 1):
5     odd_number = 1
6     output = ""
7     for j in range(1, i + 1):
8         output += str(odd_number) + " "
9         odd_number += 2
10    print(output)
```

The outer loop `for i in range(1, rows + 1)` represents the range of rows, the inner loop `for j in range(1, i + 1)` iterates through the rows, prints the odd numbers, and then increments the odd number `odd_number += 2`.

Continue

74°F Haze

Search

L 11:40 13-02-2024 ENG IN

Revolutionizing the Job Market

1 3 5

Q. Write a program to print Right angled traingle with odd numbers.

Intent of the interviewer: The interviewer is expecting to take the number of rows as user input.

```
1 def print_triangle(n):
2     current_number = 1
3     for i in range(1, n + 1):
4         row_values = ""
5         for j in range(i):
6             row_values += str(current_number) + " "
7             current_number += 2
8         print(row_values)
9
10 rows = int(input())
```

Question 1 of 2

How many arguments should be given to the `print_triangle` function?

PYTHON

```
1 def print_triangle(n):
2     current_number = 1
3     for i in range(1, n + 1):
4         row_values = ""
5         for j in range(i):
```



11:41 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=5b8e7625-f223-46e0-b16a-996a6132fbf2

NXT
WAVES™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions (2 hours, 10 mins)

- Introduction to Python (25 mins)
- Conditional Statements and Loops (15 mins)
- List and Functions (25 mins)
- Tuples, Sets and Dictionaries (25 mins)
- OOPs (20 mins)
- Miscellaneous Topics (20 mins)
- More Interview Questions (20 mins)

Q. You have a list of numbers. For each number in the list, if it is divisible by 3 and 5, print 'FizzBuzz'. If it is divisible by 3, print 'Fizz'. If it is divisible by 5, print 'Buzz'. Otherwise, print the number itself.

PYTHON

```
1 number = int(input())
2
3 for num in range(1,number+1):
4     if num % 3 == 0 and num % 5 ==0:
5         print("FizzBuzz")
6     elif num % 3 == 0:
7         print("Fizz")
8     elif num % 5 == 0:
9         print("Buzz")
10    else:
```

Explanation:

Inside the loop, it checks the following conditions for each num:

- If the number is divisible by both 3 and 5, it prints FizzBuzz .
- If the number is divisible by 3 only, it prints Fizz .
- If the number is divisible by 5 only, it prints Buzz .
- If none of the above conditions are met, it prints the number itself.

Continue

Ask doubt

74°F Haze

Search

L

11:42 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=5b8e7625-f223-46e0-b16a-996a6132fbf2

NXT
wAVG™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. You have a list of numbers. For each number in the list, if it is divisible by 3 and 5, print 'FizzBuzz'. If it is divisible by 3, print 'Fizz'. If it is divisible by 5, print 'Buzz'. Otherwise, print the number itself.

PYTHON

```
2
3 - for num in range(1,number+1):
4 -     if num % 3 == 0 and num % 5 ==0:
5 -         print("FizzBuzz")
6 -     elif num % 3 == 0:
7 -         print("Fizz")
8 -     elif num % 5 == 0:
9 -         print("Buzz")
10 -    else:
11 -        print(num)
```

Explanation:

Inside the loop, it checks the following conditions for each num:

- If the number is divisible by both 3 and 5, it prints FizzBuzz .
- If the number is divisible by 3 only, it prints Fizz .
- If the number is divisible by 5 only, it prints Buzz .
- If none of the above conditions are met, it prints the number itself.

Ask doubt

Snipping Tool

Screenshot copied to clipboard and saved Select here to mark up and share the image

74°F Haze

Search

11:42 13-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT
WAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Recently Asked Questions and Answers

1. What are the Data Structures?

Data Structures allow us to store and organize data efficiently. This will allow us to easily access and perform operations on the data.

In Python, there are four built-in data structures

- List*
- Tuple*
- Set*
- Dictionary*

2. What is a list?

The List is the most versatile python data structure that holds an ordered sequence of items.

A diagram illustrating a list in Python. It shows a dashed rectangular box containing the elements "5", "Six", "2", and "8.2". Below the box, the indices 0, 1, 2, and 3 are shown in red, corresponding to the elements above them.

Creating a List

A List can be created by enclosing elements within [square] brackets where each item is separated by a comma.

Code

PYTHON

74°F Haze

Search

11:43 13-02-2024

Revolutionizing the Job Market - X

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74 Print Search Star Code Playground Jangipally Ranjith

**NXT
wAVE™**

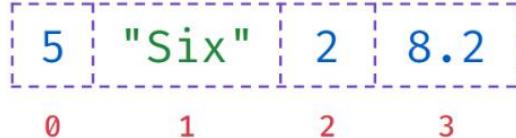
Interview Questions > List and Functions Reading Material

CONTENTS

- Interview Questions 2 hours, 10 mins
- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material** 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPS Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

2. What is a list?

The List is the most versatile python data structure that holds an ordered sequence of items.



Creating a List

A List can be created by enclosing elements within [square] brackets where each item is separated by a comma.

Code

```
1 a = 2
2 list_a = [5, "Six", a, 8.2]
3
4 print(type(list_a))
5 print(list_a)
```

Output

```
<class 'list'>
[5, 'Six', 2, 8.2]
```



Screenshot copied to clipboard and saved
Select here to mark up and share the image

74°F Haze ENG IN 11:44 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT
wAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Interview Questions > List and Functions Reading Material

3. Explain a few List Methods?

Append

Adds an element to the end of the list.

Syntax

```
list.append(value)
```

Code

```
1 list_a = []
2 for x in range(1, 4):
3     list_a.append(x)
4
5 print(list_a)
```

PYTHON

Output

```
[1, 2, 3]
```

Extend

Adds all the elements of a sequence to the end of the list.

Syntax

```
list_a.extend(list_b)
```

Code

```
list_a = [1, 2, 3]
list_b = [4, 5, 6]
list_a.extend(list_b)
print(list_a)
```

PYTHON

74°F Haze

Search

11:44 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT
wAVe™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Interview Questions > List and Functions Reading Material

Extend

Adds all the elements of a sequence to the end of the list.

Syntax

```
list_a.extend(list_b)
```

Code

```
1 list_a = [1, 2, 3]
2 list_b = [4, 5, 6]
3 list_a.extend(list_b)
4
5 print(list_a)
```

PYTHON

Output

```
[1, 2, 3, 4, 5, 6]
```

Insert

Inserts the given element at the specified index.

Syntax

```
list.insert(index, value)
```

Code

```
1 list_a = [1, 2, 3]
2 list_a.insert(1, 4)
```

PYTHON

74°F Haze

Search

L

11:44 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT
wAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Interview Questions > List and Functions Reading Material

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

Insert

Inserts the given element at the specified index.

Syntax

```
list.insert(index, value)
```

Code

```
1 list_a = [1, 2, 3]
2 list_a.insert(1, 4)
3
4 print(list_a)
```

PYTHON

Output

[1, 4, 2, 3]

Pop

Removes the last item from the list.

Syntax

```
list.pop()
```

Code

```
1 list_a = [1, 2, 3]
```

Snipping Tool

Screenshot copied to clipboard and saved
Select here to mark up and share the image

74°F Haze

Search

L

11:44 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT
WAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Interview Questions > List and Functions Reading Material

Pop

Removes the last item from the list.

Syntax

```
list.pop()
```

Code

```
1 list_a = [1, 2, 3]
2 list_a.pop()
3
4 print(list_a)
```

PYTHON

Output

```
[1, 2]
```

Remove

Removes the first matching element from the list.

Syntax

```
list.remove(value)
```

Code

```
1 list_a = [1, 3, 2, 3]
2 list_a.remove(3)
3
```

Snipping Tool

Screenshot copied to clipboard and saved
Select here to mark up and share the image

74°F Haze

Search

L

11:44 ENG IN 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT
WAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Interview Questions > List and Functions Reading Material

Remove

Removes the first matching element from the list.

Syntax

```
list.remove(value)
```

Code

```
1 list_a = [1, 3, 2, 3]
2 list_a.remove(3)
3
4 print(list_a)
```

PYTHON

Output

```
[1, 2, 3]
```

Clear

Removes all the items from the list.

Syntax

```
list.clear()
```

Code

```
1 list_a = [1, 2, 3]
2 list_a.clear()
3
```

Snipping Tool

Screenshot copied to clipboard and saved
Select here to mark up and share the image

74°F Haze

Search

L

11:45 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT
wAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Output

```
[1, 2, 3]
```

Clear

Removes all the items from the list.

Syntax

```
list.clear()
```

Code

```
1 list_a = [1, 2, 3]
2 list_a.clear()
3
4 print(list_a)
```

PYTHON

Output

```
[]
```

Index

Returns the index of the first occurrence of the specified item in the list.

Syntax

```
list.index(item)
```

74°F Haze

Search

L

11:45 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT
wAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Interview Questions > List and Functions Reading Material

Index

Returns the index of the first occurrence of the specified item in the list.

Syntax

```
list.index(item)
```

Code

```
1 list_a = [1, 3, 2, 3]
2 index = list_a.index(3)
3
4 print(index)
```

PYTHON

Output

```
1
```

Count

Returns the number of elements with the specified value.

Syntax

```
list.count(value)
```

74°F Haze

Search

L 11:45 13-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT WAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Interview Questions > List and Functions Reading Material

1

Count

Returns the number of elements with the specified value.

Syntax

```
list.count(value)
```

Code

```
1 list_a = [1, 2, 3]
2 count = list_a.count(2)
3
4 print(count)
```

PYTHON

Output

```
1
```

Sort

Sorts the items of a list in ascending order. The `sort()` method modifies the original list.

Syntax

```
list.sort()
```

Code

Snipping Tool

Screenshot copied to clipboard and saved
Select here to mark up and share the image

74°F Haze

Search

11:45 13-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT
wAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Interview Questions > List and Functions Reading Material

1

Sort

Sorts the items of a list in ascending order. The `sort()` method modifies the original list.

Syntax

```
list.sort()
```

Code

```
1 list_a = [1, 3, 2]
2 list_a.sort()
3
4 print(list_a)
```

PYTHON

Output

```
[1, 2, 3]
```

Sorted

Sorts the items of a list in ascending order. The `sorted()` method returns the modified list.

Syntax

```
sorted()
```

Code

Snipping Tool

Screenshot copied to clipboard and saved
Select here to mark up and share the image

74°F Haze

Search

11:45 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT
wAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Interview Questions > List and Functions Reading Material

Sorted

Sorts the items of a list in ascending order. The `sorted()` method returns the modified list.

Syntax

```
sorted()
```

Code

```
list_a = [1, 3, 2]
list_b = sorted(list_a)
print(list_b)
```

PYTHON

Output

```
[1, 2, 3]
```

Code

```
list_a = [1, 3, 2]
sorted(list_a)
print(list_a)
```

PYTHON

Output

```
[1, 3, 2]
```

74°F Haze

Search

Icons: File Explorer, Google Chrome, Microsoft Edge, Task View, LinkedIn, WhatsApp, Spotify, Xbox, Microsoft Store

Eng IN 11:45 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT
wAVG™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions (2 hours, 10 mins)

- Introduction to Python Reading Material (25 mins)
- Conditional Statements and Loops Reading Material (15 mins)
- List and Functions Reading Material (25 mins)
- Tuples, Sets and Dictionaries Reading Material (25 mins)
- OOPs Reading Material (20 mins)
- Miscellaneous Topics Reading Material (20 mins)
- More Interview Questions (20 mins)

Q. What is the difference between Range and List?

Range	List
range is a built-in function in Python that generates a sequence of numbers.	A list is a data structure in Python that can hold an ordered collection of items.
It is commonly used for iterating over a sequence of numbers in a loop.	Lists can store elements of different data types, and they are mutable.
More memory-efficient for large ranges.	Consumes more memory, especially for large lists.
Syntax: range(start, stop, step)	Syntax: [element1, element2, ...]

← Question 1 of 4

What is the primary task of the range function in Python?

- Storing a collection of items
- Generating a sequence of numbers
- Sorting a list
- Modifying list elements

Next

Complete the MCQs to proceed Continue

1 74°F Haze

Search

11:46 13-02-2024 ENG IN



Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT
wAVe™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions (2 hours, 10 mins)

- Introduction to Python Reading Material (25 mins)
- Conditional Statements and Loops Reading Material (15 mins)
- List and Functions Reading Material (25 mins)
- Tuples, Sets and Dictionaries Reading Material (25 mins)
- OOPs Reading Material (20 mins)
- Miscellaneous Topics Reading Material (20 mins)
- More Interview Questions (20 mins)

Q. With which data types can we use the pop() method in Python?

The `pop()` method can be used with three different data types in Python: lists, dictionaries, and sets.

- Lists:** The `pop()` method removes the specified item from a list. If no index is specified, the last element is removed.
- Dictionaries:** The `pop()` method removes the specified item from a dictionary. The method returns the value of the removed item.
- Sets:** The `pop()` method removes a random item from a set. The method returns the removed item.

► More Information

Question 1 of 4

What does the `pop()` method return when used with sets?

- The removed item's index
- The removed item
- The set itself
- An error message

Next

Snipping Tool

Screenshot copied to clipboard and saved
Select here to mark up and share the image

74°F Haze

Search

L

11:46 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT
wAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions (2 hours, 10 mins)

- Introduction to Python Reading Material (25 mins)
- Conditional Statements and Loops Reading Material (15 mins)
- List and Functions Reading Material (25 mins)
- Tuples, Sets and Dictionaries Reading Material (25 mins)
- OOPs Reading Material (20 mins)
- Miscellaneous Topics Reading Material (20 mins)
- More Interview Questions (20 mins)

Q. Explain List extends in Python?

Intent of the interviewer:

Extending a list in Python means adding one or more elements to the end of the list. There are several ways to extend a list in Python, including:

- Using the `extend()` method: Adds all the elements of an iterable to the end of the list.
- Using the `+` operator: Concatenate two lists together.
- Using `list slicing`: Insert a new element or list of elements into the end of a list.

More Information

Using the `extend()` method:

```
1 list1 = [1, 2, 3]
2 list2 = [4, 5, 6]
3 list1.extend(list2)
```

Using the `+` operator:

```
1 list3 = [1, 2, 3]
2 list4 = [4, 5, 6]
3 list5 = list3 + list4
```

Using `list slicing`:

```
1 list6 = [1, 2, 3]
2 new_element = 4
3 list6[len(list6):] = [new_element]
```

74°F Haze

Search

Continue

11:47 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT
wAVe™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions (2 hours, 10 mins)

- Introduction to Python Reading Material (25 mins)
- Conditional Statements and Loops Reading Material (15 mins)
- List and Functions Reading Material (25 mins)
- Tuples, Sets and Dictionaries Reading Material (25 mins)
- OOPs Reading Material (20 mins)
- Miscellaneous Topics Reading Material (20 mins)
- More Interview Questions (20 mins)

Q. What will be the output of this code?

PYTHON

```
1 temp=[3,4]
2 lst=[1,2]
3
4 lst.append(temp) # Output : [3, 4, [1, 2]]
5 lst.extend(temp) # Output : [3, 4, [1, 2], 1, 2]
```

Explanation:

The `append()` method is used to add an entire list to another list. In this case, `temp` is appended to `lst`, resulting in `lst` containing 1, 2, and the entire `temp` list.

The `extend()` method is used to add the elements of one list to the end of another list. In this case, the elements of `temp` are added to the end of `lst`. Therefore, `lst` will contain the following elements: 1, 2, [3, 4], 3, 4.

Question 1 of 4

What will be the output after executing the following code?

PYTHON

```
1 temp = [3, 4]
2 lst = [1, 2]
3
4 lst.extend(temp)
5 print(lst)
```

[3, 4, [1, 2]]
[3, 4, [1, 2], 1, 2]
[3, 4, 1, 2]
[1, 2, 3, 4]

Complete the MCQs to proceed Continue

1 74°F Haze

Search

L

11:48 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT
WAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Interview Questions > List and Functions Reading Material

Q. Can you create a Python list and add a new element?

Yes, I can create a Python list and add a new element.

To create python list we use square brackets.

```
1 my_list = [1,2,3]
```

PYTHON

There are four methods to add elements to a List in Python.

- **append():** append the element to the end of the list.
- **insert():** This method inserts the element at a specific position in the list.
- **extend():** Adds all the elements of an iterable to the end of the list.
- **List Concatenation:** Concatenate two lists together.

▼ More Information

Using the `append()` method:

```
1 my_list = [1,2,3]
2 my_list.append(4)
3 print(my_list) #Output: [1, 2, 3, 4]
```

PYTHON

Using the `insert()` method:

```
1 my_list = [1,2,3]
2 my_list.insert(2,0)
3 print(my_list) #Output: [1, 2, 0, 3]
```

PYTHON

74°F Haze

Search

11:48 13-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT
WAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Interview Questions > List and Functions Reading Material

More Information

Using the `append()` method:

```
my_list = [1,2,3]
my_list.append(4)
print(my_list) #Output: [1, 2, 3, 4]
```

Using the `insert()` method:

```
my_list = [1,2,3]
my_list.insert(2,0)
print(my_list) #Output: [1, 2, 0, 3]
```

Using the `extend()` method:

```
my_list = [1, 2, 3]
new_list = [4, 5, 6]
my_list.extend(new_list)
print(my_list) #Output: [1, 2, 3, 4, 5, 6]
```

Using the `+` operator:

```
my_list = [1, 2, 3]
new_list = [4, 5, 6]
concatenate_list = my_list + new_list
print(concatenate_list) #Output: [1, 2, 3, 4, 5, 6]
```

Snipping Tool

Screenshot copied to clipboard and saved
Select here to mark up and share the image

74°F Haze

Search

L

11:48 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT
wAVe™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions (2 hours, 10 mins)

- Introduction to Python Reading Material (25 mins)
- Conditional Statements and Loops Reading Material (15 mins)
- List and Functions Reading Material (25 mins)
- Tuples, Sets and Dictionaries Reading Material (25 mins)
- OOPs Reading Material (20 mins)
- Miscellaneous Topics Reading Material (20 mins)
- More Interview Questions (20 mins)

Q. What is the difference between the `append()` and `extend()` methods for lists in Python?

When I use `append()`, it adds a single element to the end of the list. This element can be of any data type, and it's added as a single item. So, if I append another list to an existing list, the entire list will be added as a single element.

On the other hand, `extend()` is used to merge two lists or add multiple elements. When I use `extend()` with a list as an argument, each element of the list argument is added individually to the end of the original list.

▼ More Information

Here's an example to explain the difference:

Let's say we have a list `a = [1, 2, 3]`.

If I use `append()` to add another list, for example `b = [4, 5, 6]`, the result would be `a = [1, 2, 3, [4, 5, 6]]`. Notice that the list `b` is added as a single element.

However, if I use `extend()` with the same list `b`, the result would be `a = [1, 2, 3, 4, 5, 6]`. Here, each element of list `b` is added individually to `a`.

Continue

74°F Haze

Search

11:50 13-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT
wAVe™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions (2 hours, 10 mins)

- Introduction to Python Reading Material (25 mins)
- Conditional Statements and Loops Reading Material (15 mins)
- List and Functions Reading Material (25 mins)
- Tuples, Sets and Dictionaries Reading Material (25 mins)
- OOPs Reading Material (20 mins)
- Miscellaneous Topics Reading Material (20 mins)
- More Interview Questions (20 mins)

insert()

5. How to use the `split()` method?

The `split()` splits a string into a list at every specified separator.

Syntax:

```
str_var.split(separator)
```

Example

Code

```
1 nums = "1 2 3 4"
2 num_list = nums.split()
3 print(num_list)
```

PYTHON

Output

```
[‘1’, ‘2’, ‘3’, ‘4’]
```

If no separator is specified, the default separator is whitespace.

Using separator

Example

Code

Continue

74°F Haze

Search

L 11:51 13-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT
wAVG™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions (2 hours, 10 mins)

- Introduction to Python Reading Material (25 mins)
- Conditional Statements and Loops Reading Material (15 mins)
- List and Functions Reading Material (25 mins)
- Tuples, Sets and Dictionaries Reading Material (25 mins)
- OOPs Reading Material (20 mins)
- Miscellaneous Topics Reading Material (20 mins)
- More Interview Questions (20 mins)

Output

```
[1, 2, 3, 4]
```

If no separator is specified, the default separator is whitespace.

Using separator

Example

Code

```
1 nums = "1,2,3,4"
2 num_list = nums.split(',')
3 print(num_list)
```

PYTHON

Output

```
[1, 2, 3, 4]
```

Continue

Ask doubt

Notes Discussions

74°F Haze

Search

L

11:51 13-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT
wAVG™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

6. How to use the `join()` method?

The `join()` takes all the items in a sequence of strings and joins them into one string.

Syntax:

```
str.join(sequence)
```

Example

Code

```
list_a = ['Python is ', ' program', ' a', ' language']  
string_a = "-".join(list_a)  
print(string_a)
```

PYTHON

Output

```
Python is a programming language
```

7. What is List Slicing?

Obtaining a part of a list is called List Slicing.

Syntax:

```
variable_name[start_index:end_index]
```

Continue

74°F Haze

Search

L

11:52 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT
wAVG™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

7. What is List Slicing?

Obtaining a part of a list is called List Slicing.

Syntax:

```
variable_name[start_index:end_index]
```

- end_index is not included in the slice.

list_a

Code

```
1 list_a = [5, "Six", 2, 8.2]
2 list_b = list_a[:2]
3 print(list_b)
```

PYTHON

Output

```
[5, 'Six']
```

Extended Slicing

Continue

74°F Haze

Search

L

11:52 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT
wAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

2 list_b = list_a[::2]
3 print(list_b)

Output

```
[5, 'Six']
```

Extended Slicing

Similar to string extended slicing, we can extract alternate items from the list using the step.

Syntax:

```
variable[start_index:end_index:step]
```

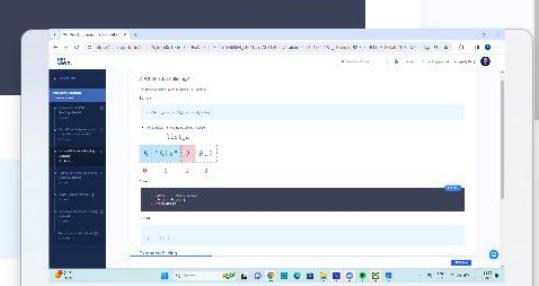
Code

```
1 list_a = ["R", "B", "G", "O", "W"]  
2 list_b = list_a[0:5:3]  
3 print(list_b)
```

PYTHON

Output

```
['R', 'O']
```



Snipping Tool

Screenshot copied to clipboard and saved
Select here to mark up and share the image

74°F Haze

Search

11:52 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT
wAVe™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions (2 hours, 10 mins)

- Introduction to Python Reading Material (25 mins)
- Conditional Statements and Loops Reading Material (15 mins)
- List and Functions Reading Material (25 mins)
- Tuples, Sets and Dictionaries Reading Material (25 mins)
- OOPs Reading Material (20 mins)
- Miscellaneous Topics Reading Material (20 mins)
- More Interview Questions (20 mins)

Q. What is Slicing in python?

Slicing in Python refers to the technique of extracting a portion of a sequence, such as a list, tuple, or string. It allows you to create a new sequence by specifying a range of indices.

Syntax:

```
1 sequence[start:stop:step]
```

PYTHON

More Information

The range() function can be used in three different ways:

- slice(start, stop):** Returns a sequence of elements starting from the start value up to, but not including, the stop value. For instance, slice(1, 5) will create a slice from index 1 to 4.
- slice(stop):** Returns a sequence of elements from the starting of the sequence up to, but not including, the stop value. For example, slice(5) will create a slice from index 0 to 4.
- slice(start, stop, step):** Returns a sequence of elements starting from the start value, up to but not including the stop value, incrementing by the step value. The step value defines the spacing between each index in the slice. For example, slice(1, 10, 2) will create a slice with indexes 1, 3, 5, 7, 9.

Note

If you don't provide any parameters to the slice function (iterable[:]), it will create a shallow copy of the original sequence.

Question 1 of 4

Which statement is correct about slicing in Python?

- Slicing always creates a deep copy of the original sequence.
- The ending index in a slice is inclusive.
- The starting index in a slice is inclusive.

Complete the MCQs to proceed

Continue

74°F Haze

Search

11:53 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT wAVG™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

8. What is Negative Indexing?

Using a negative index returns the nth item from the end of the list.

The last item in the list can be accessed with the index -1

list_a[-1]

Accessing List Items with Negative Index

Example-1

Code

```
1 list_a = [5, 4, 3, 2, 1]
2 item = list_a[-1]
3 print(item)
```

PYTHON

Output

1

Continue

74°F Haze

Search

11:53 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT
wAVe™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPS Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Example-2

Code

```
list_a = [5, 4, 3, 2, 1]
item = list_a[-4]
print(item)
```

PYTHON

Output

```
4
```

Slicing With Negative Index

You can also specify negative indices while slicing a List.

Code

```
list_a = [5, 4, 3, 2, 1]
list_b = list_a[-3:-1]
print(list_b)
```

Output

```
[3, 2]
```

list_a[-3:-1]

Snipping Tool

Screenshot copied to clipboard and saved
Select here to mark up and share the image

74°F Haze

Search

11:53 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT
wAVG™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

list_a[-3:-1]

Out of Bounds Index

While slicing, the index can go beyond the size of the list.

Code

PYTHON

```
1 list_a = [5, 4, 3, 2, 1]
2 list_b = list_a[-6:-2]
3 print(list_b)
```

Output

[5, 4, 3]

Continue

1 74°F Haze

Search

Chrome Edge File L Microsoft Store Spotify Xbox

ENG IN 11:54 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT
wAVG™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

9. How to reverse a List?

Reversing a list using `reverse()` method:

The `reverse()` method can be used to reverse a List. It updates the original list.

Code

PYTHON

```
1 week_days = ['Monday', 'Tuesday', 'Wednesday']
2 week_days.reverse()
3
4 print(week_days)
```

Output

```
['Wednesday', 'Tuesday', 'Monday']
```

Reversing a list using list slicing:

A list can be reversed using extended slicing.

Syntax:

```
variable[start:end:negative_step]
```

-1 for step will reverse the order of items in the list.

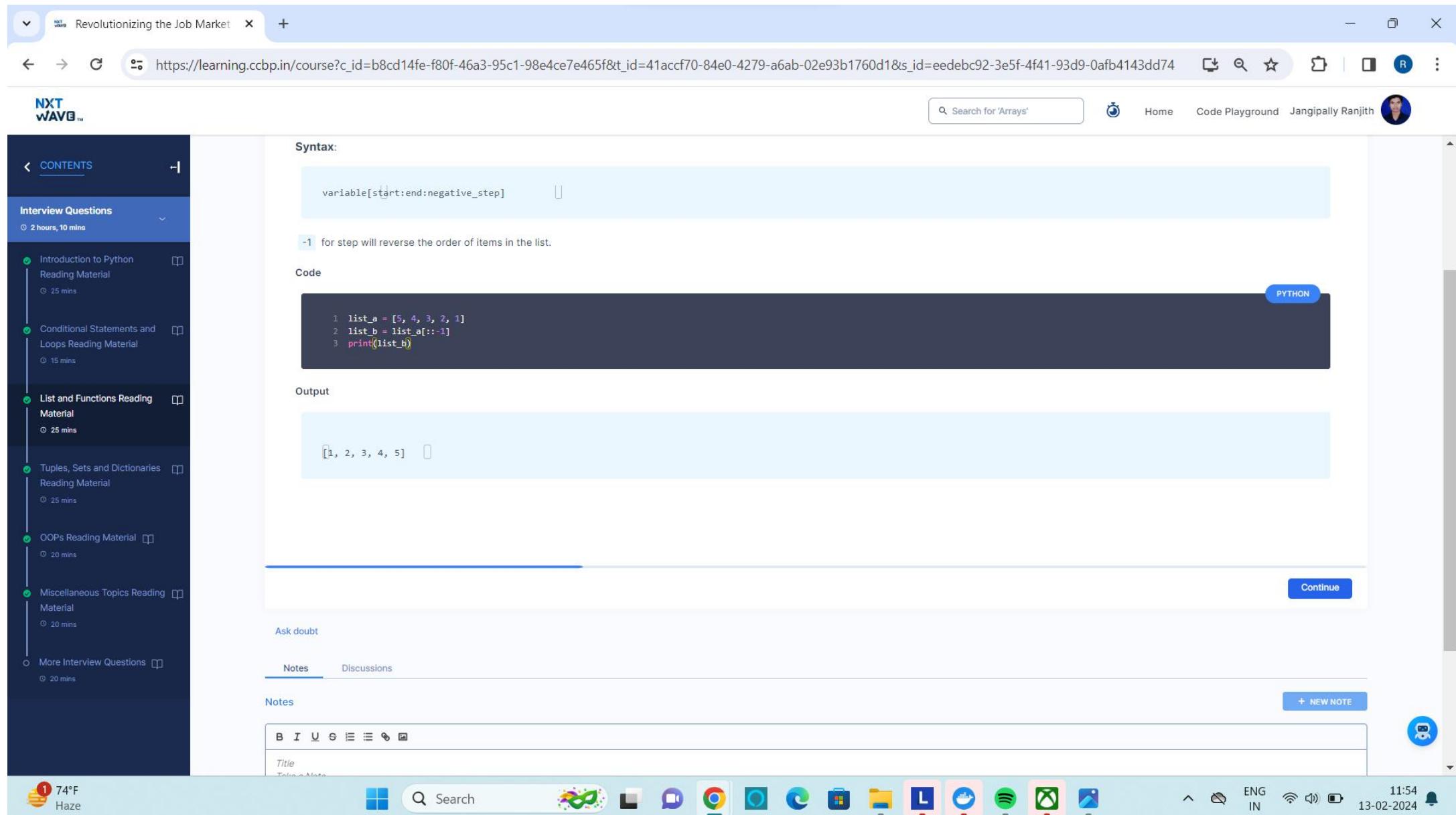
Code

Continue

74°F Haze

Search

L 11:54 13-02-2024 ENG IN



Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT
WAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. What is the difference between list and array?

Lists and arrays are data structures that store multiple items, but have different features and uses.

List	Array
Can hold items of multiple data types.	Holds items of the same data type.
Cannot handle arithmetic operations.	Can handle arithmetic operations.
Consumes more memory.	Consumes less memory.
All elements can be accessed directly.	A loop is needed to access elements.
Better suited for shorter sequences of data.	More efficient for longer sequences of data.

10. Explain about functions in Python?

A function is a block of reusable code to perform a specific action. Functions help us in using existing code without writing it every time we need it.

A function can be defined using a keyword `def`. A function is uniquely identified by the `function_name`.

`def function_name():`

Reusable Block of Code

Code

```
1 + def greet():
```

PYTHON Continue

74°F Haze

Search

13-02-2024 11:55 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT
wAVe™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

10. Explain about functions in Python?

A function is a block of reusable code to perform a specific action. Functions help us in using existing code without writing it every time we need it.

A function can be defined using a keyword `def`. A function is uniquely identified by the `function_name`.

def function_name():

Reusable Block of Code

Code

```
1 - def greet():
2     print("Hello")
3
4 greet()
5 greet()
```

PYTHON

Output

```
Hello
Hello
```

Snipping Tool

Screenshot copied to clipboard and saved
Select here to mark up and share the image

81°F Haze

Search

L

11:55 IN

13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT
wAVe™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

We can pass values to a function using Argument.

def function_name(args):

Reusable Block of Code

Code

PYTHON

```
1 - def greet(word):
2     msg = "Hello " + word
3     print(msg)
4
5 name1 = input()
6 name2 = input()
7 greet(word = name1)
8 greet(word = name2)
```

Input

Teja
Rahul

Output

Hello Teja
Hello Rahul

Continue

81°F Haze

Search

Icons for various applications: File Explorer, Task View, Edge, Google Chrome, Microsoft Edge, File Manager, LinkedIn, WhatsApp, Spotify, Xbox, and others.

ENG IN 11:55 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT
wAVe™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Providing default values

Default values indicate that the function argument will take that value if no argument value is passed during the function call.

Example

Code

PYTHON

```
1 def greet(arg_1 = "Hi", arg_2 = "Ram"):
2     print(arg_1 + " " + arg_2)
3
4 greeting = input()
5 name = input()
6 greet()
```

Input

Hello
Teja

Output

Hi Ram

Continue

Breaking news Unfolding now

Search

11:55 13-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT
wAVe™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Hello Kanui

Providing default values

Default values indicate that the function argument will take that value if no argument value is passed during the function call.

Example

Code

```
1 def greet(arg_1 = "Hi", arg_2 = "Ram"):
2     print(arg_1 + " " + arg_2)
3
4 greeting = input()
5 name = input()
6 greet()
```

PYTHON

Input

```
Hello
Teja
```

Output

```
Hi Ram
```

Continue

81°F Haze

Search

12:01 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions (2 hours, 10 mins)

- Introduction to Python (25 mins)
- Conditional Statements and Loops (15 mins)
- List and Functions (25 mins)
- Tuples, Sets and Dictionaries (25 mins)
- OOPs (20 mins)
- Miscellaneous Topics (20 mins)
- More Interview Questions (20 mins)

11. What is Recursion?

A function calling itself is called **Recursion**

```
def function_1():
```

Let's understand recursion with a simple example of multiplying N numbers

Multiply N Numbers

```
1: def factorial(n): # Recursive Function
2:     if n == 1: # Base Case
3:         return 1
4:     return n * Factorial(n - 1) # Recursion
5: num = int(input())
6: result = factorial(num)
7: print(result)
```

PYTHON

Input:

5

Output:

120

Continue

81°F Haze

Search

Icons for various applications: File Explorer, Task View, Edge, Google Chrome, Microsoft Edge, File Manager, LinkedIn, Spotify, Xbox, and others.

ENG IN 12:02 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT
wAVe™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Interview Questions > List and Functions Reading Material

Q. What is a Lambda function and what is the main purpose of it?

Companies Asked: Translytics Business Services, Xcube Labs

A lambda function is an anonymous function used for doing simple operations. Lambda functions can have any number of arguments, but can only have one expression.

The primary purpose of a lambda function is to perform simple operations without the need to formally define a function using the typical `def` keyword. They are often used for short, operations that are only needed temporarily without the need of a full function definition.

Here's a breakdown of the lambda function:

Syntax: The general syntax of a lambda function in Python is:

```
lambda arguments: expression
```

The expression is executed and returned when the lambda function is called.

Examples:

- A lambda function that adds 10 to any number you provide:

```
f = lambda x: x + 10
print(f(5)) # Outputs: 15
```

- A lambda function that multiplies two numbers:

```
mul = lambda x, y: x * y
print(mul(3, 7)) # Outputs: 21
```

Use cases:

- Lambda functions are particularly useful when you need a simple function for a short period and don't want to formally define it.

81°F Haze

Search

Icons for various applications: File Explorer, Task View, Edge, Google Chrome, Microsoft Edge, File Manager, LinkedIn, WhatsApp, Spotify, Xbox, and others.

ENG IN 12:02 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT
wAVe™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions (2 hours, 10 mins)

- Introduction to Python Reading Material (25 mins)
- Conditional Statements and Loops Reading Material (15 mins)
- List and Functions Reading Material (25 mins)
- Tuples, Sets and Dictionaries Reading Material (25 mins)
- OOPs Reading Material (20 mins)
- Miscellaneous Topics Reading Material (20 mins)
- More Interview Questions (20 mins)

The expression is executed and returned when the lambda function is called.

Examples:

- A lambda function that adds 10 to any number you provide:

```
1 f = lambda x: x + 10
2 print(f(5)) # Outputs: 15
```

Use cases:

- Lambda functions are particularly useful when you need a simple function for a short period and don't want to formally define it.
- They're commonly used with functions like `map()`, `filter()`, and `sorted()` in Python.

```
1 # Using lambda with sorted() to sort a list of tuples by the second item
2 tuple_list = [(1, 2), (3, 1), (5, 10), (0, 6)]
3 sorted_list = sorted(tuple_list, key=lambda x: x[1])
4 print(sorted_list) # Outputs: [(3, 1), (1, 2), (0, 6), (5, 10)]
```

Continue

Ask doubt

Notes Discussions

81°F Haze

Search

Chrome Edge Microsoft Store Spotify LinkedIn Xbox

ENG IN 12:02 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT
wAVe™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. What are generators and what are the uses of it?

Companies Asked: Translytics Business Services

A Python generator is a special function that returns an iterable set of items, one at a time. It is created using `yield` instead of `return`. Instead of running to completion and returning a single value, it produces a stream of values. Its execution can be paused after yielding each value and later resumed. This makes it perfect for handling large data sets.

Generators are defined like a normal function but with the `yield` keyword replacing the `return` keyword.

Example:

Here's a Python generator function that produces numbers in the Fibonacci sequence:

```
1 def fibonacci_generator(n):
2     a, b = 0, 1
3     for _ in range(n):
4         yield a
5         a, b = b, a + b
6
7 # Create a generator object
8 gen = fibonacci_generator(5)
9 # Iterate over the generator
10 for number in gen:
```

PYTHON

Outputs:

```
1 0
2 1
3 1
4 2
5 3
```

Uses:

- Memory Efficiency:** Ideal for producing large sequences without storing them entirely in memory.
- Large Data Streams:** For reading large files or database results in chunks, rather than loading everything at once.

Continue

81°F Haze

Search

L 12:03 13-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT
WAVES™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions (2 hours, 10 mins)

- Introduction to Python Reading Material (25 mins)
- Conditional Statements and Loops Reading Material (15 mins)
- List and Functions Reading Material (25 mins)
- Tuples, Sets and Dictionaries Reading Material (25 mins)
- OOPs Reading Material (20 mins)
- Miscellaneous Topics Reading Material (20 mins)
- More Interview Questions (20 mins)

Q. Write code for the Fibonacci series.

Companies Asked: Translytics Business Services

There are multiple ways to solve the Fibonacci series problem:

1. Using Recursion (Most expected in interviews)

PYTHON

```
1 def fibonacci_recursive(n):
2     if n == 0:
3         return 0
4     elif n == 1:
5         return 1
6     else:
7         return fibonacci_recursive(n - 1) + fibonacci_recursive(n - 2)
8
9 for i in range(10):
10    print(fibonacci_recursive(i), end=' ')
```

After executing the above code, the output will be 0 1 1 2 3 5 8 13 21 34 .

Continue

81°F Haze

Search

L 12:04 13-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT
wAVG™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

After executing the above code, the output will be 0 1 1 2 3 5 8 13 21 34 .

← Question 2 of 4

```
1 - def fibonacci_recursive(n):
2 -     if n == 0:
3 -         return 0
4 -     elif n == 1:
5 -         return 1
6 -     else:
7 -         return fibonacci_recursive(n - 1) + fibonacci_r
```

0 1 n - 1 n - 2

Next

Complete the MCQs to proceed Continue

81°F Haze

Search

12:04 13-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT
wAVe™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Question 4 of 4

In the recursive version of the Fibonacci series code, what will be the return value for the input `n = 2` ?

PYTHON

```
1 def fibonacci_recursive(n):
2     if n == 0:
3         return 0
4     elif n == 1:
5         return 1
6     else:
7         return fibonacci_recursive(n - 1) + fibonacci_recursive(n - 2)
```

0 1 2 3

Continue

81°F Haze

Search

12:05 13-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT
wAVe™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. Write code for the factorial of a number.

Recursive Approach:

```
2 - if n < 0:
3 -     print("n must be non-negative")
4 - elif n == 0:
5 -     return 1
6 - else:
7 -     return n * factorial_recursive(n - 1)
8 -
9 - n = 5
10 result = factorial_recursive(n)
11 print(result)
```

The `factorial_recursive` function calculates the factorial of a non-negative integer `n` by multiplying `n` by the factorial of `n-1` (previous element), until `n` reaches 0.

▼ Another Approach:

Iterative approach:

```
1 - def factorial_iterative(n):
2 -     if n < 0:
3 -         print("n must be non-negative")
4 -     result = 1
5 -     for i in range(1, n + 1):
6 -         result *= i
7 -     return result
8 -
9 - n = 5
10 result = factorial_iterative(n)
```

The iterative function `factorial_iterative` calculates the factorial of a non-negative integer `n`. It checks for `n` non-negativity, initializes `result` as 1, and iteratively multiplies `result` by numbers from 1 to `n`.

Continue

81°F Haze

Search

L 12:05 13-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT
wAVe™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

8
9 n = 5
10 result = factorial_recursive(n)
11 print(result)

The `factorial_recursive` function calculates the factorial of a non-negative integer `n` by multiplying `n` by the factorial of `n-1` (previous element), until `n` reaches 0.

▼ Another Approach:
Iterative approach:

PYTHON

```
1- def factorial_iterative(n):
2-     if n < 0:
3-         print("n must be non-negative")
4-     result = 1
5-     for i in range(1, n + 1):
6-         result *= i
7-     return result
8-
9 n = 5
10 result = factorial_iterative(n)
```

The iterative function `factorial_iterative` calculates the factorial of a non-negative integer `n`. It checks for `n` non-negativity, initializes `result` as 1, and iteratively multiplies `result` by numbers from 1 to `n`.

Ask doubt

Notes Discussions

Notes + NEW NOTE

81°F Haze

Search

12:06 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT
wAVG™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

The iterative function `factorial_iterative` calculates the factorial of a non-negative integer `n`. It checks for `n` non-negativity, initializes result as 1, and iteratively multiplies result by numbers from 1 to `n`.

Question 1 of 3

```
1 def factorial_iterative(n):
2     if n == 0:
3         return 1
4     else:
5         return n * factorial_iterative(n - 1)
6
7 n = 5
8 result = factorial_iterative(n)
9 print(result)
```

It calculates the sum of numbers from 1 to `n`.

It calculates the factorial of a non-negative integer `n`.

It checks if `n` is an odd or even number.

It calculates the average of `n` numbers.

Next

Complete the MCQs to proceed Continue

81°F Haze

Search

12:06 13-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT
wAVG™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

The iterative function `factorial_iterative` calculates the factorial of a non-negative integer `n`. It checks for `n` non-negativity, initializes result as 1, and iteratively multiplies result by numbers from 1 to `n`.

Question 2 of 3

```
1 - def factorial_recursive(n):  
2 -     if n == 0:  
3 -         return 1  
4 -     else:  
5 -         return n * factorial_recursive(n - 1)  
6 -  
7 - n = 5  
8 - result = factorial_recursive(n)  
9 - print(result)
```

It checks if `n` is an even number.

It checks if `n` is an odd number.

It serves as the base case for the recursion, terminating the recursion when `n` is 0.

It calculates the sum of all numbers up to `n`.

Next

Complete the MCQs to proceed Continue

81°F Haze

Search

12:06 13-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT
wAVe™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. Write a program to find multiplication of digits in a number?

PYTHON

```
1 def multiply_digits(num):
2     result = 1
3
4     for char in num:
5         digit = int(char)
6
7         if digit == 0:
8             return 0
9
10        result *= digit
```

▼ More Information

Explanation:

1. Iteration Through Digits:

PYTHON

```
1 def multiply_digits(num):
2     result = 1
3
4     for char in num:
```

The code defines a function `multiply_digits` that takes a parameter `num`. Inside the function, there's a for loop that iterates through each character(`char`) in the input `num`.

2. Checking for Zero:

PYTHON

```
1 if result == 0:
2     return 0
3
4 result
```

Continue

Ask doubt

Notes Discussions

81°F Haze

Search

12:07 13-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT
WAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. Write a program to find multiplication of digits in a number?

PYTHON

```
5     digit = int(char)
6
7     if digit == 0:
8         return 0
9
10    result *= digit
11
12    return result
13
14 user_input = input()
15 print(multiply_digits(user_input))
```

▼ More Information

Explanation:

1. Iteration Through Digits:

PYTHON

```
1 def multiply_digits(num):
2     result = 1
3
4     for char in num:
```

The code defines a function `multiply_digits` that takes a parameter `num`. Inside the function, there's a `for` loop that iterates through each character(`char`) in the input `num`.

2. Checking for Zero:

```
1     if digit == 0:
2         return 0
```

Ask doubt

Notes Discussions

81°F Haze

Search

Snipping Tool

Screenshot copied to clipboard and saved Select here to mark up and share the image

12:07 13-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT
WAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions (2 hours, 10 mins)

- Introduction to Python Reading Material (25 mins)
- Conditional Statements and Loops Reading Material (15 mins)
- List and Functions Reading Material (25 mins)
- Tuples, Sets and Dictionaries Reading Material (25 mins)
- OOPs Reading Material (20 mins)
- Miscellaneous Topics Reading Material (20 mins)
- More Interview Questions (20 mins)

The code defines a function multiply_digits that takes a parameter num. Inside the function, there's a for loop that iterates through each character(char) in the input num.

2. Checking for Zero:

```
1 if digit == 0:  
2     return 0
```

Within the loop, each character is converted to an integer. There's a conditional check to see if the digit is 0. If it is, the function immediately returns 0.

3. Multiplying Digits:

```
1 result *= digit
```

For each digit, it is multiplied with the result variable. After processing all digits, the function returns the final result.

Continue

Ask doubt

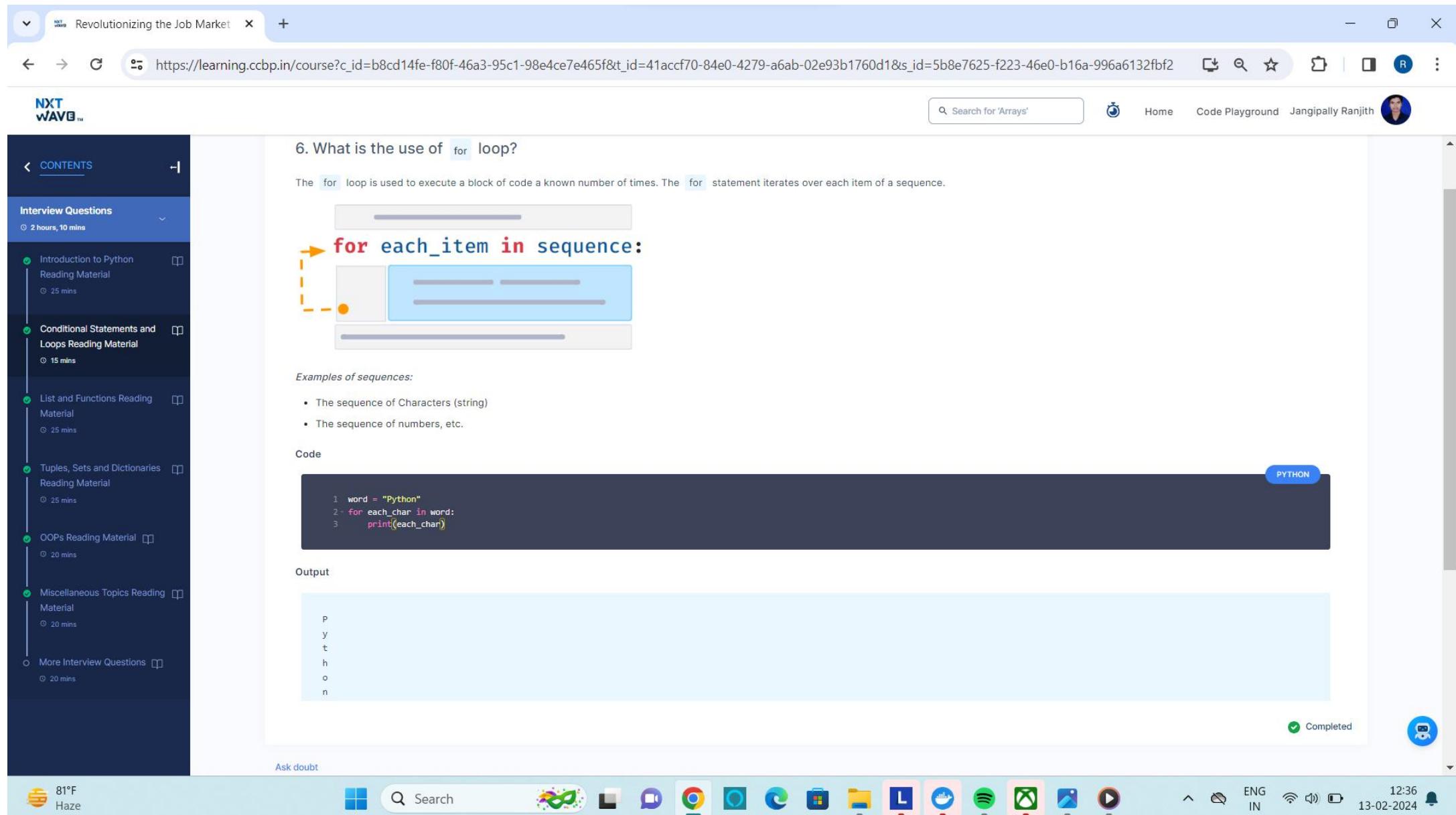
Notes Discussions

+ NEW NOTE

81°F Haze

Search

L 12:07 13-02-2024 ENG IN



Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=5b8e7625-f223-46e0-b16a-996a6132fbf2

NXT
wAVe™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

7. What are `range()` and `xrange()` functions?

`range():`

The `range()` function generates a sequence of integers starting from 0 to n(n is not included) and returns it.

Syntax: `range(n)`

Code

```
1 for number in range(3):
2     print(number)
```

PYTHON

Output

```
0
1
2
```

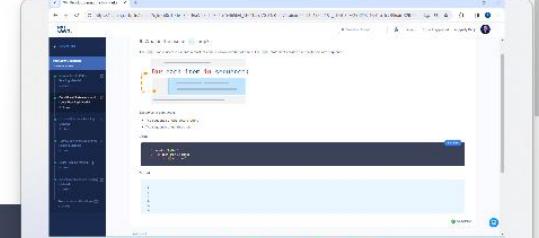
Range with Start and End

Syntax: `range(start, end)`

Generates a sequence of numbers starting from `start` to `end` (`end` is not included).

Code

```
1 for number in range(5, 8):
2     print(number)
```



Snipping Tool

Screenshot copied to clipboard and saved
Select here to mark up and share the image

Ask doubt

81°F Haze

Search

12:36 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=5b8e7625-f223-46e0-b16a-996a6132fbf2

NXT
wAVe™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Output

```
5
6
7
```

xrange():

The `xrange()` function generates a sequence of integers starting from 0 similar to `range()` function.

Code

```
1 - for number in xrange(4):
2 -     print(number)
```

PYTHON

Output

```
0
1
2
3
```

Note: The `xrange()` function is deprecated from Python3. You cannot run `xrange()` function in our code playground.

Q. What is the range function in Python?

The `range()` function returns a sequence of numbers, starting from 0 by default, and increment the step by 1 (by default), and stops before a specified number.

Completed

Ask doubt

81°F Haze

Search

L

12:36 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=5b8e7625-f223-46e0-b16a-996a6132fbf2

NXT
wAVe™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. What is the range function in Python?

The `range()` function returns a sequence of numbers, starting from 0 by default, and increment the step by 1 (by default), and stops before a specified number.

Syntax:

```
range([start, stop, step])
```

```
1 x = range(3, 9, 2)
2 for n in x:
3     print(n) #Output: 3 5 7
```

PYTHON

Question 1 of 6

By default, what is the starting number when using `range()` in Python?

- 1
- 0
- 1
- It starts at the end of the list.

Show Answer Submit

Ask doubt

81°F Haze

Search

Snipping Tool

Screenshot copied to clipboard and saved
Select here to mark up and share the image

12:36 13-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=5b8e7625-f223-46e0-b16a-996a6132fbf2

NXT
wAVe™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

8. When does an infinite loop occurs?

An infinite loop occurs when the condition always evaluates to True i.e. incorrect termination condition.

Example

Code

```
1 a = 10
2 while a > 3:
3     a = a + 1
4     print(a)
```

PYTHON

In the above code the while loop will run infinite times as the condition always evaluates to True .

9. What are Nested Loops?

An inner loop within the repeating block of an outer loop is called Nested Loop.

The **Inner Loop** will be executed one time for each iteration of the **Outer Loop**.

for item in sequence A: Outer Loop

for item in sequence B: Inner Loop

Block 1

Block 2

Completed

Ask doubt

81°F Haze

Search

12:36 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=5b8e7625-f223-46e0-b16a-996a6132fbf2

NXT WAVE

CONTENTS

Interview Questions (2 hours, 10 mins)

- Introduction to Python (25 mins)
- Conditional Statements and Loops (15 mins)
- List and Functions (25 mins)
- Tuples, Sets and Dictionaries (25 mins)
- OOPs (20 mins)
- Miscellaneous Topics (20 mins)
- More Interview Questions (20 mins)

9. What are Nested Loops?

An inner loop within the repeating block of an outer loop is called Nested Loop.

The **Inner Loop** will be executed one time for each iteration of the **Outer Loop**.

```
for item in sequence A:
    Block 1
        for item in sequence B:
            Block 2
            Block 3
            Block 4
```

Code

```
1. for i in range(2):
2.     print("Outer: " + str(i))
3.     for j in range(2):
4.         print("Inner: " + str(j))
```

PYTHON

Output

```
Outer: 0
Inner: 0
Inner: 1
Outer: 1
Inner: 0
Inner: 1
```

Daily News

Do not display sponsored content

China and Russia no longer perceived as top security threats, research finds

Stock gains to survive a market correction if rate cuts are delayed, one economist says

Ask doubt

81°F Haze

Search

Windows Start

Google Chrome

Microsoft Edge

File Explorer

OneDrive

PowerShell

Task View

Spotify

Xbox

YouTube

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=5b8e7625-f223-46e0-b16a-996a6132fbf2

NXT WAVE

CONTENTS

Interview Questions (2 hours, 10 mins)

- Introduction to Python (Reading Material, 25 mins)
- Conditional Statements and Loops (Reading Material, 15 mins)
- List and Functions (Reading Material, 25 mins)
- Tuples, Sets and Dictionaries (Reading Material, 25 mins)
- OOPs (Reading Material, 20 mins)
- Miscellaneous Topics (Reading Material, 20 mins)
- More Interview Questions (20 mins)

Q. How does Inner loop work in nested loops in Python?

In Python, the term **inner loop** typically refers to a loop that is nested (inside) within another loop.

Example:

```
1: for i in range(2): # Outer loop
2:     print("Outer loop iteration:", i)
3:
4:     for j in range(3): # Inner loop
5:         print("Inner loop iteration:", j)
```

Explanation:

We have two loops here, first one is the `Outer loop` and the nested one is the `Inner loop`. For each iteration of the outer loop, the inner loop runs completely from start to finish. So, for each value of `i` in the outer loop, the inner loop executes its iterations.

← Question 1 of 2

How many times the `print` statement is going to execute in the given Python code?

```
1: for i in range(4): # Outer loop
2:     for j in range(3): # Inner loop
3:         print("Executing inner loop iteration:", j)
```

4 times
 3 times
 12 times
 7 times

Show Answer Submit

10. What is a break statement?/ How to exit from a loop?

Break statement makes the program exit a loop early.

Completed

Ask doubt

81°F Haze

Search

12:37 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=5b8e7625-f223-46e0-b16a-996a6132fbf2

NXT WAVE

CONTENTS

Interview Questions (2 hours, 10 mins)

- Introduction to Python (25 mins)
- Conditional Statements and Loops (15 mins)
- List and Functions (25 mins)
- Tuples, Sets and Dictionaries (25 mins)
- OOPs (20 mins)
- Miscellaneous Topics (20 mins)
- More Interview Questions (20 mins)

10. What is a break statement?/ How to exit from a loop?

Break statement makes the program exit a loop early.

while condition:

Using Break

Generally, `break` is used to exit a loop when a condition is satisfied.

for item in Sequence:

In the below example, when the variable `i` value equals to `3`, then the `break` statement gets executed and stops the execution of the loop further.

Code

```
1: for i in range(5):
2:     if i == 3:
3:         break
4:     print(i)
5: print("END")
```

Output

```
0
1
```

Ask doubt

81°F Haze

Search

Snipping Tool

Screenshot copied to clipboard and saved
Select here to mark up and share the image

12:37 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=5b8e7625-f223-46e0-b16a-996a6132fbf2

NXT
WAVB

CONTENTS

Interview Questions (2 hours, 10 mins)

- Introduction to Python (25 mins)
- Conditional Statements and Loops (15 mins)
- List and Functions (25 mins)
- Tuples, Sets and Dictionaries (25 mins)
- OOPs (20 mins)
- Miscellaneous Topics (20 mins)
- More Interview Questions (20 mins)

Break in Nested Loop

The `break` in the inner loop stops the execution of the inner loop.

Code

```
1. for i in range(4):
2.     for j in range(4):
3.         if i * j > 10:
4.             break
5.         if (i > 6) and (j > 6):
6.             print(i * j)
```

PYTHON

Output

```
3
6
9
```

11. What is `continue` in loops?

The `continue` statement makes the program skip the remaining statements in the current iteration and begin the next iteration.

Ask doubt

81°F Haze

Search

Snipping Tool

Screenshot copied to clipboard and saved
Select here to mark up and share the image

12:37 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=5b8e7625-f223-46e0-b16a-996a6132fbf2

NXT
WAVB

CONTENTS

Interview Questions (2 hours, 10 mins)

- Introduction to Python (Reading Material, 25 mins)
- Conditional Statements and Loops (Reading Material, 15 mins)
- List and Functions (Reading Material, 25 mins)
- Tuples, Sets and Dictionaries (Reading Material, 25 mins)
- OOPs (Reading Material, 20 mins)
- Miscellaneous Topics (Reading Material, 20 mins)
- More Interview Questions (20 mins)

11. What is `continue` in loops?

The `continue` statement makes the program skip the remaining statements in the current iteration and begin the next iteration.

while condition A:

Using continue

Generally, `continue` is used to skip the remaining statements in the current iteration when a condition is satisfied.

for item in Sequence:

In the below example, when the variable `i` value equals to `3`, then the next statements in the loop body are skipped.

Code

```
1: for i in range(5):
2:     if i == 3:
3:         continue
4:     print(i)
5: print("END")
```

PYTHON

Output

```
0
1
2
4
END
```

Completed

Ask doubt

81°F Haze

Search

12:37 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=5b8e7625-f223-46e0-b16a-996a6132fbf2

NXT
wAVe™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

12. What is `pass` in Python?

The `pass` statement is a syntactic placeholder. When it is executed, nothing happens. Generally it used when we have to test the code before writing the complete code.

```
if condition A:  
    Block 1  
elif condition B:  
    pass  
else:  
    Block 3
```

Empty Loops

We can use `pass` statements to test code written so far, before writing loop logic.

```
while condition A:  for item in Sequence:  
    pass            pass
```

Completed

Ask doubt

Notes Discussions

81°F Haze

Search

12:38 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=5b8e7625-f223-46e0-b16a-996a6132fbf2

NXT
WAVES™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. Write a Python program to print from 1 to 100.

PYTHON

```
1 - for i in range(1,101):
2 -     print(i)
```

Explanation:

The code uses a `for` loop to iterate over a range of numbers. `range(1, 101)` generates a sequence of numbers starting from 1 up to 101, but not including 101. So, the loop will iterate through the numbers 1 to 100.

Question 1 of 2

Which of the following generate the numbers 5 to 15 in Python?

- `for i in range(5, 16):`
- `for i in range(1, 11):`
- `for i in range(5, 15):`
- `for i in range(15, 5):`

Next

Completed

Ask doubt

Notes Discussions

81°F Haze

Search

12:38 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=5b8e7625-f223-46e0-b16a-996a6132fbf2

NXT
wAVe™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions (2 hours, 10 mins)

- Introduction to Python (25 mins)
- Conditional Statements and Loops (15 mins)
- List and Functions (25 mins)
- Tuples, Sets and Dictionaries (25 mins)
- OOPs (20 mins)
- Miscellaneous Topics (20 mins)
- More Interview Questions (20 mins)

Q. Write a Python program that reverses a number using mathematical operations.

PYTHON

```
1 number = int(input())
2
3 reversed_number = 0
4
5 while number > 0:
6     digit = number % 10
7     reversed_number = (reversed_number * 10) + digit
8     number = number // 10
9
10 print(reversed_number)
```

Explanation:

The Python code takes an integer number as input. The while loop repeatedly extracts the last digit of the original number, adds it to the reversed number, and removes it from the original number until all digits are processed.

← Question 1 of 2

What is the purpose of the `digit = number % 10` line in the given Python code?

PYTHON

```
1 number = int(input())
2
```

Completed

Ask doubt

Notes Discussions

Notes

+ NEW NOTE

81°F Haze

Search

12:38 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT
wAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Interview Questions

Recently Asked Questions and Answers

1. What are the Data Structures?

Data Structures allow us to store and organize data efficiently. This will allow us to easily access and perform operations on the data.

In Python, there are four built-in data structures

- List*
- Tuple*
- Set*
- Dictionary*

2. What is a list?

The List is the most versatile python data structure that holds an ordered sequence of items.

Creating a List

Miralam Tank Ro... Construction

Search

12:42 13-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT
wAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPS Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Interview Questions > List and Functions Reading Material

2. What is a list?

The List is the most versatile python data structure that holds an ordered sequence of items.

A diagram illustrating a list in Python. It shows four items: "5", "Six", 2, and 8.2, each enclosed in a dashed box. Below the list, their corresponding indices are shown: 0, 1, 2, and 3. The indices are in red, while the list elements are in green.

Creating a List

A List can be created by enclosing elements within [square] brackets where each item is separated by a comma.

Code

```
1 a = 2
2 list_a = [5, "Six", a, 8.2]
3
4 print(type(list_a))
5 print(list_a)
```

Output

```
<class 'list'>
[5, 'Six', 2, 8.2]
```

A screenshot of the Windows Snipping Tool interface, showing the captured area of the browser window. The captured area includes the left sidebar with course contents, the main content area with the list example, and the top navigation bar. The status bar at the bottom of the snipper shows 'Screenshot copied to clipboard and saved'.

Screenshot copied to clipboard and saved
Select here to mark up and share the image

Miralam Tank Ro... Construction

Search

12:43 13-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT
WAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Interview Questions > List and Functions Reading Material

3. Explain a few List Methods?

Append

Adds an element to the end of the list.

Syntax

```
list.append(value)
```

Code

```
1 list_a = []
2 for x in range(1, 4):
3     list_a.append(x)
4
5 print(list_a)
```

PYTHON

Output

```
[1, 2, 3]
```

Snipping Tool

Screenshot copied to clipboard and saved Select here to mark up and share the image

Miralam Tank Ro... Construction

Search

12:43 13-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT
WAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Interview Questions > List and Functions Reading Material

Extend

Adds all the elements of a sequence to the end of the list.

Syntax

```
list_a.extend(list_b)
```

Code

```
1 list_a = [1, 2, 3]
2 list_b = [4, 5, 6]
3 list_a.extend(list_b)
4
5 print(list_a)
```

PYTHON

Output

```
[1, 2, 3, 4, 5, 6]
```

Snipping Tool

Screenshot copied to clipboard and saved Select here to mark up and share the image

Miramal Tank Ro... Construction

Search

12:43 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT
wAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Interview Questions > List and Functions Reading Material

Insert

Inserts the given element at the specified index.

Syntax

```
list.insert(index, value)
```

Code

```
1 list_a = [1, 2, 3]
2 list_a.insert(1, 4)
3
4 print(list_a)
```

PYTHON

Output

```
[1, 4, 2, 3]
```

Snipping Tool

Screenshot copied to clipboard and saved Select here to mark up and share the image

Miramal Tank Ro... Construction

Search

12:43 13-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT
wAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Interview Questions > List and Functions Reading Material

Pop

Removes the last item from the list.

Syntax

```
list.pop()
```

Code

```
1 list_a = [1, 2, 3]
2 list_a.pop()
3
4 print(list_a)
```

PYTHON

Output

```
[1, 2]
```

Remove

Removes the first matching element from the list.

Syntax

```
list.remove(value)
```

Code

```
1 list_a = [1, 3, 2, 3]
```

Snipping Tool

Screenshot copied to clipboard and saved
Select here to mark up and share the image

Miramal Tank Ro... Construction

Search

12:43 13-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT WAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Interview Questions > List and Functions Reading Material

Remove

Removes the first matching element from the list.

Syntax

```
list.remove(value)
```

Code

```
1 list_a = [1, 3, 2, 3]
2 list_a.remove(3)
3
4 print(list_a)
```

PYTHON

Output

```
[1, 2, 3]
```

Clear

Removes all the items from the list.

Syntax

```
list.clear()
```

Code

```
1 list_a = [1, 2, 3]
2 list_a.clear()
```

Snipping Tool

Screenshot copied to clipboard and saved
Select here to mark up and share the image

Miramal Tank Ro... Construction

Search

12:43 13-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT
WAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Interview Questions > List and Functions Reading Material

Clear

Removes all the items from the list.

Syntax

```
list.clear()
```

Code

```
1 list_a = [1, 2, 3]
2 list_a.clear()
3
4 print(list_a)
```

PYTHON

Output

```
[]
```

Index

Returns the index of the first occurrence of the specified item in the list.

Syntax

```
list.index(item)
```

Code

```
1 list_a = [1, 3, 2, 3]
2 index = list_a.index(3)
3
```

Snipping Tool

Screenshot copied to clipboard and saved
Select here to mark up and share the image

84°F Mostly cloudy

Search

L

12:43 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT
wAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Interview Questions > List and Functions Reading Material

Index

Returns the index of the first occurrence of the specified item in the list.

Syntax

```
list.index(item)
```

Code

```
1 list_a = [1, 3, 2, 3]
2 index = list_a.index(3)
3
4 print(index)
```

PYTHON

Output

```
1
```

Count

Returns the number of elements with the specified value.

Syntax

```
list.count(value)
```

Code

```
1 list_a = [1, 2, 3]
2 count = list_a.count(2)
3
```

Snipping Tool

Screenshot copied to clipboard and saved
Select here to mark up and share the image

84°F Mostly cloudy

Search

L

12:43 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT
wAVe™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Interview Questions > List and Functions Reading Material

Count

Returns the number of elements with the specified value.

Syntax

```
list.count(value)
```

Code

```
1 list_a = [1, 2, 3]
2 count = list_a.count(2)
3
4 print(count)
```

PYTHON

Output

```
1
```

Sort

Sorts the items of a list in ascending order. The `sort()` method modifies the original list.

Syntax

```
list.sort()
```

Code

```
1 list_a = [1, 3, 2]
2 list_a.sort()
3
```

Snipping Tool

Screenshot copied to clipboard and saved
Select here to mark up and share the image

84°F Mostly cloudy

Search

L

12:43 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT
WAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Interview Questions > List and Functions Reading Material

1

Sort

Sorts the items of a list in ascending order. The `sort()` method modifies the original list.

Syntax

```
list.sort()
```

Code

```
1 list_a = [1, 3, 2]
2 list_a.sort()
3
4 print(list_a)
```

PYTHON

Output

```
[1, 2, 3]
```

Sorted

Sorts the items of a list in ascending order. The `sorted()` method returns the modified list.

Syntax

```
sorted()
```

Code

```
list_a = [1, 3, 2]
print(sorted(list_a))
```

Snipping Tool

Screenshot copied to clipboard and saved
Select here to mark up and share the image

84°F Mostly cloudy

Search

L

12:44 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT
wAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Interview Questions > List and Functions Reading Material

Sorted

Sorts the items of a list in ascending order. The `sorted()` method returns the modified list.

Syntax

```
sorted()
```

Code

```
1 list_a = [1, 3, 2]
2 list_b = sorted(list_a)
3
4 print(list_b)
```

PYTHON

Output

```
[1, 2, 3]
```

Code

```
1 list_a = [1, 3, 2]
2 sorted(list_a)
3
4 print(list_a)
```

Output

```
[1, 3, 2]
```

Snipping Tool

Screenshot copied to clipboard and saved
Select here to mark up and share the image

84°F Mostly cloudy

Search

L

12:44 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT
wAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Interview Questions > List and Functions Reading Material

Q. What is the difference between Range and List?

Range	List
range is a built-in function in Python that generates a sequence of numbers.	A list is a data structure in Python that can hold an ordered collection of items.
It is commonly used for iterating over a sequence of numbers in a loop.	Lists can store elements of different data types, and they are mutable.
More memory-efficient for large ranges.	Consumes more memory, especially for large lists.
Syntax: range(start, stop, step)	Syntax: [element1, element2, ...]

Question 1 of 4

What is the primary task of the range function in Python?

- Storing a collection of items
- Generating a sequence of numbers
- Sorting a list
- Modifying list elements

Show Answer Submit

Q. With which data types can we use the pop() method in Python?

The `pop()` method can be used with three different data types in Python: lists, dictionaries, and sets.

- Lists:** The `pop()` method removes the specified item from a list. If no index is specified, the last element is removed.

84°F Mostly cloudy

Search

Snipping Tool Screenshot copied to clipboard and saved Select here to mark up and share the image

ENG IN 12:44 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT wAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Interview Questions > List and Functions Reading Material

Q. With which data types can we use the pop() method in Python?

The `pop()` method can be used with three different data types in Python: lists, dictionaries, and sets.

- Lists:** The `pop()` method removes the specified item from a list. If no index is specified, the last element is removed.
- Dictionaries:** The `pop()` method removes the specified item from a dictionary. The method returns the value of the removed item.
- Sets:** The `pop()` method removes a random item from a set. The method returns the removed item.

► More Information

Question 1 of 4

What does the `pop()` method return when used with sets?

- The removed item's index
- The removed item
- The set itself
- An error message

Show Answer Submit

Snipping Tool Screenshot copied to clipboard and saved Select here to mark up and share the image

84°F Mostly cloudy

Search

Cloud, Chat, Google Chrome, Microsoft Edge, File Explorer, LinkedIn, Spotify, YouTube

ENG IN 12:44 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT
wAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Interview Questions > List and Functions Reading Material

Q. Explain List extends in Python?

Intent of the interviewer:

Extending a list in Python means adding one or more elements to the end of the list. There are several ways to extend a list in Python, including:

- Using the `extend()` method: Adds all the elements of an iterable to the end of the list.
- Using the `+` operator: Concatenate two lists together.
- Using `list slicing`: Insert a new element or list of elements into the end of a list.

► More Information

Question 1 of 5

When using list slicing to extend a list, which of the following is a correct statement?

- You can only use slicing to replace existing elements, not to add new ones
- Slicing can be used to add elements at any position within the list, including the end
- Using slicing to add elements to a list will always create a new list
- Slicing is not an appropriate method to extend a list with new elements

Show Answer Submit

Q. What will be the output of this code?

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

PYTHON

84°F Mostly cloudy

Search

Cloud, Chat, Google Chrome, Microsoft Edge, File, L, Spotify, Xbox, YouTube

ENG IN 12:44 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT
WAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Interview Questions > List and Functions Reading Material

Q. What will be the output of this code?

PYTHON

```
1 temp=[3,4]
2 lst=[1,2]
3
4 lst.append(temp) # Output : [3, 4, [1, 2]]
5 lst.extend(temp) # Output : [3, 4, [1, 2], 1, 4]
```

Explanation:

The **append()** method is used to add an entire list to another list. In this case, `temp` is appended to `lst`, resulting in `lst` containing 1, 2, and the entire `temp` list.

The **extend()** method is used to add the elements of one list to the end of another list. In this case, the elements of `temp` are added to the end of `lst`. Therefore, `lst` will contain the following elements: 1, 2, [3, 4], 3, 4.

Question 1 of 4

What will be the output after executing the following code?

PYTHON

```
1 temp = [3, 4]
2 lst = [1, 2]
3
4 lst.extend(temp)
5 print(lst)
```

[3, 4, [1, 2]]
[3, 4, [1, 2], 1, 2]
[3, 4, 1, 2]
[1, 2, 3, 4]

84°F Mostly cloudy

Search

L

12:44 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT
wAVe™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Interview Questions > List and Functions Reading Material

Q. Can you create a Python list and add a new element?

Yes, I can create a Python list and add a new element.

To create python list we use [] square brackets.

```
1 my_list = [1,2,3]
```

PYTHON

There are four methods to add elements to a List in Python.

- append():** append the element to the end of the list.
- insert():** This method inserts the element at a specific position in the list.
- extend():** Adds all the elements of an iterable to the end of the list.
- List Concatenation:** Concatenate two lists together.

► More Information

Question 1 of 4

How do you concatenate two Python lists together?

- Using the + operator
- Using the concat() method
- Using the merge() function
- Using the join() method

Next

84°F Mostly cloudy

Search

Cloud, Search, Chat, Google Chrome, Microsoft Edge, File Explorer, LinkedIn, Spotify, Xbox, YouTube, Volume, Network, Battery, ENG IN, 12:45, 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT
wAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Interview Questions > List and Functions Reading Material

Q. What is the difference between the `append()` and `extend()` methods for lists in Python?

When I use `append()`, it adds a single element to the end of the list. This element can be of any data type, and it's added as a single item. So, if I append another list to an existing list, the entire list will be added as a single element.

On the other hand, `extend()` is used to merge two lists or add multiple elements. When I use `extend()` with a list as an argument, each element of the list argument is added individually to the end of the original list.

► More Information

Question 1 of 5

What will be the output after executing the following code?

PYTHON

```
1 my_list = [1, 2, 3]
2 another_list = [4, 5, 6]
3 my_list.extend(another_list)
4 print(my_list)
```

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

[4, 5, 6]

[1, 2, 3]

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

Next

5. How to use the `split()` method?

84°F Mostly cloudy

Search

L

ENG IN

12:45 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT
WAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPS Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Interview Questions > List and Functions Reading Material

5. How to use the `split()` method?

The `split()` splits a string into a list at every specified separator.

Syntax:

```
str_var.split(separator)
```

Example

Code

```
1 nums = "1 2 3 4"
2 num_list = nums.split()
3 print(num_list)
```

PYTHON

Output

```
[1, 2, 3, 4]
```

If no separator is specified, the default separator is whitespace.

Using separator

Example

Code

```
1 nums = "1,2,3,4"
2 num_list = nums.split(',')
3 print(num_list)
```

Snipping Tool

Screenshot copied to clipboard and saved
Select here to mark up and share the image

84°F Mostly cloudy

Search

L

12:45 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT
WAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPS Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Interview Questions > List and Functions Reading Material

If no separator is specified, the default separator is whitespace.

Using separator

Example

Code

```
1 nums = "1,2,3,4"
2 num_list = nums.split(',')
3 print(num_list)
```

PYTHON

Output

['1', '2', '3', '4']

6. How to use the `join()` method?

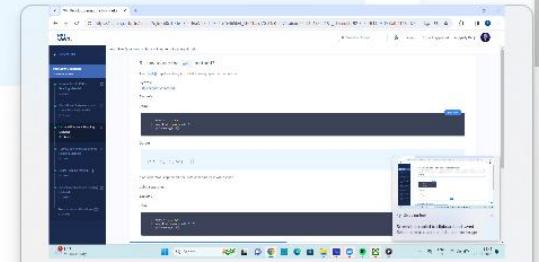
The `join()` takes all the items in a sequence of strings and joins them into one string.

Syntax:

```
str.join(sequence)
```

Example

Code



Snipping Tool

Screenshot copied to clipboard and saved
Select here to mark up and share the image

84°F Mostly cloudy

Search

12:45 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT
wAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Interview Questions > List and Functions Reading Material

6. How to use the `join()` method?

The `join()` takes all the items in a sequence of strings and joins them into one string.

Syntax:

```
str.join(sequence)
```

Example

Code

```
list_a = ['Python is ', ' program', 'mming l', 'ngu', 'ge']
string_a = "-".join(list_a)
print(string_a)
```

Output

```
Python is a programming language
```

Snipping Tool

Screenshot copied to clipboard and saved
Select here to mark up and share the image

7. What is List Slicing?

Obtaining a part of a list is called List Slicing.

Syntax:

```
variable_name[start_index:end_index]
```

end index is not included in the slice.

84°F Mostly cloudy

Search

L

12:45 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT WAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Interview Questions > List and Functions Reading Material

7. What is List Slicing?

Obtaining a part of a list is called List Slicing.

Syntax:

```
variable_name[start_index:end_index]
```

- `end_index` is not included in the slice.

list_a

Code

```
1 list_a = [5, "Six", 2, 8.2]
2 list_b = list_a[:2]
3 print(list_b)
```

Output

```
[5, 'Six']
```

Extended Slicing

Snipping Tool

Screenshot copied to clipboard and saved
Select here to mark up and share the image

84°F Mostly cloudy

Search

12:45 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT
WAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material** 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Interview Questions > List and Functions Reading Material

Extended Slicing

Similar to string extended slicing, we can extract alternate items from the list using the step.

Syntax:

```
variable[start_index:end_index:step]
```

Code

```
list_a = ["R", "B", "G", "O", "W"]
list_b = list_a[0:5:2]
print(list_b)
```

PYTHON

Output

```
[‘R’, ‘O’]
```

Snipping Tool

Screenshot copied to clipboard and saved Select here to mark up and share the image

Q. What is Slicing in python?

Slicing in Python refers to the technique of extracting a portion of a sequence, such as a list, tuple, or string. It allows you to create a new sequence by specifying a range of indices.

Syntax:

```
sequence[start:stop:step]
```

84°F Mostly cloudy

Search

L

12:45 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT
wAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Interview Questions > List and Functions Reading Material

Q. What is Slicing in python?

Slicing in Python refers to the technique of extracting a portion of a sequence, such as a list, tuple, or string. It allows you to create a new sequence by specifying a range of indices.

Syntax:

```
1 sequence[start:stop:step]
```

PYTHON

► More Information

Question 1 of 4

Which statement is correct about slicing in Python?

- Slicing always creates a deep copy of the original sequence.
- The ending index in a slice is inclusive.
- The starting index in a slice is inclusive.
- The step parameter is mandatory in a slice.

Show Answer Submit

8. What is Negative Indexing?

Using a negative index returns the nth item from the end of the list.

84°F Mostly cloudy

Search

Snipping Tool

Screenshot copied to clipboard and saved Select here to mark up and share the image

ENG IN 12:46 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT
wAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Interview Questions > List and Functions Reading Material

8. What is Negative Indexing?

Using a negative index returns the nth item from the end of the list.

The last item in the list can be accessed with the index -1

list_a[-1]

Accessing List Items with Negative Index

Example-1

Code

PYTHON

```
1 list_a = [5, 4, 3, 2, 1]
2 item = list_a[-1]
3 print(item)
```

Output

1

84°F Mostly cloudy

Search

L

ENG IN

12:46 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT
wAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Interview Questions > List and Functions Reading Material

Example-2

Code

```
list_a = [5, 4, 3, 2, 1]
item = list_a[-4]
print(item)
```

PYTHON

Output

```
4
```

Slicing With Negative Index

You can also specify negative indices while slicing a List.

Code

```
list_a = [5, 4, 3, 2, 1]
list_b = list_a[-3:-1]
print(list_b)
```

PYTHON

Output

```
[3, 2]
```

list_a[-3:-1]

84°F Mostly cloudy

Search

Cloud, Mask, Photo, Chat, Google Chrome, Microsoft Edge, File, LinkedIn, Spotify, Xbox, YouTube

ENG IN

12:46 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT WAVE

CONTENTS

Interview Questions (2 hours, 10 mins)

- Introduction to Python Reading Material (25 mins)
- Conditional Statements and Loops Reading Material (15 mins)
- List and Functions Reading Material (25 mins) **Current Section**
- Tuples, Sets and Dictionaries Reading Material (25 mins)
- OOPs Reading Material (20 mins)
- Miscellaneous Topics Reading Material (20 mins)
- More Interview Questions (20 mins)

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

Interview Questions > List and Functions Reading Material

Output: [3, 2]

list_a[-3:-1]

Out of Bounds Index

While slicing, the index can go beyond the size of the list.

Code:

```
1 list_a = [5, 4, 3, 2, 1]
2 list_b = list_a[-6:-2]
3 print(list_b)
```

PYTHON

Output: [5, 4, 3]

9. How to reverse a List?

Reversing a list using `reversed()` method:

84°F Mostly cloudy

Search

Cloud, Search, Chat, Google Chrome, Microsoft Edge, File Explorer, LinkedIn, Spotify, Xbox, YouTube, Weather, Network, Battery, Volume, Language (ENG IN), Date (13-02-2024)

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT WAVE

CONTENTS

Interview Questions < 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Interview Questions > List and Functions Reading Material

9. How to reverse a List?

Reversing a list using `reverse()` method:

The `reverse()` method can be used to reverse a List. It updates the original list.

Code:

```
1 week_days = ['Monday', 'Tuesday', 'Wednesday']
2 week_days.reverse()
3
4 print(week_days)
```

PYTHON

Output

```
[['Wednesday', 'Tuesday', 'Monday']]
```

Reversing a list using list slicing:

A list can be reversed using extended slicing.

Syntax:

```
variable[start:end:negative_step]
```

-1 for step will reverse the order of items in the list.

Code

```
1 list_a = [5, 4, 3, 2, 1]
2 list_b = list_a[::-1]
3 print(list_b)
```

Output

```
[1, 2, 3, 4, 5]
```

Snipping Tool

Screenshot copied to clipboard and saved
Select here to mark up and share the image

84°F Mostly cloudy

Search

L

12:46 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT WAVE

CONTENTS

Interview Questions < 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Interview Questions > List and Functions Reading Material

Q. What is the difference between list and array?

Lists and arrays are data structures that store multiple items, but have different features and uses.

List	Array
Can hold items of multiple data types.	Holds items of the same data type.
Cannot handle arithmetic operations.	Can handle arithmetic operations.
Consumes more memory.	Consumes less memory.
All elements can be accessed directly.	A loop is needed to access elements.
Better suited for shorter sequences of data.	More efficient for longer sequences of data.

10. Explain about functions in Python?

A function is a block of reusable code to perform a specific action. Functions help us in using existing code without writing it every time we need it.

A function can be defined using a keyword `def`. A function is uniquely identified by the `function_name`.

`def function_name():`

Reusable Block of Code

Code

```
1: def greet():
2:     print("Hello")
3:
4: greet()
5: greet()
```

Output

```
Hello
Hello
```

Snipping Tool

Screenshot copied to clipboard and saved
Select here to mark up and share the image

84°F Mostly cloudy

Search

L

12:46 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT WAVE

Interview Questions > List and Functions Reading Material

Function Arguments

We can pass values to a function using Argument.

```
def function_name(args):
```

Reusable Block of Code

Code

```
1: def greet(word):
2:     msg = "Hello " + word
3:     print(msg)
4:
5: name1 = input()
6: name2 = input()
7: greet(word = name1)
8: greet(word = name2)
```

PYTHON

Input

```
Teja
Rahul
```

Output

```
Hello Teja
Hello Rahul
```

Providing default values

Default values indicate that the function argument will take that value if no argument value is passed during the function call.

Example

Code

```
1: def greet(arg_1 = "Hi", arg_2 = "Ran"):
2:     print(arg_1 + " " + arg_2)
3:
4: greeting = input()
5: name = input()
```

Snipping Tool

Screenshot copied to clipboard and saved
Select here to mark up and share the image

84°F
Mostly cloudy

Search

L

ENG IN

12:47
13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT WAVE

Interview Questions > List and Functions Reading Material

Hello Teja
Hello Rahul

Providing default values

Default values indicate that the function argument will take that value if no argument value is passed during the function call.

Example

Code

```
1. def greet(arg_1 = "Hi", arg_2 = "Ram"):
2.     print(arg_1 + " " + arg_2)
3.
4. greeting = input()
5. name = input()
6. greet()
```

PYTHON

Input

Hello
Teja

Output

Hi Ram

11. What is Recursion?

A function calling itself is called **Recursion**

```
def function_1():
```

function_1()

NXT WAVE

Snipping Tool

Screenshot copied to clipboard and saved
Select here to mark up and share the image

84°F
Mostly cloudy

Search

L

12:47
13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT WAVE

CONTENTS

Interview Questions < 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Interview Questions > List and Functions Reading Material

11. What is Recursion?

A function calling itself is called **Recursion**

```
def function_1():
```

Let's understand recursion with a simple example of multiplying N numbers

Multiply N Numbers

```
1. def factorial(n): # Recursive Function
2.     if n == 1: # Base Case
3.         return 1
4.     return n * factorial(n - 1) # Recursion
5. num = int(input())
6. result = factorial(num)
7. print(result)
```

PYTHON

Input

5

Output

120

Snipping Tool

Screenshot copied to clipboard and saved
Select here to mark up and share the image

Companies Asked: Translytics Business Services, Xcube Labs

84°F Mostly cloudy

Search

12:47 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT WAVE

Interview Questions > List and Functions Reading Material

CONTENTS

Interview Questions (2 hours, 10 mins)

- Introduction to Python Reading Material (25 mins)
- Conditional Statements and Loops Reading Material (15 mins)
- List and Functions Reading Material (25 mins)
- Tuples, Sets and Dictionaries Reading Material (25 mins)
- OOPs Reading Material (20 mins)
- Miscellaneous Topics Reading Material (20 mins)
- More Interview Questions (20 mins)

Q. What is a Lambda function and what is the main purpose of it?

Companies Asked: Translytics Business Services, Xcube Labs

A lambda function is an anonymous function used for doing simple operations. Lambda functions can have any number of arguments, but can only have one expression.

The primary purpose of a lambda function is to perform simple operations without the need to formally define a function using the typical `def` keyword. They are often used for short, operations that are only needed temporarily without the need of a full function definition.

Here's a breakdown of the lambda function:

Syntax: The general syntax of a lambda function in Python is:

```
1 lambda arguments: expression
```

The expression is executed and returned when the lambda function is called.

Examples:

- A lambda function that adds 10 to any number you provide:

```
1 f = lambda x: x + 10
2 print(f(5)) # Outputs: 15
```

- A lambda function that multiplies two numbers:

```
1 mul = lambda x, y: x * y
2 print(mul(3, 7)) # Outputs: 21
```

Use cases:

- Lambda functions are particularly useful when you need a simple function for a short period and don't want to formally define it.
- They're commonly used with functions like `map()`, `filter()`, and `sorted()` in Python.

```
1 # Using lambda with sorted() to sort a list of tuples by the second item
2 tuple_list = [(1, 2), (3, 1), (5, 10), (6, 6)]
3 sorted_list = sorted(tuple_list, key=lambda x: x[1])
4 print(sorted_list) # Outputs: [(3, 1), (1, 2), (6, 6), (5, 10)]
```

Question 1 of 5

Which code will give the required result of sorting the `tuple_list` by the second item in each tuple?

Snipping Tool

Screenshot copied to clipboard and saved
Select here to mark up and share the image

84°F Mostly cloudy

Search

12:47 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT WAVE

CONTENTS

Interview Questions < 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Interview Questions > List and Functions Reading Material

Q. What are generators and what are the uses of it?

Companies Asked: Translytics Business Services

A Python generator is a special function that returns an iterable set of items, one at a time. It is created using `yield` instead of `return`. Instead of running to completion and returning a single value, it produces a stream of values. Its execution can be paused after yielding each value and later resumed. This makes it perfect for handling large data sets.

Generators are defined like a normal function but with the `yield` keyword replacing the `return` keyword.

Example:

Here's a Python generator function that produces numbers in the Fibonacci sequence:

```
2. a, b = 0, 1
3. for _ in range(n):
4.     yield a
5.     a, b = b, a + b
6.
7. # Create a generator object
8. gen = fibonacci_generator(5)
9. # Iterate over the generator
10. for number in gen:
11.     print(number)
```

PYTHON

Outputs:

```
1 0
2 1
3 1
4 2
5 3
```

Uses:

- Memory Efficiency:** Ideal for producing large sequences without storing them entirely in memory.
- Large Data Streams:** For reading large files or database results in chunks, rather than loading everything at once.

Question 1 of 4

What is the output of the given Python code?

```
1. def fibonacci_generator(n):
2.     a, b = 0, 1
3.     for _ in range(n):
4.         yield a
```

PYTHON

Snipping Tool

Screenshot copied to clipboard and saved
Select here to mark up and share the image

84°F Mostly cloudy

Search

12:47 13-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT WAVE

Interview Questions > List and Functions Reading Material

Q. Write code for the Fibonacci series.

Companies Asked: Translytics Business Services

There are multiple ways to solve the Fibonacci series problem:

1. Using Recursion (Most expected in interviews)

```
1 def fibonacci_recursive(n):
2     if n == 0:
3         return 0
4     elif n == 1:
5         return 1
6     else:
7         return fibonacci_recursive(n - 1) + fibonacci_recursive(n - 2)
8
9 for i in range(10):
10    print(fibonacci_recursive(i), end=" ")
```

PYTHON

After executing the above code, the output will be 0 1 1 2 3 5 8 13 21 34.

Question 1 of 4

In the iterative version of the Fibonacci series code, what will be the value of variable `a` after the 3rd iteration in a call with `n = 5`?

```
1 def fibonacci_iterative(n):
2     a, b = 0, 1
3     for i in range(n):
4         a, b = b, a + b
5     return a
```

PYTHON

1
 2
 3
 4
 5

Show Answer Submit

Snipping Tool

Screenshot copied to clipboard and saved
Select here to mark up and share the image

Q. Write code for the factorial of a number.

Recursive Approach:

84°F Mostly cloudy

Search

12:47 13-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT WAVE

CONTENTS

Interview Questions (2 hours, 10 mins)

- Introduction to Python Reading Material (25 mins)
- Conditional Statements and Loops Reading Material (15 mins)
- List and Functions Reading Material (25 mins)
- Tuples, Sets and Dictionaries Reading Material (25 mins)
- OOPs Reading Material (20 mins)
- Miscellaneous Topics Reading Material (20 mins)
- More Interview Questions (20 mins)

Interview Questions > List and Functions Reading Material

Q. Write code for the factorial of a number.

Recursive Approach:

```
1. def factorial_recursive(n):
2.     if n < 0:
3.         print("n must be non-negative")
4.     elif n == 0:
5.         return 1
6.     else:
7.         return n * factorial_recursive(n - 1)
8.
9. n = 5
10. result = factorial_recursive(n)
```

The `factorial_recursive` function calculates the factorial of a non-negative integer `n` by multiplying `n` by the factorial of `n-1` (previous element), until `n` reaches 0.

► Another Approach:

Question 1 of 3

What does the `factorial_recursive` function do in the given Python code?

```
1. def factorial_recursive(n):
2.     if n == 0:
3.         return 1
4.     else:
5.         return n * factorial_recursive(n - 1)
6.
7. n = 5
8. result = factorial_recursive(n)
9. print(result)
```

It calculates the sum of numbers from 1 to `n`.

It calculates the factorial of a non-negative integer `n`.

It checks if `n` is an odd or even number.

It calculates the average of `n` numbers.

Q. Write a program to find multiplication of digits in a number?

```
1. def multiply_digits(num):
2.     result = 1
3.
```

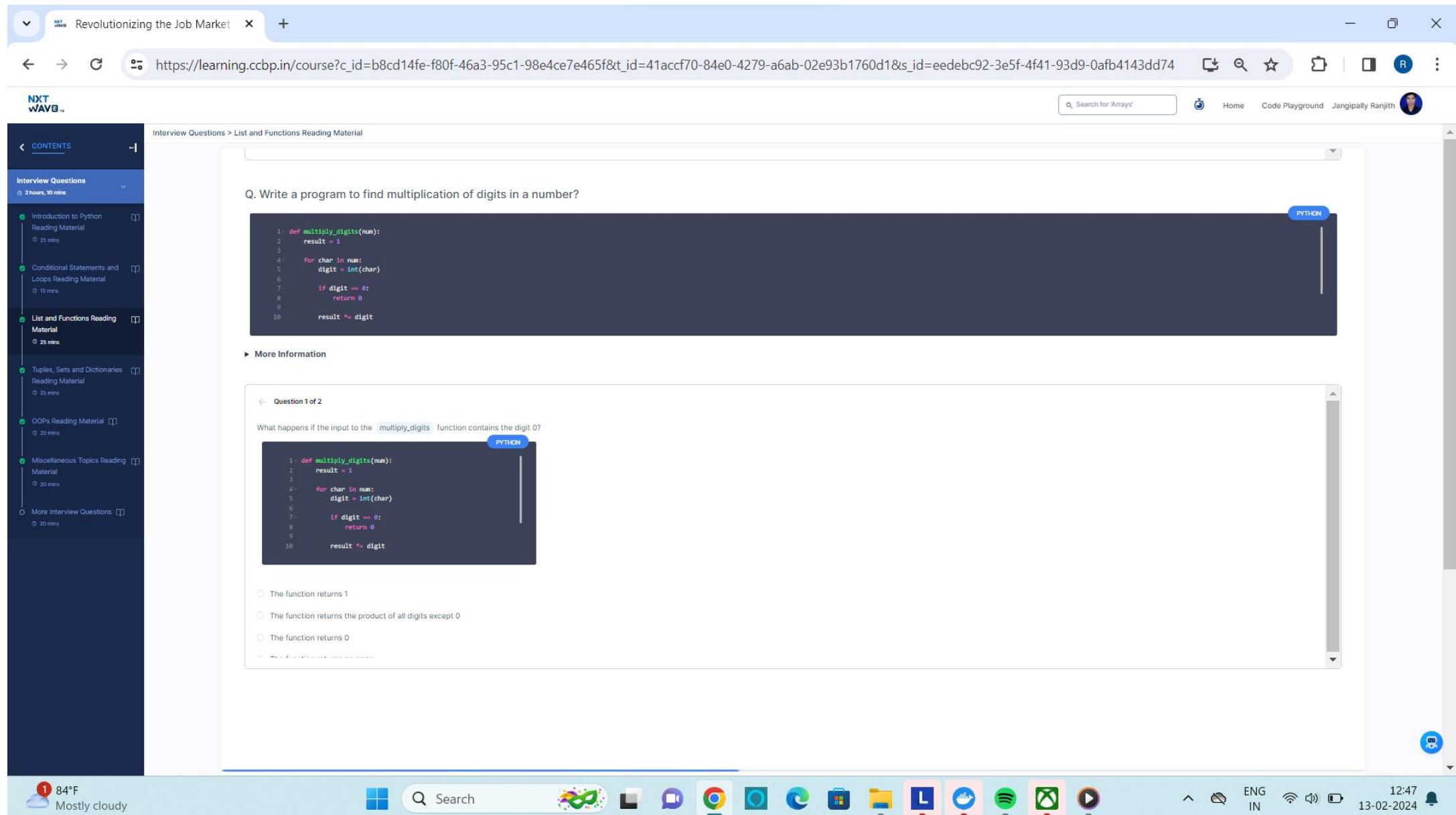
Snipping Tool

Screenshot copied to clipboard and saved
Select here to mark up and share the image

84°F Mostly cloudy

Search

12:47 13-02-2024 ENG IN



Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT WAVE

CONTENTS

Interview Questions (2 hours, 10 mins)

- Introduction to Python Reading Material (25 mins)
- Conditional Statements and Loops Reading Material (15 mins)
- List and Functions Reading Material (25 mins)
- Tuples, Sets and Dictionaries Reading Material (25 mins)
- OOPs Reading Material (20 mins)
- Miscellaneous Topics Reading Material (20 mins)
- More Interview Questions (20 mins)

Interview Questions > List and Functions Reading Material

Q. Write a program to find multiplication of digits in a number?

```
5     digit = int(char)
6
7     if digit == 0:
8         return 0
9
10    result *= digit
11
12    user_input = input()
13
14    print(multiply_digits(user_input))
```

PYTHON

More Information

Question 1 of 2

What happens if the input to the `multiply_digits` function contains the digit 0?

```
1 def multiply_digits(num):
2     result = 1
3
4     for char in num:
5         digit = int(char)
6
7         if digit == 0:
8             return 0
9
10        result *= digit
```

PYTHON

The function returns 1
The function returns the product of all digits except 0
The function returns 0

Snipping Tool

Screenshot copied to clipboard and saved
Select here to mark up and share the image

84°F Mostly cloudy

Search

L

12:48 13-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT
wAVe™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

More Information

Explanation:

1. Iteration Through Digits:

```
1 def multiply_digits(num):
2     result = 1
3
4     for char in num:
```

The code defines a function `multiply_digits` that takes a parameter `num`. Inside the function, there's a for loop that iterates through each character(`char`) in the input `num`.

2. Checking for Zero:

```
1 if digit == 0:
2     return 0
```

Within the loop, each character is converted to an integer. There's a conditional check to see if the digit is 0. If it is, the function immediately returns 0.

3. Multiplying Digits:

```
1 result *= digit
```

For each digit, it is multiplied with the `result` variable. After processing all digits, the function returns the final result.

Question 1 of 2

Complete the MCQs to proceed Continue

Search

18:13 13-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT wAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. Write a Python function to calculate the total population of all cities.

```
cities=[["Pittsburgh","Allegheny",3049384],  
       ["philadelphia","philadelphia",35361436],  
       ["denmark","Erie",836453]]
```

```
1- def total_population(cities):  
2-     population = 0  
3-  
4-     for res in cities:  
5-         population +=res[2]  
6-  
7-     print(population)  
8-  
9-  
10-    cities=[["Pittsburgh","Allegheny",3049384],
```

PYTHON

Explanation:

In the `total_population` function, there is a for loop that iterates through each city in the input list `(cities)`. For each city `(res)`, the population value (the third element in the sublist) is extracted and added to the variable `population`.

Question 1 of 3

What does the function `total_population` in the given Python code do?

```
1- def total_population(cities):
```

PYTHON

Complete the MCQs to proceed Continue

Search

18:14 13-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT
wAVe™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. Write a Python function to calculate the total population of all cities.

```
cities=[["Pittsburgh","Allegheny",3049384],  
       ["philadelphia","philadelphia",35361436],  
       ["denmark","Erie",836453]]
```

PYTHON

```
5     population +=res[2]  
6  
7     print(population)  
8  
9  
10    cities=[["Pittsburgh","Allegheny",3049384],  
11        ["philadelphia","philadelphia",35361436],  
12        ["denmark","Erie",836453]]  
13  
14    total_population(cities)
```

Explanation:

In the `total_population` function, there is a for loop that iterates through each city in the input list `(cities)`. For each city `(res)`, the population value (the third element in the sublist) is extracted and added to the variable `population`.

Question 1 of 3

What does the function `total_population` in the given Python code do?

```
1. def total_population(cities):  
2.     population = 0  
3.  
4.     for res in cities:  
5.         population +=res[2]
```

PYTHON

Complete the MCQs to proceed Continue

Search

18:14 13-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT
wAVe™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. Write a program to print this pattern

```
1
3 5
7 9 11
```

PYTHON

```
1 n = int(input())
2
3 num = 1
4
5 for row in range(1, n+1):
6     each_row = ""
7     for number in range(1, row+1):
8         each_row = each_row + (str(num) + " ")
9     num+=2
10
```

▼ More Information

Explanation:

- User Input:

```
1 n = int(input())
```

PYTHON
- Outer Loop:

```
1 for row in range(1, n+1):
2     each_row = ""
```

PYTHON

The code takes an integer input `(n)` from the user.

Continue

31°C Partly sunny

Search

18:14 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT
WAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. Write a program to print this pattern

```
1
3 5
7 9 11
```

PYTHON

```
2
3 num = 1
4
5 for row in range(1, n+1):
6     each_row = ""
7     for number in range(1, row+1):
8         each_row = each_row + (str(num) + " ")
9     num+=2
10
11 print(each_row)
```

▼ More Information

Explanation:

- User Input:

```
1 n = int(input())
```

The code takes an integer input (n) from the user.

- Outer Loop:

```
1 for row in range(1, n+1):
2     each_row = ""
```

Continue

31°C Partly sunny

Search

Cloud, Search, Mask, Photo, Video, Google Chrome, Microsoft Edge, File Explorer, LinkedIn, Spotify

ENG IN 18:14 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT
wAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

10
11 print(each_row)

▼ More Information

Explanation:

1. User Input:

```
1 n = int(input())
```

The code takes an integer input (`n`) from the user.

2. Outer Loop:

```
1 for row in range(1, n+1):
 2     each_row = ""
```

The outer loop iterates through each row from 1 to `n` (inclusive) and it has variable `each_row` which is initialized to an empty string for each row.

3. Inner Loop:

```
1     for number in range(1, row+1):
 2         each_row = each_row + (str(num) + " ")
 3         num+=2
```

The inner loop iterates through each number in the current row, starting from 1 up to the row number. `each_row` is concatenating the current value of `num` (which starts at 1) to the row string and then increments `num` by 2 for the next iteration.

Ask doubt Continue

31°C Sunset

Search

18:15 13-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT
wAVe™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. Write a program to print the below pattern

```
1
2 3
4 5 6
```

```
1 n = int(input())
2
3 num = 1
4
5 for row in range(1, n+1):
6     each_row = ""
7     for number in range(1, row+1):
8         each_row = each_row + (str(num) + " ")
9     num+=1
10
```

PYTHON

► More Information

Continue

Upcoming Earnings

Search

18:15 13-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT
wAVe™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. Write a program to print the below pattern

```
1
2 3
4 5 6
```

```
2
3 num = 1
4
5 for row in range(1, n+1):
6     each_row = ""
7     for number in range(1, row+1):
8         each_row = each_row + (str(num) + " ")
9         num+=1
10    print(each_row)
```

PYTHON

► More Information

Continue

Upcoming Earnings

Search

18:16 13-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT
wAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

8 each_row = each_row + (str(num) + " ")
9 num+=1
10 print(each_row)

▼ More Information

Explanation:

1. User Input:

```
1 n = int(input())
```

The code takes an integer input (n) from the user.

2. Outer Loop:

```
1 for row in range(1, n+1):  
2     each_row = ""
```

The outer loop iterates through each row from 1 to n (inclusive) and it has variable each_row which is initialized to an empty string for each row.

3. Inner Loop:

```
1 for number in range(1, row+1):  
2     each_row = each_row + (str(num) + " ")  
3     num+=1
```

The inner loop iterates through each number in the current row, starting from 1 up to the row number. each_row is concatenating the current value of num (which starts by 1 for the next iteration).

Snipping Tool

Screenshot copied to clipboard and saved
Select here to mark up and share the image

31°C Partly sunny

Search

18:16 13-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT
wAVe™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. Write a Python program to print a triangle pyramid.

PYTHON

```
1 def print_pyramid(rows):
2     for i in range(1, rows + 1):
3         print(" " * (rows - i) + "*" * (2 * i - 1))
4
5 rows = int(input())
6 print_pyramid(rows)
```

Explanation:

In this Python code, the for loop iterates from 1 to the specified number of rows. Within each iteration of the loop, the code first prints leading spaces creating the left alignment of the pyramid. After printing the spaces, the code prints * characters in an increasing pattern, with the number of * characters being $2 * i - 1$ for each row i . This creates the pyramid pattern.

Continue

Miramal Tank Ro... Construction

Search

18:16 13-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT
wAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. Write a Python program to print a triangle pyramid.

PYTHON

```
1 - def print_pyramid(rows):
2 -     for i in range(1, rows + 1):
3 -         print(" " * (rows - i) + "*" * (2 * i - 1))
4 -
5 - rows = int(input())
6 - print_pyramid(rows)
```

Explanation:

In this Python code, the for loop iterates from 1 to the specified number of rows. Within each iteration of the loop, the code first prints leading spaces creating the left alignment of the pyramid. After printing the spaces, the code prints * characters in an increasing pattern, with the number of * characters being $2 \times i - 1$ for each row i . This creates the pyramid pattern.

Question 1 of 2

What will be the output after executing the following code?

PYTHON

```
1 - def print_pyramid(rows):
2 -     for i in range(1, rows + 1):
3 -         print(" " * (rows - i) + "*" * (2
4 -
5 - rows = 1
6 - print_pyramid(rows)
```

Complete the MCQs to proceed Continue

31°C Partly sunny

Search

18:17 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT wAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions

2 hours, 10 mins

Introduction to Python Reading Material 25 mins

Conditional Statements and Loops Reading Material 15 mins

List and Functions Reading Material 25 mins

Tuples, Sets and Dictionaries Reading Material 25 mins

OOPs Reading Material 20 mins

Miscellaneous Topics Reading Material 20 mins

More Interview Questions 20 mins

Q. Write a function to return the longest word in the sentence.

Companies Asked: Translytics Business Services

There are multiple ways to solve the problem of finding the longest word in a sentence:

1. Using a Loop

```
1 def find_longest_word(sentence):
2     words = sentence.split()
3     longest_word = ""
4
5     for word in words:
6         if len(word) > len(longest_word):
7             longest_word = word
8
9     return longest_word
10
11 sentence = "Cats quickly jump over fences"
```

PYTHON

After executing the above code, the output will be "quickly".

2. Using Built-in Functions

```
1 def find_longest_word(sentence):
2     words = sentence.split()
3     return max(words, key=len)
4
5 sentence = "Cats quickly jump over fences"
6 print(find_longest_word(sentence))
```

PYTHON

After executing the above code, the output will be "quickly".

3. Using Sorted List

```
1 def find_longest_word(sentence):
```

PYTHON

Continue

31°C Partly sunny

Search

18:18 13-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT
wAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. Write a function to return the longest word in the sentence.

Companies Asked: Translytics Business Services

There are multiple ways to solve the problem of finding the longest word in a sentence:

1. Using a Loop

```
2 words = sentence.split()
3 longest_word = ""
4
5 for word in words:
6     if len(word) > len(longest_word):
7         longest_word = word
8
9 return longest_word
10
11 sentence = "Cats quickly jump over fences"
12 print(find_longest_word(sentence))
```

PYTHON

After executing the above code, the output will be "quickly".

2. Using Built-in Functions

```
1 def find_longest_word(sentence):
2     words = sentence.split()
3     return max(words, key=len)
4
5 sentence = "Cats quickly jump over fences"
6 print(find_longest_word(sentence))
```

After executing the above code, the output will be "quickly".

3. Using Sorted List

```
1 def find_longest_word(sentence):
```

Snipping Tool

Screenshot copied to clipboard and saved
Select here to mark up and share the image

31°C Partly sunny

Search

18:18 13-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143dd74

NXT
wAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

After executing the above code, the output will be "quickly".

2. Using Built-in Functions

PYTHON

```
1 sentence = "Cats quickly jump over fences"
2 print(find_longest_word(sentence))
```

After executing the above code, the output will be "quickly".

3. Using Sorted List

PYTHON

```
1 def find_longest_word(sentence):
2     words = sentence.split()
3     sorted_words = sorted(words, key=len)
4     return sorted_words[-1]
5
6 sentence = "Cats quickly jump over fences"
7 print(find_longest_word(sentence))
```

After executing the above code, the output will be "quickly".

Continue

Ask doubt

Notes Discussions

31°C Partly sunny

Search

18:18 13-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143d...

NXT
wAV™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

What are data structures?

Companies Asked: Fretron

Data structures are fundamental tools in computer science used to organize and manage data efficiently within computer programs. They dictate how data is stored, accessed, and manipulated, enabling optimized algorithm performance. Examples include arrays, linked lists, and trees.

Example: An array is a linear data structure that stores elements of the same data type in contiguous memory locations. In the context of the contact management application, you can use an array to store the contact information of multiple individuals. Each element of the array represents a person's contact details.

```
1 contacts = []
2 [
3     {
4         'name': 'John Smith',
5         'phone': '123-456-7890',
6         'email': 'john@example.com'
7     },
8     {
9         'name': 'Alice Johnson',
10        'phone': '987-654-3210',
11        'email': 'alice@example.com'
12    }
13]
```

PYTHON

Continue

31°C Partly sunny

Search

Cloud, Search, Mask, Photo, Video, Google Chrome, Microsoft Edge, File Explorer, Spotify, Volume, Network, ENG IN, 18:19, 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143d...

NXT WAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPS Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

What are data structures?

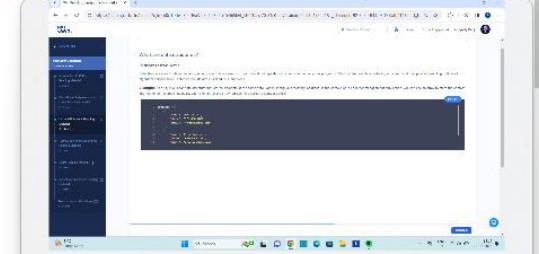
Companies Asked: Fretron

Data structures are fundamental tools in computer science used to organize and manage data efficiently within computer programs. They dictate how data is stored, accessed, and manipulated, enabling optimized algorithm performance. Examples include arrays, linked lists, and trees.

Example: An array is a linear data structure that stores elements of the same data type in contiguous memory locations. In the context of the contact management application, you can use an array to store the contact information of multiple individuals. Each element of the array represents a person's contact details.

```
4     'phone': '123-456-7890',
5     'email': 'john@example.com'
6 },
7 [
8     {
9         'name': 'Alice Johnson',
10        'phone': '987-654-3210',
11        'email': 'alice@example.com'
12    },
13 ] # More contacts can be added here
```

PYTHON



Snipping Tool

Screenshot copied to clipboard and saved
Select here to mark up and share the image

31°C Partly sunny

Search

Cloud, Search, Mask, Photo, Video, Google Chrome, Microsoft Edge, File Explorer, LinkedIn, Spotify

ENG IN 18:19 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143d...

NXT
wAVe™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. Given an array of integers nums and an integer target, return indices of the two numbers such that they add up to target. You may assume that each input would have exactly one solution, and you may not use the same element twice.

Companies Asked: Reyans Digital

Two Sum Leet Code

There are multiple ways to solve this problem:

1. Using Loops

PYTHON

```
1 - def two_sum(nums, target):
2 -     for i in range(len(nums)):
3 -         for j in range(i + 1, len(nums)):
4 -             if nums[i] + nums[j] == target:
5 -                 return [i, j]
6 -
7 - print(two_sum([2, 7, 11, 15], 9))
```

After executing the above code, the output will be [0, 1].

2. Using Dictionary

PYTHON

```
1 - def two_sum(nums, target):
2 -     seen = {}
3 -     for i in range(len(nums)):
4 -         complement = target - nums[i]
5 -         if complement in seen:
6 -             return [seen[complement], i]
7 -         seen[nums[i]] = i
8 -
9 - print(two_sum([2, 7, 11, 15], 9))
```

After executing the above code, the output will be [0, 1].

Continue

Ask doubt

31°C Partly sunny

Search

18:20 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143d...

NXT wAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

After executing the above code, the output will be [0, 1].

3. Using Recursions

PYTHON

```
1 def two_sum_recursive(nums, target, start=0):
2     if len(nums) < 2:
3         return None
4
5     for i in range(len(nums)):
6         for j in range(i + 1, len(nums)):
7             if nums[i] + nums[j] == target:
8                 return [start + i, start + j]
9
10    mid = len(nums) // 2
```

After executing the above code, the output will be [0, 1].

Continue

Ask doubt

Notes Discussions

+ NEW NOTE

B I U S E E T

Title Take a Note

CANCEL SAVE

31°C Partly sunny

Search

Cloud, Search, Mask, File, Google Chrome, Microsoft Edge, Microsoft Word, Microsoft Excel, Spotify, L, 18:21, ENG IN, 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143d...

NXT wAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

9 print(two_sum([2, 7, 11, 15], 9))

After executing the above code, the output will be [0, 1].

3. Using Recursions

PYTHON

```
10     mid = len(nums) // 2
11     left_result = two_sum_recursive(nums[:mid], target, start)
12     right_result = two_sum_recursive(nums[mid:], target, start + mid)
13
14     if left_result:
15         return left_result
16     else:
17         return right_result
18
19 print(two_sum_recursive([2, 7, 11, 15], 9))
```

After executing the above code, the output will be [0, 1].

Continue

Ask doubt

Notes Discussions

+ NEW NOTE

B I U S E E T

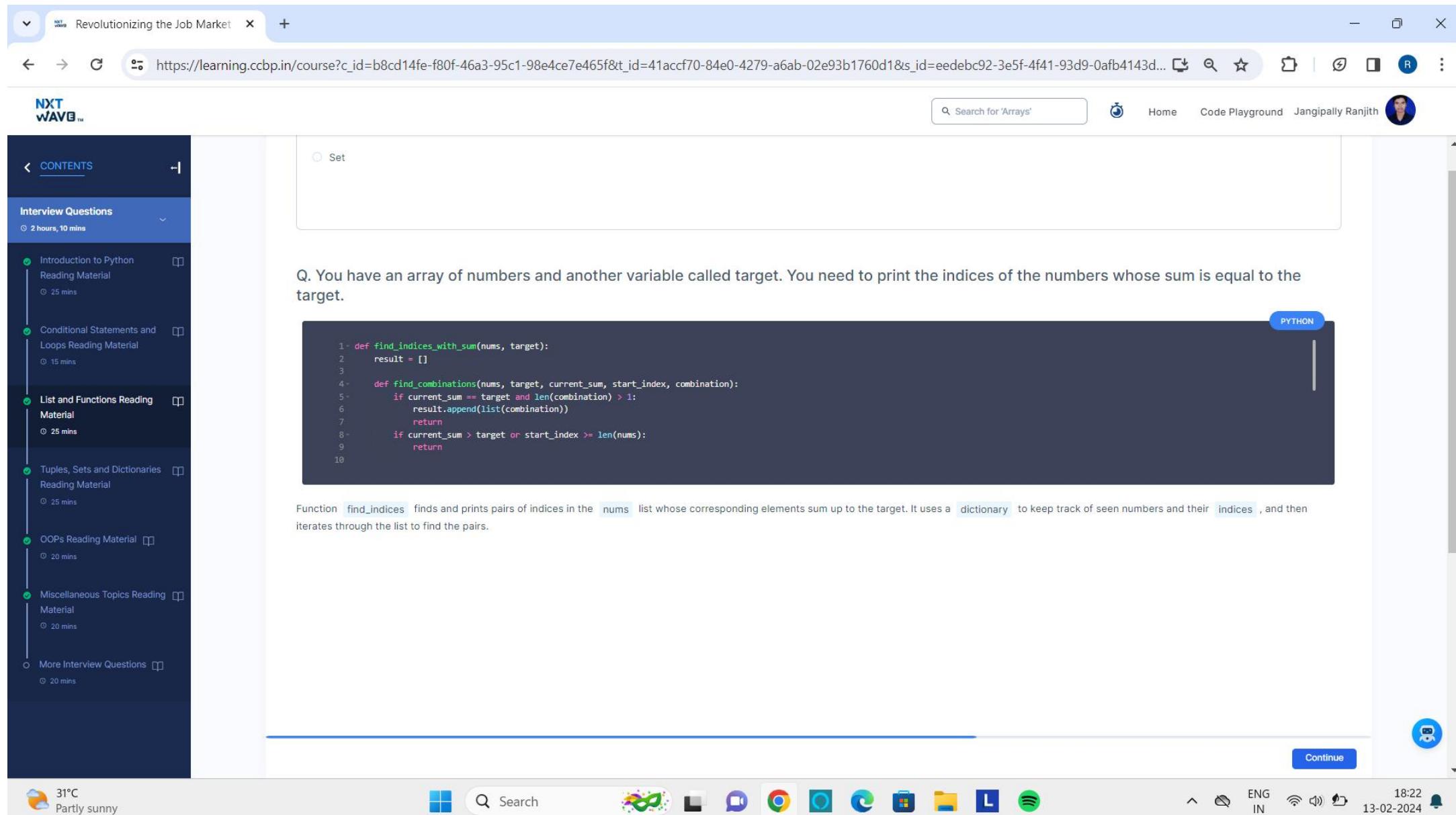
Title Take a Note

CANCEL SAVE

31°C Partly sunny

Search

18:21 13-02-2024 ENG IN



Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143d...

NXT
wAVG™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Set

Q. You have an array of numbers and another variable called target. You need to print the indices of the numbers whose sum is equal to the target.

PYTHON

```
10
11     combination.append(start_index)
12     find_combinations(nums, target, current_sum + nums[start_index], start_index + 1, combination)
13
14     combination.pop()
15     find_combinations(nums, target, current_sum, start_index + 1, combination)
16
17     find_combinations(nums, target, 0, 0, [])
18
19     return result
20
```

Function `find_indices` finds and prints pairs of indices in the `nums` list whose corresponding elements sum up to the target. It uses a `dictionary` to keep track of seen numbers and their `indices`, and then iterates through the list to find the pairs.

Continue

31°C Partly sunny

Search

18:22 13-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143d...

NXT
wAVe™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Set

Q. You have an array of numbers and another variable called target. You need to print the indices of the numbers whose sum is equal to the target.

PYTHON

```
16     find_combinations(nums, target, 0, 0, [])
17
18     return result
19
20
21     nums = [1, 2, 3, -4, 5]
22     target = 4
23     results = find_indices_with_sum(nums, target)
24     for res in results:
25         print(*res)
```

Function `find_indices` finds and prints pairs of indices in the `nums` list whose corresponding elements sum up to the target. It uses a `dictionary` to keep track of so iterates through the list to find the pairs.

Snipping Tool

Screenshot copied to clipboard and saved Select here to mark up and share the image

31°C Partly sunny

Search

18:22 13-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143d...       

NXT
wAVe™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith 

CONTENTS

Interview Questions 

Introduction to Python Reading Material 
25 mins

Conditional Statements and Loops Reading Material 
15 mins

List and Functions Reading Material 
25 mins

Tuples, Sets and Dictionaries Reading Material 
25 mins

OOPs Reading Material 
20 mins

Miscellaneous Topics Reading Material 
20 mins

More Interview Questions 
20 mins

Q. You are given a list of intervals, where each interval is represented by a pair of start and end times. Your task is to merge overlapping intervals. Write a function `merge_intervals(intervals)` that takes a list of intervals as input and returns a new list of merged intervals.

Input:

```
[(1,3),(2,6),(17,20),(8,10),(15,18)]**
```

Output:

```
[(1,6),(8,10),(15,20)]**
```

PYTHON

```
1- def merge_intervals(intervals):
2-     if len(intervals) <= 1:
3-         return intervals
4-
5-     intervals.sort()
6-
7-     merged_intervals = []
8-     current_interval = intervals[0]
9-
10-    for interval in intervals[1:]:
```

► More Information

Continue 

31°C Partly sunny

Search              

ENG IN    18:23 13-02-2024 

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143d...

NXT
wAVe™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. You are given a list of intervals, where each interval is represented by a pair of start and end times. Your task is to merge overlapping intervals. Write a function `merge_intervals(intervals)` that takes a list of intervals as input and returns a new list of merged intervals.

Input:

```
[(1,3),(2,6),(17,20),(8,10),(15,18)]**
```

Output:

```
[(1,6),(8,10),(15,20)]**
```

PYTHON

```
10 -     for interval in intervals[1:]:
11 -         if interval[0] <= current_interval[1]:
12 -             current_interval = (current_interval[0], max(current_interval[1], interval[1]))
13 -         else:
14 -             merged_intervals.append(current_interval)
15 -             current_interval = interval
16 -
17 -     merged_intervals.append(current_interval)
18 -
19 - return merged_intervals
```

► More Information

Continue

31°C Partly sunny

Search

18:23 13-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143d...

NXT
wAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPS Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. You are given a list of intervals, where each interval is represented by a pair of start and end times. Your task is to merge overlapping intervals. Write a function `merge_intervals(intervals)` that takes a list of intervals as input and returns a new list of merged intervals.

Input:

```
[(1,3),(2,6),(17,20),(8,10),(15,18)]**
```

Output:

```
[(1,6),(8,10),(15,20)]**
```

PYTHON

```
14     merged_intervals.append(current_interval)
15     current_interval = interval
16
17     merged_intervals.append(current_interval)
18
19     return merged_intervals
20
21 intervals = [(1, 3), (2, 6), (17, 20), (8, 10), (15, 18)]
22 merged_intervals = merge_intervals(intervals)
23 print(merged_intervals)
```

More Information

Snipping Tool

Screenshot copied to clipboard and saved Select here to mark up and share the image

31°C Partly sunny

Search

18:23 13-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143d...

NXT wAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPS Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Output:

```
[(1,3),(2,6),(17,20),(8,10),(15,18)]**
```

```
[(1,6),(8,10),(15,20)]**
```

```
14     merged_intervals.append(current_interval)
15     current_interval = interval
16
17     merged_intervals.append(current_interval)
18
19     return merged_intervals
20
21 intervals = [(1, 3), (2, 6), (17, 20), (8, 10), (15, 18)]
22 merged_intervals = merge_intervals(intervals)
23 print(merged_intervals)
```

PYTHON

More Information

The `merge_intervals` function combines overlapping intervals into larger intervals that represent the entire range of time covered.

- Interval `(1, 3)` overlaps with interval `(2, 6)` because the end time of `(1, 3)` is greater than the start time of `(2, 6)`.
- Interval `(8, 10)` is not overlapping with any other intervals.
- Interval `(15, 18)` overlaps with interval `(17, 20)` because the end time of `(15, 18)` is greater than the start time of `(17, 20)`.

Ask doubt

Notes Discussions

31°C Partly sunny

Search

Snipping Tool

Screenshot copied to clipboard and saved Select here to mark up and share the image

18:23 13-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143d...

NXT WAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions (2 hours, 10 mins)

- Introduction to Python Reading Material (25 mins)
- Conditional Statements and Loops Reading Material (15 mins)
- List and Functions Reading Material (25 mins)
- Tuples, Sets and Dictionaries Reading Material (25 mins)
- OOPs Reading Material (20 mins)
- Miscellaneous Topics Reading Material (20 mins)
- More Interview Questions (20 mins)

None of the given options

Q. What is difference between Remove and del functions in python?

Companies Asked: Bluecloud

In Python, `remove` and `del` are used to remove items from a list, but they do so in slightly different ways:

- `remove` is a method that is used to remove the first occurrence of a specified value from a list. If the specified value is not found, it raises a `ValueError`. For example:

```
1 list = [1, 2, 3, 2]
2 list.remove(2)
```

2. `del` is a statement that is used to delete an item at a specific index from a list or to delete a slice of a list. If the specified index does not exist, it raises an `IndexError`. It can also be used to delete variables. For example:

```
1 list = [1, 2, 3, 2]
2 del list[1]
```

It is recommended to use `remove` when you want to remove an item by its value and `del` when you want to remove an item by its index.

Continue

31°C Partly sunny

Search

18:24 13-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143d...

NXT
WAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. What are the methods for lists in Python?

Companies Asked: Bluecloud

There are several methods associated with lists in Python. Some of the commonly used ones include:

- `append()` : Adds an element to the end of the list.
- `extend()` : Adds multiple elements to the end of the list.
- `insert()` : Inserts an element at a specified position.
- `remove()` : Removes the first occurrence of a specified element.
- `pop()` : Removes and returns an element from a specified position.
- `index()` : Returns the index of the first occurrence of a specified element.
- `count()` : Returns the number of occurrences of a specified element.
- `sort()` : Sorts the list.
- `reverse()` : Reverses the order of the list.
- `clear()` : Removes all elements from the list.

These methods provide functionality to manipulate and query list data structures in Python.

Continue

31°C Partly sunny

Search

18:24 13-02-2024 ENG IN

Q. What is the difference between remove and pop method in Python?

Remove	Pop
The <code>remove()</code> method is used to remove the first occurrence of a specified element from the list.	The <code>pop()</code> method is used to remove and return an element from a specific index in the list.
It requires the actual value of the element to be removed as an argument.	It takes an optional index as an argument. If the index is not provided, it removes and returns the last element in the list.
If the specified element is not found, it raises a <code>ValueError</code> .	If the specified index is out of range, it raises an <code>IndexError</code> .
It does not return the removed element; it only modifies the original list in place.	The removed element is returned, allowing you to capture and use its value.

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143d...

NXT
wAVG™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. Write a program to print the frequencies of elements in an array.

Companies Asked: Fretron

Here are two alternative approaches to printing the frequencies of elements in an array in Python:

1. Using Dictionary:

```
1 # Initialize an array
2 my_array = [1, 2, 2, 3, 1, 4, 2, 3, 5]
3
4 # Create an empty dictionary to store frequencies
5 element_freq = {}
6
7 # Loop through the array
8 for element in my_array:
9     # Check if the element is in the dictionary
10    if element in element_freq:
```

2. Using Set:

```
1 def print_element_frequencies(arr):
2     unique_elements = set(arr)
3
4     for element in unique_elements:
5         count = arr.count(element)
6         print(f"{element}: {count}")
7
8     # Example usage:
9 my_array = [1, 2, 2, 3, 3, 3]
10 print_element_frequencies(my_array)
```

Continue

31°C Partly sunny

Search

18:26 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143d...

NXT
wAVG™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. Write a program to print the frequencies of elements in an array.

Companies Asked: Fretron

Here are two alternative approaches to printing the frequencies of elements in an array in Python:

1. Using Dictionary:

```
10 -     if element in element_freq:  
11 -         # If it is, increment the count  
12 -         element_freq[element] += 1  
13 -     else:  
14 -         # If it's not, add it to the dictionary with a count of 1  
15 -         element_freq[element] = 1  
16 -  
17 - # Print the frequencies  
18 - for element, frequency in element_freq.items():  
19 -     print(f"Element {element} appears {frequency} times.")
```

2. Using Set:

```
1- def print_element_frequencies(arr):  
2-     unique_elements = set(arr)  
3-  
4-     for element in unique_elements:  
5-         count = arr.count(element)  
6-         print(f"{element}: {count}")  
7-  
8- # Example usage:  
9- my_array = [1, 2, 2, 3, 3, 3]  
10- print_element_frequencies(my_array)
```

Continue

31°C Partly sunny

Search

18:26 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143d...

NXT
wAVG™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions (2 hours, 10 mins)

- Introduction to Python Reading Material (25 mins)
- Conditional Statements and Loops Reading Material (15 mins)
- List and Functions Reading Material (25 mins)
- Tuples, Sets and Dictionaries Reading Material (25 mins)
- OOPs Reading Material (20 mins)
- Miscellaneous Topics Reading Material (20 mins)
- More Interview Questions (20 mins)

1 print(element_freq[2])

1 print(element_freq[2]) PYTHON

Q. What are the prerequisites for binary search?

Companies Asked: Fretron

The prerequisites for binary search are:

1. A sorted array or list.
2. A target element to search for.

Binary search works by repeatedly dividing the sorted array in half and comparing the middle element with the target element until a match is found or the search range is exhausted.

Continue

31°C Partly sunny

Search

18:27 13-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143d...

NXT WAVE™

A target element to search for only.

CONTENTS

Interview Questions (2 hours, 10 mins)

- Introduction to Python Reading Material (25 mins)
- Conditional Statements and Loops Reading Material (15 mins)
- List and Functions Reading Material (25 mins)
- Tuples, Sets and Dictionaries Reading Material (25 mins)
- OOPs Reading Material (20 mins)
- Miscellaneous Topics Reading Material (20 mins)
- More Interview Questions (20 mins)

Q. Sort an array without using built-in methods. PYTHON

```
1+ def sort_array(arr):
2+     n = len(arr)
3+
4+     for i in range(n - 1):
5+         for j in range(0, n - i - 1):
6+             if arr[j] > arr[j + 1]:
7+                 temp = arr[j]
8+                 arr[j] = arr[j + 1]
9+                 arr[j + 1] = temp
10+    ..
```

Explanation:

The **sort_array** function implements the **Bubble Sort** algorithm to arrange the elements of an input array `arr` in ascending order. It utilizes nested loops to compare adjacent elements, swapping them if the current element is greater than its adjacent right element. The process is repeated until the entire array is sorted. The outer loop iterates `n-1` times, where `n` is the array length, and the inner loop compares and swaps elements, step by step moving the largest unsorted element to its correct position.

Continue

31°C Partly sunny

Search

18:28 13-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143d...

NXT
wAVG™

A target element to search for only.

CONTENTS

Interview Questions

2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. Sort an array without using built-in methods.

PYTHON

```
8         arr[j] = arr[j + 1]
9         arr[j + 1] = temp
10
11
12 input_str = input()
13 user_array = [int(x) for x in input_str.split()]
14
15 sort_array(user_array)
16
17 print(user_array)
```

Explanation:

The **sort_array** function implements the **Bubble Sort** algorithm to arrange the elements of an input array `arr` in ascending order. It utilizes nested loops to compare adjacent elements. If the current element is greater than its adjacent right element, they are swapped. This process is repeated until the entire array is sorted. The outer loop iterates `n-1` times, where `n` is the array length. The inner loop compares adjacent elements, step by step moving the largest unsorted element to its correct position.

Snipping Tool

Screenshot copied to clipboard and saved
Select here to mark up and share the image

31°C Partly sunny

Search

18:28 13-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143d...

NXT wAVES™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions (2 hours, 10 mins)

- Introduction to Python (Reading Material, 25 mins)
- Conditional Statements and Loops (Reading Material, 15 mins)
- List and Functions (Reading Material, 25 mins)
- Tuples, Sets and Dictionaries (Reading Material, 25 mins)
- OOPs (Reading Material, 20 mins)
- Miscellaneous Topics (Reading Material, 20 mins)
- More Interview Questions (20 mins)

Q. Sort an array without using built-in methods.

PYTHON

```
8         arr[j] = arr[j + 1]
9         arr[j + 1] = temp
10
11
12 input_str = input()
13 user_array = [int(x) for x in input_str.split()]
14
15 sort_array(user_array)
16
17 print(user_array)
```

Explanation:

The `sort_array` function implements the **Bubble Sort** algorithm to arrange the elements of an input array `arr` in ascending order. It utilizes nested loops to compare adjacent elements, swapping them if the current element is greater than its adjacent right element. The process is repeated until the entire array is sorted. The outer loop iterates `n-1` times, where `n` is the array length, and the inner loop compares and swaps elements, step by step moving the largest unsorted element to its correct position.

Continue

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143d...

NXT
WAVES™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions (2 hours, 10 mins)

- Introduction to Python Reading Material (25 mins)
- Conditional Statements and Loops Reading Material (15 mins)
- List and Functions Reading Material (25 mins)
- Tuples, Sets and Dictionaries Reading Material (25 mins)
- OOPs Reading Material (20 mins)
- Miscellaneous Topics Reading Material (20 mins)
- More Interview Questions (20 mins)

Q. Explain the approach of binary search.

Companies Asked: Fretron

Binary search is an efficient algorithm for finding a target value within a sorted array. The approach starts by comparing the middle element of the array to the target value. If they match, the target's index is returned. If the target is less than the middle element, the search continues on the left half of the array. If the target is greater, the search moves to the right half. This process is repeated, narrowing the range of possible locations until the target is found or the range is empty.

For example, let's say we have a sorted list of numbers: [1, 3, 5, 7, 9, 11, 13] and the target value is 5.

- First, we compare 5 to the middle item, which is 7. Since 5 is smaller than 7, we focus on the left half: [1, 3, 5]
- In this new list, we compare 5 to the middle item, which is 3. Since 5 is larger than 3, we focus on the right half: [5]
- Now, let's compare 5 to the only item left, which is also 5. Since they match, it returns the target's index, which is 2.

Continue

31°C Partly sunny

Search

18:29 13-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143d...

NXT
wAVG™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

The search terminates, and the target is considered not in the array.

The middle element is immediately returned as the target.

Q. Write a program to check if the given string is present in the array of strings. If string is present return the index. If the string is not present "The string is not present in the array"?

Intent of the interviewer: Interviewer is expecting user inputs strings.

Solution:

```
1: def find_string_in_array(target, array):
2:
3:     indices = []
4:     i = 0
5:     while i < len(array):
6:         if array[i] == target:
7:             indices.append(i)
8:             i += 1
9:         else:
10:            i += 1
```

PYTHON

For each string, check if the string is equal to the target string. If the string matches the target, the position (index) of that string within the array is added to the `indices` list.

Continue

31°C Partly sunny

Search

18:30 13-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143d...

NXT
wAV™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions (2 hours, 10 mins)

- Introduction to Python Reading Material (25 mins)
- Conditional Statements and Loops Reading Material (15 mins)
- List and Functions Reading Material (25 mins)
- Tuples, Sets and Dictionaries Reading Material (25 mins)
- OOPs Reading Material (20 mins)
- Miscellaneous Topics Reading Material (20 mins)
- More Interview Questions (20 mins)

The middle element is immediately returned as the target.

Q. Write a program to check if the given string is present in the array of strings. If string is present return the index. If the string is not present "The string is not present in the array"?

Intent of the interviewer: Interviewer is expecting user inputs strings.

Solution:

PYTHON

```
11
12     return indices
13
14 array_of_strings = input().split(",")
15 target_string = input()
16
17 result = find_string_in_array(target_string, array_of_strings)
18
19 if result:
20     print(result)
```

For each string, check if the string is equal to the target string. If the string matches the target, the position (index) of that string within the array is added to the `indices` list.

Continue

31°C Partly sunny

Search

18:30 13-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143d...

NXT
WAVES™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

The middle element is immediately returned as the target.

Q. Write a program to check if the given string is present in the array of strings. If string is present return the index. If the string is not present "The string is not present in the array".

Intent of the interviewer: Interviewer is expecting user inputs strings.

Solution:

PYTHON

```
13
14 array_of_strings = input().split(",")
15 target_string = input()
16
17 result = find_string_in_array(target_string, array_of_strings)
18
19 if result:
20     print(result)
21 else:
22     print("The string is not present in the array")
```

For each string, check if the string is equal to the target string. If the string matches the target, the position (index) of that string within the array is added to the indices

Snipping Tool

Screenshot copied to clipboard and saved
Select here to mark up and share the image

31°C Partly sunny

Search

18:30 13-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143d...

NXT
WAVES™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. Write code to print the prime number?

PYTHON

```
1  nums = [2, 3, 4, 5, 6, 7, 8, 9, 10, 11]
2
3- def is_prime(number):
4-     if number > 1:
5-         for i in range(2, int(number / 2) + 1):
6-             if (number % i) == 0:
7-                 return False
8-             return True
9-         else:
10            return False
```

Checks if a number is prime by iterating from 2 to `num/2` of the number and checking if it is divisible by any number in that range. If it is, then the number is not prime.

Continue

31°C Partly sunny

Search

18:31 13-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143d...

NXT
WAVES™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions (2 hours, 10 mins)

- Introduction to Python Reading Material (25 mins)
- Conditional Statements and Loops Reading Material (15 mins)
- List and Functions Reading Material (25 mins)
- Tuples, Sets and Dictionaries Reading Material (25 mins)
- OOPs Reading Material (20 mins)
- Miscellaneous Topics Reading Material (20 mins)
- More Interview Questions (20 mins)

Q. Write code to print the prime number?

PYTHON

```
5-     for i in range(2, int(number / 2) + 1):
6-         if (number % i) == 0:
7-             return False
8-         else:
9-             return True
10-
11
12- for num in nums:
13-     if is_prime(num):
14-         print(num, "is a prime number")
```

Checks if a number is prime by iterating from 2 to `num/2` of the number and checking if it is divisible by any number in that range. If it is, then the number is not prime.

Snipping Tool

Screenshot copied to clipboard and saved
Select here to mark up and share the image

31°C Partly sunny

Search

18:31 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143d...

NXT
WAVES™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. How to reverse a string?

Rephrased Question: Write code to reverse the string.

A string can be reversed using extended slicing.

Syntax:

```
variable[start:end:negative_step]
```

-1 for step will reverse the order of the characters.

Code

```
1 string_1 = "Program"
2 string_2 = string_1[::-1]
3 print(string_2)
```

PYTHON

Continue

31°C Partly sunny

Search

18:31 13-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143d...

NXT
WAVES™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions (2 hours, 10 mins)

- Introduction to Python (25 mins)
- Conditional Statements and Loops (15 mins)
- List and Functions (25 mins)
- Tuples, Sets and Dictionaries (25 mins)
- OOPs (20 mins)
- Miscellaneous Topics (20 mins)
- More Interview Questions (20 mins)

Q. Write code to reverse the string without using `[::-1]`?

Intent of the interviewer: The interviewer is expecting not to use slicing method i.e. [::-1]

- We can convert the string to list and use list method to reverse the string.

PYTHON

```
1+ def reverse_string(str1):
2+     str1 = list(str1)
3+     str1.reverse()
4+     result = ''.join(str1)
5+     return result
6+
7 str1 = "hello"
8 print(reverse_string(str1))
```

Convert the string into a list, reverse the list using the reverse() method, and then join the characters back into a string using the join() method with an empty string ("") as the separator.

► Another approach

Continue

31°C Partly sunny

Search

18:32 13-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143d...

NXT
WAVES™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPS Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. Write code to reverse the string without using `[::-1]` ?

Intent of the interviewer: The interviewer is expecting not to use slicing method i.e. [::-1]

- We can convert the string into a list, reverse the list using the `reverse()` method, and then join the characters back into a string using the `join()` method with an empty string ("") as the separator.

```
def reverse_string(str1):  
    str1 = list(str1)  
    str1.reverse()  
    result = ''.join(str1)  
    return result  
  
str1 = "hello"  
print(reverse_string(str1))
```

Convert the string into a list, reverse the list using the `reverse()` method, and then join the characters back into a string using the `join()` method with an empty string ("") as the separator.

Another approach

```
def reverse_string(str1):  
    result = ""  
    for i in range(len(str1)):  
        result += str1[len(str1) - i - 1]  
    return result  
  
str1 = "hello"  
print(reverse_string(str1))
```

The loop starts at the last character of the string, `len(str1) - 1`, and iterates backward until it reaches the first character, `-1`. The step value of `-1` ensures that the current character is appended to the result string.

Snipping Tool

Screenshot copied to clipboard and saved
Select here to mark up and share the image

31°C Partly sunny

Search

18:32 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143d...

NXT
wAVG™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions

Introduction to Python Reading Material 2 hours, 10 mins

Conditional Statements and Loops Reading Material 15 mins

List and Functions Reading Material 25 mins

Tuples, Sets and Dictionaries Reading Material 25 mins

OOPs Reading Material 20 mins

Miscellaneous Topics Reading Material 20 mins

More Interview Questions 20 mins

Q. What is the transpose of a matrix?

Transpose of a matrix is the interchanging of rows and columns. In other words, transpose of $A[0][0]$ is obtained by changing $A[i][j]$ to $A[j][i]$.

```
1 - def transpose_matrix(matrix):
2 -     transposed_matrix = []
3 -
4 -     for column_index in range(len(matrix[0])):
5 -         row = []
6 -         for Row_index in range(len(matrix)):
7 -             row.append(matrix[Row_index][column_index])
8 -         transposed_matrix.append(row)
9 -
10 -    return transposed_matrix
```

PYTHON

Question 1 of 2

Complete the MCQs to proceed Continue

31°C Partly sunny

Search

18:32 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143d...

NXT
WAVES™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions

Introduction to Python Reading Material 2 hours, 10 mins

Conditional Statements and Loops Reading Material 15 mins

List and Functions Reading Material 25 mins

Tuples, Sets and Dictionaries Reading Material 25 mins

OOPs Reading Material 20 mins

Miscellaneous Topics Reading Material 20 mins

More Interview Questions 20 mins

Q. What is the transpose of a matrix?

Transpose of a matrix is the interchanging of rows and columns. In other words, transpose of $A[0][0]$ is obtained by changing $A[i][j]$ to $A[j][i]$.

```
11
12 original_matrix = [
13     [1, 2, 3],
14     [4, 5, 6]
15 ]
16
17 transposed = transpose_matrix(original_matrix)
18
19 for row in transposed:
20     print(row)
```

PYTHON

Question 1 of 2

Complete the MCQs to proceed Continue

31°C Partly sunny

Search

18:33 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143d...

NXT
WAVES™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. Diagonal Matrix of 0 and 1?

A diagonal matrix is a square matrix whose non-diagonal entries are all equal to zero.

```
1
2 n = 3
3
4 diag_matrix = []
5 for row_index in range(n):
6     row = []
7     for column_index in range(n):
8         if row_index == column_index and row_index == 1:
9             row.append(1)
10        else:
```

PYTHON

Continue

31°C Partly sunny

Search

18:33 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=eedebc92-3e5f-4f41-93d9-0afb4143d...

NXT
wAVG™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. Diagonal Matrix of 0 and 1?

A diagonal matrix is a square matrix whose non-diagonal entries are all equal to zero.

```
6 row = []
7 for column_index in range(n):
8     if row_index == column_index and row_index == 1:
9         row.append(1)
10    else:
11        row.append(0)
12 diag_matrix.append(row)
13
14 for row in diag_matrix:
15     print(row)
```

PYTHON

Continue

31°C Partly sunny

Search

18:33 13-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fc58bfa6-14ab-4a80-aed6-d23dc8286f

NXT
WAVES™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

Interview Questions

Recently Asked Questions and Answers

1. What is a Tuple?

A tuple holds an ordered collection of items. A tuple is an immutable object, i.e. we cannot change the items of the tuple.

Creating a Tuple

- Created by enclosing elements within (round) brackets.
- Each item is separated by a comma.

Code

```
1 a = 2
2 tuple_a = (5, "Six", a, 8.2)
3 print(type(tuple_a))
4 print(tuple_a)
```

PYTHON

Output

```
<class 'tuple'>
(5, 'Six', 2, 8.2)
```

Tuple with a Single Item

Code

```
1 a = (1)
```

PYTHON

Continue

1 28°C Haze

Search

Cloud, Camera, Video, Google Chrome, Microsoft Edge, Microsoft Word, File, LinkedIn, Spotify

ENG IN 11:53 14-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fc58bfa6-14ab-4a80-aed6-d23dc8286f

NXT
wAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPS Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Tuple with a Single Item

Code

```
1 a = (1,)  
2 print(type(a))  
3 print(a)
```

PYTHON

Output

```
<class 'tuple'>  
(1,)
```

Accessing Tuple Elements

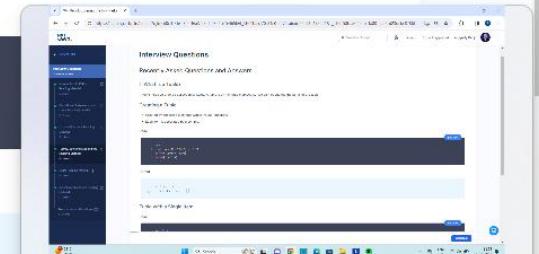
Accessing `tuple` elements is also similar to string and list accessing.

Code

```
1 a = 2  
2 tuple_a = (5, "Six", a, 8.2)  
3 print(tuple_a[1])
```

Output

```
Six
```



Snipping Tool

Screenshot copied to clipboard and saved
Select here to mark up and share the image

28°C Haze

Search

11:53 14-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fc58bfa6-14ab-4a80-aed6-d23dcbc8286f

NXT WAVE

CONTENTS

Interview Questions (2 hours, 10 mins)

- Introduction to Python (25 mins)
- Conditional Statements and Loops (15 mins)
- List and Functions (25 mins)
- Tuples, Sets and Dictionaries (25 mins)
- OOPs (20 mins)
- Miscellaneous Topics (20 mins)
- More Interview Questions (20 mins)

2. Are Tuples mutable?

No, Tuples are immutable. They cannot be modified.

Code (PYTHON)

```
1 tuple_a = (1, 2, 3, 5)
2 tuple_a[3] = 4
3 print(tuple_a)
```

Output

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

3. How to Unpack a tuple?

The values of any sequence can be directly assigned to variables. The number of variables on the left should match the length of the sequence.

Code (PYTHON)

```
1 tuple_a = ('R', 'e', 'd')
2 (s_1, s_2, s_3) = tuple_a
3 print(s_1)
4 print(s_2)
5 print(s_3)
```

Output

```
R
e
d
```

Continue

DOW -1.35%

Search

11:54 14-02-2024 ENG IN

Q. What is the difference between List and Tuple?

Basis for comparison	List	Tuple
Mutability	Lists are mutable	Tuples are immutable
Time Consumption	The list iteration is much slower compared to the tuple.	The tuple iteration is much faster compared to the list.
Usage	It is very helpful in the case of deletion and insertion operations.	It is comparatively helpful in the case of read-only operations, such as accessing elements.

← Question 1 of 3

Which of the following statements is correct regarding time consumption for iteration in lists and tuples?

- List iteration is faster than tuple iteration.
- Tuple iteration is slower than list iteration.
- List iteration is much slower than tuple iteration.
- Tuple and List both takes equal time.

Next



DOW
-1.35%



Search



ENG
IN

11:55
14-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fc58bfa6-14ab-4a80-aed6-d23dcbc8286f

- Read-only operations
- Deletion and insertion operations
- Both for read-only and modification operations
- Neither for read-only nor modification operations

Q. In tuples and lists which consumes more memory

Tuples consume less memory than lists in Python. This is because tuples are immutable and have a fixed size. Lists are mutable and need more memory for changes like adding or removing elements.

Question 1 of 3

Why do tuples generally consume less memory compared to lists?

- Because tuples are shorter.
- Because tuples are immutable and have a fixed size.
- Because tuples do not support methods like append and remove.
- Because tuples are less popular.

Next



28°C
Haze



Search



ENG
IN



11:56
14-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fc58bfa6-14ab-4a80-aed6-d23dcbc8286f

- Read-only operations
- Deletion and insertion operations
- Both for read-only and modification operations
- Neither for read-only nor modification operations

Q. In tuples and lists which consumes more memory

Tuples consume less memory than lists in Python. This is because tuples are immutable and have a fixed size. Lists are mutable and need more memory for changes like adding or removing elements.

← Question 3 of 3

Which data structure in Python can be modified after its creation by methods like `append` and `remove` ?

- List
- Tuple
- String
- All of the above



28°C
Haze



Search



ENG
IN



11:56
14-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fc58bfa6-14ab-4a80-aed6-d23dc8286f

NXT
WAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions (2 hours, 10 mins)

- Introduction to Python Reading Material (25 mins)
- Conditional Statements and Loops Reading Material (15 mins)
- List and Functions Reading Material (25 mins)
- Tuples, Sets and Dictionaries Reading Material (25 mins)
- OOPS Reading Material (20 mins)
- Miscellaneous Topics Reading Material (20 mins)
- More Interview Questions (20 mins)

Q. As tuples are immutable, how to combine them?

Since tuples are immutable in Python, we cannot directly modify or concatenate them. However, we can create a new tuple by combining elements from existing tuples. We can combine them:

- Using the + Operator:
- Using the tuple() Constructor:
- Using list() and extend() methods:

More information

- Using the + Operator:

```
tuple1 = (1, 2, 3)
tuple2 = (4, 5, 6)
combined_tuple = tuple1 + tuple2
```

PYTHON

- Using the tuple() Constructor:

```
tuple1 = (1, 2, 3)
tuple2 = (4, 5, 6)
combined_tuple = tuple(tuple1) + tuple(tuple2)
```

PYTHON

- Using list() and extend() methods:

```
tuple1 = (1, 2, 3)
tuple2 = (4, 5, 6)
x = list(tuple1)
y = list(tuple2)
x.extend(y)
combined_tuple = tuple(x)
```

PYTHON

Continue

SENSEX -0.76%

Search

11:57 14-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fc58bfa6-14ab-4a80-aed6-d23dc8286f

NXT
WAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions (2 hours, 10 mins)

- Introduction to Python Reading Material (25 mins)
- Conditional Statements and Loops Reading Material (15 mins)
- List and Functions Reading Material (25 mins)
- Tuples, Sets and Dictionaries Reading Material (25 mins)
- OOPs Reading Material (20 mins)
- Miscellaneous Topics Reading Material (20 mins)
- More Interview Questions (20 mins)

Q. Can we modify a tuple after it has been created in Python?

No, we cannot change the name of a tuple after it has been created. Tuples are immutable, which means that their values cannot be changed after they have been assigned. This includes the name of the tuple.

▼ More Information

- Create a new one with the desired name and copy values from the old tuple.

```
1 old_tuple = (1, 2, 3)
2 new_tuple = old_tuple
3 print(new_tuple) # Output: [1, 2, 3]
```

PYTHON

Question 1 of 2

Which of the following statements is true about tuple renaming?

- Tuples can be renamed using the rename function
- Tuples can be renamed by assigning them to a new tuple but the original tuple name remains the same (checked)
- Tuples can be renamed using the reassign function
- None of these

Next

Complete the MCQs to proceed Continue

SENSEX -0.76%

Search

11:57 14-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fc58bfa6-14ab-4a80-aed6-d23dc8286f

NXT WAVE

Interview Questions > Tuples, Sets and Dictionaries Reading Material

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions (2 hours, 10 mins)

- Introduction to Python Reading Material (25 mins)
- Conditional Statements and Loops Reading Material (15 mins)
- List and Functions Reading Material (25 mins)
- Tuples, Sets and Dictionaries Reading Material (25 mins)
- OOPs Reading Material (20 mins)
- Miscellaneous Topics Reading Material (20 mins)
- More Interview Questions (20 mins)

Q. What are Sets?

Rephrased Question: **What is a set and write its syntax in Python?**

In Python, a set is an unordered collection of unique elements, defined by enclosing elements with curly braces {}. Sets are used to store unique collections regardless of order where the duplicates are automatically removed.

Syntax:

```
1 my_set = {value1, value2, value3, ...}
```

More information

Code

```
1 a = 2
2 set_a = {5, "Six", a, 8.2}
3 print(type(set_a))
4 print(set_a)
```

Output

```
<class 'set'>
{8.2, 2, 'Six', 5}
```

Set items need not be in the same order as defined.

No Duplicate Items

Sets contain unique elements

Code

```
1 set_a = {"a", "b", "c", "a"}
2 print(set_a)
```

Output

```
{'b', 'a', 'c'}
```

SENSEX -0.76%

Search

11:58 14-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fc58bfa6-14ab-4a80-aed6-d23dc8286f

NXT WAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions (2 hours, 10 mins)

- Introduction to Python (25 mins)
- Conditional Statements and Loops (15 mins)
- List and Functions (25 mins)
- Tuples, Sets and Dictionaries (25 mins)
- OOPs (20 mins)
- Miscellaneous Topics (20 mins)
- More Interview Questions (20 mins)

Q. Is set ordered or unordered collection of items?

A set is an unordered collection of unique items. The elements in a set do not have a specific order or sequence, unlike lists or tuples. The order in which items are added to a set does not guarantee the order in which they are stored or retrieved.

← Question 1 of 3

What is a set in Python?

- An ordered collection of items.
- A collection of key-value pairs.
- An unordered collection of unique items.
- A sequence of mutable elements.

Next

Complete the MCQs to proceed Continue

1 28°C Haze

Search

11:58 14-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fc58bfa6-14ab-4a80-aed6-d23dc8286f

NXT WAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions (2 hours, 10 mins)

- Introduction to Python (25 mins)
- Conditional Statements and Loops (15 mins)
- List and Functions (25 mins)
- Tuples, Sets and Dictionaries (25 mins)
- OOPs (20 mins)
- Miscellaneous Topics (20 mins)
- More Interview Questions (20 mins)

Q. Is set ordered or unordered collection of items?

A set is an unordered collection of unique items. The elements in a set do not have a specific order or sequence, unlike lists or tuples. The order in which items are added to a set does not guarantee the order in which they are stored or retrieved.

← Question 2 of 3

What happens when you add duplicate items to a set?

- The set will keep all duplicates.
- The set will remove all previous occurrences of the item.
- The set will ignore the duplicates and only keep unique items.
- The set will raise an error.

Next

Complete the MCQs to proceed Continue

1 28°C Haze

Search

Clouds, Heart, Camera, Google Chrome, Microsoft Edge, Microsoft Word, File, LinkedIn, Spotify

ENG IN 11:59 14-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fc58bfa6-14ab-4a80-aed6-d23dc8286f

NXT WAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions (2 hours, 10 mins)

- Introduction to Python (25 mins)
- Conditional Statements and Loops (15 mins)
- List and Functions (25 mins)
- Tuples, Sets and Dictionaries (25 mins)
- OOPs (20 mins)
- Miscellaneous Topics (20 mins)
- More Interview Questions (20 mins)

Q. Can we access the data from set through indexing?

No, we cannot access the elements of a set in Python using indexing or slicing. This is because sets in Python are unordered collections of unique elements.

← Question 1 of 3

What will happen if you try to index a set in Python?

- It will return the first element.
- It will reorder the set and then return the indexed element.
- It will raise a **TypeError**.
- It will convert the set to a list and return the indexed element.

Next

Complete the MCQs to proceed Continue

1 28°C Haze

Search

11:59 14-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccgp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fc58bfa6-14ab-4a80-aed6-d23dc8286f

NXT WAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. How can we add the data to the set?

In Python, we can add elements to a set using the `add()` method for a single element or the `update()` method to add multiple elements.

- Here is an example of how to add an element to a set:

```
1 my_set = {1, 2, 3}
2 my_set.add(4)
3 print(my_set)
```

PYTHON

- Here is an example of how to update a set

```
1 my_set = {1, 2, 3}
2 my_set.update([4, 5])
3 print(my_set)
```

PYTHON

Question 1 of 3

How do you add a single element `5` to a set named `my_set` in Python?

`my_set.add(5)`

`my_set.append(5)`

`my_set.insert(5)`

Complete the MCQs to proceed Continue

NH163 / Assemb... Closed road

Search

12:00 14-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fc58bfa6-14ab-4a80-aed6-d23dc8286f

NXT WAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. What is a Dictionary?

Companies Asked: Uniserved

A Dictionary is an unordered collection of items. Every dictionary item is a **Key-Value** pair.

The diagram illustrates a dictionary structure with two entries. Each entry consists of a blue rectangular box labeled 'key_1' or 'key_2' containing a white box labeled 'value_1' or 'value_2'. An orange arrow points from each key to its corresponding value. Below the first entry, there are three dots (two vertical, one horizontal) indicating more entries. The entire diagram is enclosed in a thin orange border.

key_1 → value_1

key_2 → value_2

• •

NXT WAVE™ Powered by IB HUBS

← Question 1 of 2

What happens if you try to access a key that doesn't exist in a dictionary?

It returns null .

It raises a **KeyError**.

Complete the MCQs to proceed Continue

NH163 / Assemb...
Closed road

Search

12:00 14-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fc58bfa6-14ab-4a80-aed6-d23dc8286f

NXT WAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

7. How to create a Dictionary?

A dictionary is created by enclosing items within **{curly}** brackets.

Each item in the dictionary has a Key-Value pair separated by a **comma**.

Code

```
1 - dict_a = {}
2 -     "name": "Teja",
3 -     "age": 15
4 - }
```

PYTHON

In the above dictionary, the

- keys are name and age
- values are Teja and 15

8. How to get the keys of a dictionary?

The `keys()` method allows us to get the keys of dictionary. It returns a view object of the type `dict_keys` that holds a list of all keys.

Code

```
dict_a.keys()
```

PYTHON Continue

UV 1 28°C High UV

Search

12:01 14-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fc58bfa6-14ab-4a80-aed6-d23dc8286f

NXT WAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

8. How to get the keys of a dictionary?

The `keys()` method allows us to get the keys of dictionary. It returns a view object of the type `dict_keys` that holds a list of all keys.

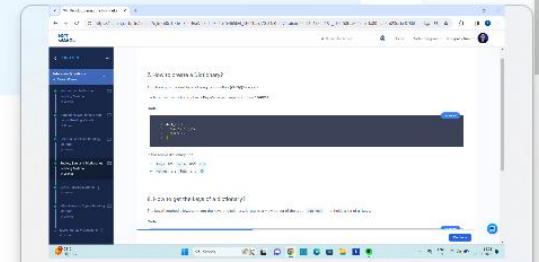
Code

```
1+ dict_a = {  
2+     "name": "Teja",  
3+     "age": 15  
4+ }  
5+ print(dict_a.keys())
```

PYTHON

Output

```
dict_keys(['name', 'age'])
```

 Snipping Tool

Screenshot copied to clipboard and saved
Select here to mark up and share the image

28°C High UV

Search

12:01 14-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccgp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fc58bfa6-14ab-4a80-aed6-d23dc8286f

NXT WAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. Define a dictionary in Python and add an item to it.

In Python, a dictionary is a collection of key-value pairs. We can define a dictionary using curly braces {} and separate each key-value pair with a colon : . Keys in a Python dictionary are unique and immutable.

Example:

Create a empty dictionary my_dict and add a value Ford with key brand to it.

PYTHON

```
1 # Defining a dictionary
2 my_dict = {}
3
4 # Adding items to the dictionary
5 my_dict['brand'] = 'Ford'
```

← Question 1 of 3

Which of the following methods removes a key-value pair from a dictionary?

- my_dict.remove('key')
- my_dict.pop('key')
- my_dict.delete('key')
- my_dict.discard('key')

Next

Complete the MCQs to proceed Continue

1 28°C High UV

Search

12:01 14-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fc58bfa6-14ab-4a80-aed6-d23dc8286f

NXT WAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. What is Key Value pair in python?

In Python, a key-value pair is a fundamental component of certain data structures, most notably dictionaries. A key-value pair consists of two elements: the key, which is a unique identifier, and the value, which is the data associated with the key.

Keys are unique within a dictionary and must be immutable. They can be of various types such as strings, numbers, or tuples. Values can be of any type, including numbers, strings, lists, or even other dictionaries, and they can be modified or updated.

More Information

Syntax:

```
1 dict[key] = value
```

Example:

```
1 student = {
2     'name': 'John',
3     'age': 20,
4     'grade': 'A',
5     'courses': ['Math', 'English', 'Science']
6 }
```

Note: If you try to add a key-value pair to a dictionary using an existing key then the existing value will be replaced with the new one.

Continue

1 28°C High UV

Search

12:02 14-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fc58bfa6-14ab-4a80-aed6-d23dc8286f

NXT WAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

9. Why List can't be used as a key in the dictionary/ Are the dictionary keys mutable?

A Python dictionary can have only keys of immutable types. Lists cannot be used as keys in a dictionary because they are mutable.

Q. How to use List and Dictionary in Python?

Lists:

Lists are ordered, mutable collections. They are created with square brackets [].

- Example:**

```
my_list = [1, 2, "hello"] # Creation
print(my_list[1]) # Accessing
my_list[2] = "world" # Modifying
my_list.append(6) # Adding
```

Dictionary:

Dictionaries are unordered collections of key-value pairs, created with curly braces {} .

- Example:**

```
my_dict = {"name": "John", "age": 25} # Creation
print(my_dict["name"]) # Accessing
my_dict["age"] = 26 # Modifying
my_dict["city"] = "New York" # Adding
```

Continue

1 28°C High UV

Search

12:02 14-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fc58bfa6-14ab-4a80-aed6-d23dc8286f

NXT
WAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

10. How to create an empty dictionary?

Using `dict()` built-in function

code

```
1 dict_a = dict()
2 print(type(dict_a))
3 print(dict_a)
```

PYTHON

Output

```
<class 'dict'>
{}
```

Using `{}` symbol

code

```
1 dict_a = {}
2 print(type(dict_a))
3 print(dict_a)
```

PYTHON

Output

```
<class 'dict'>
{}
```

Continue

1 28°C Haze

Search

12:03 14-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fc58bfa6-14ab-4a80-aed6-d23dcbc8286f

NXT
WAVES™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPS Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

11. How to combine two dictionaries?

We can combine two dictionaries using `update()` method:

Syntax

```
dictionary.update(iterable)
```

code

```
1 dict_1 = {'a': 1, 'b': 2}
2 dict_2 = {'c': 3, 'd': 4}
3
4 dict_1.update(dict_2)
5 print(dict_1)
```

PYTHON

Output

```
{'a': 1, 'b': 2, 'c': 3, 'd': 4}
```

Continue

1 28°C Haze

Search

12:03 14-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fc58bfa6-14ab-4a80-aed6-d23dc8286f

12. What are the differences between Lists, Tuples, Sets and Dictionaries?

Lists	Tuples	Sets	Dictionaries
A list is an ordered collection of data	A tuple is an ordered collection of data	A set is an unordered collection	A dictionary is an unordered collection of data that stores data in Key-Value pairs
Lists are mutable	Tuples are immutable	Sets are mutable	Dictionaries are mutable
Lists are enclosed within square braces []	Tuples are enclosed within parenthesis ()	Sets are enclosed within curly brackets {}	Dictionaries are enclosed within curly brackets {} in the form of Key-Value pairs
Example: [1, 2, 3, 4, 5]	Example: (1, 2, 3, 4, 5)	Example: {1, 2, 3, 4, 5}	Example: { a:1, b:2, c:3, d:4, e:5 }
List allows duplicate elements	Tuple allows duplicate elements	Set will not allow duplicate elements	Dictionary doesn't allow duplicate keys
List can be created using list() function	Tuple can be created using tuple() function	Set can be created using set() function	Dictionary can be created using dict() function

28°C Haze

Search

12:03 14-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fc58bfa6-14ab-4a80-aed6-d23dcbc8286f

Set	List
Sets are unordered collections, meaning the elements in a set have no specific order.	Lists are ordered collections, meaning the elements in a list maintain a specific order.
We cannot access elements by index.	We can access elements in a list by their index.
Sets only contain unique elements.	Lists can contain duplicate elements.
Sets are created using curly braces {}.	Lists are created using square brackets [].
Example: my_set = {1, 2, 3, 4}	Example: my_list = [1, 2, 3, 4]

Question 1 of 4

Which data structure is suitable for maintaining an ordered collection of elements in Python?

- Dictionary
- List
- Set
- None of the above

Next

1 28°C Haze

Search

Search

Next

12:04 14-02-2024

Q. What is the difference between set and dictionary?

Parameters	Set	Dictionary
Definition	A set is an unordered collection of unique elements.	A dictionary is an unordered collection of key-value pairs.
Syntax	Syntax includes curly brackets {}.	Syntax includes curly brackets { , } with ',' separated key-value data.
Duplicate Data	All elements are unique in a set.	Keys are unique; values can be duplicated.
Indexing	Does not have an index based mechanism.	Has key-based indexing.
Sorting	Unordered, so sorting is not applicable.	Dictionaries can be sorted by keys.
Search	Sets do not support direct searching.	The get() method retrieves values by keys.

28°C
Haze

Search

ENG
IN12:04
14-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fc58bfa6-14ab-4a80-aed6-d23dc8286f

NXT
WAVES™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

13. How to swap two numbers without using a third variable?

Companies Asked: Translytics Business Services

There are multiple ways to swap two numbers without using a third variable:

1. Using Arithmetic Operations

```
1 a = 5
2 b = 3
3
4 a = a + b
5 b = a - b
6 a = a - b
```

PYTHON

After executing the above code, the value of `a` will be `3` and the value of `b` will be `5`.

2. Using Python's tuple assignment

```
1 a = 5
2 b = 3
3
4 a, b = b, a
```

PYTHON

After executing the code, the value of `a` will be `3` and the value of `b` will be `5`.

Continue

1 28°C Haze

Search

12:05 14-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fc58bfa6-14ab-4a80-aed6-d23dc8286f

NXT
wAVe™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions < 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Interview Questions > Tuples, Sets and Dictionaries Reading Material

Given data:

```
1 - data = []
2   'period' :['31-03-2023', '30-04-2023', '31-05-2023', '30-06-2023', '31-07-2023', '30-08-2023']
3   'method':['pr1','pr1','pr2','pr1','pr2','pr2'],
4   'value':[2125,150,6400,1000,4000,2749]
5 }
```

PYTHON

Companies Asked: Translytics Business Services

To group the data by "method" and calculate the sum of the values, using dictionary data structures, we can loop over the values and use a dictionary to store the summed values grouped by the method. Here is the code to achieve this:

```
1 - data = {
2   'period': ['31-03-2023', '30-04-2023', '31-05-2023', '30-06-2023', '31-07-2023', '30-08-2023'],
3   'method': ['pr1', 'pr1', 'pr2', 'pr1', 'pr2', 'pr2'],
4   'value': [2125, 150, 6400, 1000, 4000, 2749]
5 }
6
7 result = {}
8
9 for i, method in enumerate(data['method']):
10    if method not in result:
```

PYTHON

In this code:

- We initialized an empty dictionary `result` to store the sums for each method.
- We looped over each method in the 'method' list using a for loop where `i` is the index and `method` is the value.
- Inside the loop, we checked if the method is not already a key in the `result` dictionary, if not, we add it with a value of 0.
- Then, we added the corresponding value from the 'value' list to the method's sum in the `result` dictionary using the index `i`.
- Finally, we printed the `result` dictionary to see the output.

This will give you the summed values for each method.

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fc58bfa6-14ab-4a80-aed6-d23dc8286f

NXT
WAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Interview Questions > Tuples, Sets and Dictionaries Reading Material

Write a code to group the data by method and calculate the sum of the values using the dictionary.

Given data:

```
1- data = []
2-   'period': ['31-03-2023', '30-04-2023', '31-05-2023', '30-06-2023', '31-07-2023', '30-08-2023'],
3-   'method': ['pr1', 'pr1', 'pr2', 'pr1', 'pr2', 'pr2'],
4-   'value': [2125, 150, 6400, 1000, 4000, 2749]
5- }
```

Companies Asked: Translytics Business Services

To group the data by "method" and calculate the sum of the values, using dictionary data structures, we can loop over the values and use a dictionary to store the summed values grouped by the method. Here is the code to achieve this:

```
1- data = [
2-   'period': ['31-03-2023', '30-04-2023', '31-05-2023', '30-06-2023', '31-07-2023', '30-08-2023'],
3-   'method': ['pr1', 'pr1', 'pr2', 'pr1', 'pr2', 'pr2'],
4-   'value': [2125, 150, 6400, 1000, 4000, 2749]
5- ]
6-
7- result = {}
8-
9- for i, method in enumerate(data['method']):
10-   if method not in result:
```

In this code:

- We initialized an empty dictionary `result` to store the sums for each method.
- We looped over each method in the 'method' list using a for loop where `i` is the index and `method` is the value.
- Inside the loop, we checked if the method is not already a key in the `result` dictionary, if not, we add it with a value of 0.
- Then, we added the corresponding value from the 'value' list to the method's sum in the `result` dictionary using the index `i`.
- Finally, we printed the `result` dictionary to see the output.

This will give you the summed values for each method.

28°C Haze

Search

12:06 14-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fc58bfa6-14ab-4a80-aed6-d23dc8286f

NXT
WAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions

Introduction to Python Reading Material (2 hours, 10 mins)

Conditional Statements and Loops Reading Material (25 mins)

List and Functions Reading Material (25 mins)

Tuples, Sets and Dictionaries Reading Material (25 mins)

OOPs Reading Material (20 mins)

Miscellaneous Topics Reading Material (20 mins)

More Interview Questions (20 mins)

Given data:

```
1 - data = []
2 -     'period' :['31-03-2023','30-04-2023','31-05-2023','30-06-2023','31-07-2023','30-08-2023']
3 -     'method':['pr1','pr1','pr2','pr1','pr2','pr2'],
4 -     'value':[2125,150,6400,1000,4000,2749]
5 - }
```

Companies Asked: Translytics Business Services

To group the data by "method" and calculate the sum of the values, using dictionary data structures, we can loop over the values and use a dictionary to store the summed values grouped by the method. Here is the code to achieve this:

```
5 }
6
7 result = {}
8
9 for i, method in enumerate(data['method']):
10    if method not in result:
11        result[method] = 0
12        result[method] += data['value'][i]
13
14 print(result)
```

In this code:

- We initialized an empty dictionary `result` to store the sums for each method.
- We looped over each method in the 'method' list using a for loop where `i` is the index and `method` is the value.
- Inside the loop, we checked if the method is not already a key in the `result` dictionary, if not, we add it with a value of 0.
- Then, we added the corresponding value from the 'value' list to the method's sum in the `result` dictionary using the index `i`.
- Finally, we printed the `result` dictionary to see the output.

This will give you the summed values for each method.

28°C Haze

Search

12:06 14-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fc58bfa6-14ab-4a80-aed6-d23dc8286f

NXT
wAVG™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions (2 hours, 10 mins)

- Introduction to Python (25 mins)
- Conditional Statements and Loops (15 mins)
- List and Functions (25 mins)
- Tuples, Sets and Dictionaries (25 mins)
- OOPs (20 mins)
- Miscellaneous Topics (20 mins)
- More Interview Questions (20 mins)

Q. What is the difference between a tuple and a dictionary?

Companies Asked: GEPL Capital

A **tuple** is an ordered, immutable collection of elements. Once created, elements cannot be changed or removed. Elements are accessed via index, starting from zero.

A **dictionary** is an unordered collection of key-value pairs. It is mutable, meaning you can add, remove, or change items. Items are accessed using keys rather than index.

Question 1 of 3

Which of the following is true about the mutability of tuples and dictionaries in Python?

- Tuples are mutable, while dictionaries are immutable.
- Tuples are immutable, while dictionaries are mutable.**
- Both tuples and dictionaries are mutable.
- Both tuples and dictionaries are immutable.

Next

Complete the MCQs to proceed Continue

1 28°C Haze

Search

12:07 14-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fc58bfa6-14ab-4a80-aed6-d23dc8286f

NXT
wAVG™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions (2 hours, 10 mins)

- Introduction to Python (Reading Material, 25 mins)
- Conditional Statements and Loops (Reading Material, 15 mins)
- List and Functions (Reading Material, 25 mins)
- Tuples, Sets and Dictionaries (Reading Material, 25 mins)
- OOPs (Reading Material, 20 mins)
- Miscellaneous Topics (Reading Material, 20 mins)
- More Interview Questions (20 mins)

Q. What is the difference between tuple and list?

Lists are mutable, which means you can change their elements (add, remove, or modify) after creation. Lists are defined using square brackets [].

```
1 my_list = [1, 2, 3]
```

Tuples are immutable, meaning once you create a tuple, you cannot change its elements. If you need to make any changes, you have to create a new tuple. Tuples are defined using parentheses ().

```
1 my_tuple = (1, 2, 3)
```

Question 1 of 3

Which data structure is defined using square brackets in Python?

- Tuples
- Lists
- Sets
- Dictionaries

Next

Complete the MCQs to proceed Continue

1 28°C Haze

Search

12:07 14-02-2024 ENG IN

Q. Why do we use tuples, and what is their purpose?

Companies Asked: GEPL Capital

Tuples are used to store ordered collections of items, and their primary purpose is to group related data together while ensuring it remains unchangeable (immutable).

This immutability is valuable in situations where data should not be changed, such as when using them as dictionary keys.

◀ Question 1 of 3

What is the primary purpose of using tuples in Python?

- To store unordered collections of items.
- To group related data together while ensuring immutability. 
- To perform mathematical operations.
- To create dynamic arrays.

Next



28°C
Haze



Search



ENG
IN



12:08
14-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fc58bfa6-14ab-4a80-aed6-d23dc8286f

NXT
wAVG™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPS Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. Have you learnt tuples and arrays?

Yes, I'm familiar with both tuples and arrays. These are common data structures used in Python languages.

1. Tuples:
Tuples are immutable, which follows the defined order. In tuples we can store different data types.
To create tuple we use parentheses `()`

Example:

```
1 my_tuple = (1, "apple", 3.14)
```

2. Arrays:
Arrays are mutable, which follows the defined order. In array we can store items of the same data type.
To create Arrays we use square brackets `[]`.

Example:

```
1 my_array = [1, 2, 3, 4, 5]
```

Question 1 of 5

What will be the result if you attempt to modify the first element of a tuple?

The tuple will allow modification.
 Python will raise a `TypeError`.

Complete the MCQs to proceed Continue

28°C Haze

Search

12:09 14-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fc58bfa6-14ab-4a80-aed6-d23dc8286f

NXT
WAVES™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. You have nested dictionaries, with names as keys in the outer dictionary and another dictionary for each name key, with subjects as keys and marks as values. Get the names of the people who have more than 50 marks in English.

Solution

PYTHON

```
1: dictionary = {
2:     "Ram": {"English": 75, "Maths": 90, "Physics": 80},
3:     "Rahul": {"English": 60, "Maths": 70, "Physics": 85},
4:     "Ramesh": {"English": 40, "Maths": 70, "Physics": 85},
5: }
6:
7: result = []
8:
9: for name, score in dictionary.items():
10:     if 'English' in score and score['English'] > 50:
```

Output:

```
[('Ram', 'Ramesh')]
```

For each student name and score in the dictionary, Check if the student's score in English is greater than 50, if condition satisfied, add the student's name to the result

▼ Another approach using get method

- get() method is used to returns the value of the item with the specified key.

PYTHON

```
1: dictionary = {
2:     "Ram": {"English": 75, "Maths": 90, "Physics": 80},
3:     "Rahul": {"English": 60, "Maths": 70, "Physics": 85},
4:     "Ramesh": {"English": 40, "Maths": 70, "Physics": 85},
5: }
6:
7: result = []
8:
9:
10: for name, score in dictionary.items():
```

Continue

1 28°C Haze

Search

12:09 14-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fc58bfa6-14ab-4a80-aed6-d23dc8286f

NXT
WAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions (2 hours, 10 mins)

- Introduction to Python Reading Material (25 mins)
- Conditional Statements and Loops Reading Material (15 mins)
- List and Functions Reading Material (25 mins)
- Tuples, Sets and Dictionaries Reading Material (25 mins)
- OOPs Reading Material (20 mins)
- Miscellaneous Topics Reading Material (20 mins)
- More Interview Questions (20 mins)

INTERVIEW QUESTIONS > Tuples, Sets and Dictionaries Reading Material

Output:

```
[['Ram', 'Ramesh']]
```

For each student name and score in the dictionary, Check if the student's score in English is greater than 50, if condition satisfied, add the student's name to the result

▼ Another approach using get method

- get() method is used to returns the value of the item with the specified key.

PYTHON

```
5 }  
6  
7 result = []  
8  
9  
10 for name, score in dictionary.items():  
11     if score.get('English', 0) > 50:  
12         result.append(name)  
13  
14 print(result)
```

Output:

```
[['Ram', 'Ramesh']]
```

For each student name and score in the dictionary, Check if the student's score in English is greater than 50 by using the get() method, if condition satisfied, add the student's name to the result list

28°C Haze

Search

Continuous

12:10 14-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fc58bfa6-14ab-4a80-aed6-d23dc8286f

NXT
WAVES™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPS Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. Imagine that there are 100 students attending English exams. Tell me a code for those students who got 50 marks in English.

Solution

- get() method is used to returns the value of the item with the specified key.

```
1 dictionary = {  
2     "Ram": {"English": 75, "Maths": 90, "Physics": 80},  
3     "Rahul": {"English": 60, "Maths": 70, "Physics": 85},  
4     "Ramesh": {"English": 40, "Maths": 70, "Physics": 85},  
5 }  
6  
7 result = []  
8  
9 for name, score in dictionary.items():  
10    if score.get('English', 0) == 50:  
11        result.append(name)
```

PYTHON

For each student name and score in the dictionary, Check if the student's score in English is equal to 50 by using the get() method, if condition satisfied, add the student's name to the result list

▼ Another approach

```
1 dictionary = {  
2     "Ram": {"English": 75, "Maths": 90, "Physics": 80},  
3     "Rahul": {"English": 60, "Maths": 70, "Physics": 85},  
4     "Ramesh": {"English": 40, "Maths": 70, "Physics": 85},  
5 }  
6  
7 result = []  
8  
9 for name, score in dictionary.items():  
10    if 'English' in score and score['English'] == 50:  
11        result.append(name)
```

PYTHON

For each student name and score in the dictionary, Check if the student's score in English is greater than 50, if condition satisfied, add the student's name to the result.

Continue

Ask doubt

Notes Discussions

28°C Haze

Search

12:11 14-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fc58bfa6-14ab-4a80-aed6-d23dc8286f

NXT
WAVES™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPS Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

get() method is used to returns the value of the item with the specified key.

```
4     "Ramesh": {"English": 40, "Maths": 70, "Physics": 85},  
5 }  
6  
7 result = []  
8  
9 for name, score in dictionary.items():  
10 if score.get('English', 0) == 50:  
11     result.append(name)  
12  
13 print(result)
```

PYTHON

For each student name and score in the dictionary, Check if the student's score in English is equal to 50 by using the get() method, if condition satisfied, add the student's name to the result list

▼ Another approach

```
4     "Ramesh": {"English": 40, "Maths": 70, "Physics": 85},  
5 }  
6  
7 result = []  
8  
9 for name, score in dictionary.items():  
10 if 'English' in score and score['English'] == 50:  
11     result.append(name)  
12  
13 print(result)
```

PYTHON

For each student name and score in the dictionary, Check if the student's score in English is greater than 50, if condition satisfied, add the student's name to the result.

Continue

Ask doubt

Notes Discussions

28°C Haze

Search

12:11 14-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fc58bfa6-14ab-4a80-aed6-d23dc8286f

NXT
wAVG™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. A dictionary containing the values of students and subjects marks, return the student's name who has scored more than 30 marks in maths.

PYTHON

```
1- dictionary = {  
2-     "Sai": {"English": 75, "Maths": 90, "Physics": 80},  
3-     "Praveen": {"English": 60, "Maths": 28, "Physics": 85},  
4-     "Rahul": {"English": 40, "Maths": 70, "Physics": 85},  
5- }  
6-  
7- students_above_30_in_maths = []  
8-  
9- for name, subject_score in dictionary.items():  
10-     if subject_score.get("Maths", 0) > 30:
```

Explanation:

For each student `name` and `subject_score` in the dictionary, check if the student's `subject_score` in `Maths` is greater than 30 by using the `get()` method, we are providing two arguments in the `get` method first one is used to check the `key` and second one sets a default value `0` if the key is not present. If condition is satisfied, we would add the student's name to the `students_above_30_in_maths` list.

Note

`get()` method is used to returns the value of the item with the specified key.

Question 1 of 3

What does the `dictionary.items()` method do in Python?

- It returns a list of all the keys in the dictionary
- It returns a list of all the values in the dictionary
- It returns a list of tuples where each tuple contains a key-value pair from the dictionary
- It returns a dictionary containing the items of the original dictionary

Complete the MCQs to proceed Continue

28°C Haze

Search

12:12 14-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fc58bfa6-14ab-4a80-aed6-d23dc8286f

NXT WAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. A dictionary containing the values of students and subjects marks, return the student's name who has scored more than 30 marks in maths.

PYTHON

```
4     "Rahul": {"English": 40, "Maths": 70, "Physics": 85},  
5 }  
6  
7 students_above_30_in_maths = []  
8  
9 for name, subject_score in dictionary.items():  
10 if subject_score.get("Maths", 0) > 30:  
11     students_above_30_in_maths.append(name)  
12  
13 print(students_above_30_in_maths)
```

Explanation:

For each student `name` and `subject_score` in the dictionary, check if the student's `subject_score` in `Maths` is greater than 30 by using the `get()` method, we are providing two arguments in the `get` method first one is used to check the `key` and second one sets a default value `0` if the key is not present. If condition is satisfied, we would add the student's name to the `students_above_30_in_maths` list.

Note

`get()` method is used to returns the value of the item with the specified key.

Question 1 of 3

What does the `dictionary.items()` method do in Python?

- It returns a list of all the keys in the dictionary
- It returns a list of all the values in the dictionary
- It returns a list of tuples where each tuple contains a key-value pair from the dictionary
- It returns a dictionary containing the items of the original dictionary

Snipping Tool

Screenshot copied to clipboard and saved
Select here to mark up and share the image

28°C Haze

Search

12:12 14-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fc58bfa6-14ab-4a80-aed6-d23dc8286f

NXT
wAVG™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPS Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. There is a set of students named A, B, C, and D, and their marks will be stored in a dictionary. Within A, there is another dictionary that stores their marks for Python and JavaScript. The same applies to B, C, and D. We want to print the Python marks to the console.

Solution

PYTHON

```
1 - students = {  
2 -     'A': {'python': 20, 'js': 30},  
3 -     'B': {'python': 25, 'js': 35},  
4 -     'C': {'python': 22, 'js': 28},  
5 -     'D': {'python': 18, 'js': 32}  
6 - }  
7 -  
8 - for student, marks in students.items():  
9 -     print(student,marks['python'])
```

Print the Python mark from `python` key of each student in the `students` dictionary.

▼ Another approach

- `get()` method is used to extract the key-value from the dictionary.

PYTHON

```
1 - students = {  
2 -     'A': {'python': 20, 'js': 30},  
3 -     'B': {'python': 25, 'js': 35},  
4 -     'C': {'python': 22, 'js': 28},  
5 -     'D': {'python': 18, 'js': 32}  
6 - }  
7 -  
8 - for student, subject_marks in students.items():  
9 -     python_marks = subject_marks.get('python', 0) # Get Python marks, default to 0 if not found  
10 -    print(student,python_marks)
```

use `get()` method to extract the `python` key and print Python mark from `students` dictionary.

Continue

1 29°C Partly sunny

Search

Clouds, Heart, Photo, Video, Google Chrome, Microsoft Edge, File, L, Spotify

ENG IN 12:13 14-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fc58bfa6-14ab-4a80-aed6-d23dc8286f

NXT
WAVES™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions (2 hours, 10 mins)

- Introduction to Python Reading Material (25 mins)
- Conditional Statements and Loops Reading Material (15 mins)
- List and Functions Reading Material (25 mins)
- Tuples, Sets and Dictionaries Reading Material (25 mins)
- OOPs Reading Material (20 mins)
- Miscellaneous Topics Reading Material (20 mins)
- More Interview Questions (20 mins)

Q. What is the output of the given code? Explain me the code.

PYTHON

```
1 d={"PA":"Pittsburgh", "NY": "New York"}  
2 d["WA"] = "Seattle"  
3 d["NY"] = "Buffalo"  
4  
5 if "Pittsburgh" in d:  
6     d.pop("Pittsburgh")  
7 print(d)
```

Explanation:

The code creates a dictionary `d` and adds entries to it. It then checks if the value `Pittsburgh` is present in the dictionary using the `in` operator. However, since `Pittsburgh` is a value, not a key, the condition will always be false and the subsequent block of code will not be executed.

Output:

PYTHON

```
1 { 'PA': 'Pittsburgh', 'NY': 'Buffalo', 'WA': 'Seattle'}
```

Question 1 of 3

What does the `pop` method do in Python when applied to a dictionary?

- Adds a new key-value pair to the dictionary
- Removes duplicate elements from the dictionary
- Removes the key-value pair with the specified key from the dictionary

Complete the MCQs to proceed

Continue

DOW -1.35%

Search

12:13 14-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fc58bfa6-14ab-4a80-aed6-d23dc8286f

NXT
wAVG™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. In the given dictionary, if I want to remove Pittsburgh from the dictionary, what do I need to do? Make the change in the code to remove Pittsburgh.

PYTHON

```
1 d={"PA":"Pittsburgh", "NY": "New York"}  
2 d["WA"] = "Seattle"  
3 d["NY"] = "Buffalo"  
4  
5 if "Pittsburgh" in d:  
6     d.pop("Pittsburgh")  
7 print(d)
```

Using pop method:

PYTHON

```
1 d = {"PA": "Pittsburgh", "NY": "New York"}  
2 d["WA"] = "Seattle"  
3 d["NY"] = "Buffalo"  
4  
5 if "PA" in d:  
6     d.pop("PA")  
7 print(d)
```

Explanation:

We are using the `value` instead of the `key` to check if the city is in the dictionary. So, `if` condition will always be `false` since there is no key named `Pittsburgh` in the dictionary. Hence, it won't remove anything from the dictionary. We should replace `Pittsburgh` in the if condition block with its corresponding key, `PA`, to successfully remove the entry from the dictionary.

► More Information

Continue

DOW -1.35%

Search

Clouds ENG IN 12:14 14-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fc58bfa6-14ab-4a80-aed6-d23dc8286f

NXT
WAVES™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPS Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

We are using the `value` instead of the `key` to check if the city is in the dictionary. So, `if` condition will always be `false` since there is no key named `Pittsburgh` in the dictionary. Hence, it won't remove anything from the dictionary. We should replace `Pittsburgh` in the `if` condition block with its corresponding key, `PA`, to successfully remove the entry from the dictionary.

▼ More Information

Using `del` keyword:

```
1 d = {"PA": "Pittsburgh", "NY": "New York"}  
2 d["WA"] = "Seattle"  
3 d["NY"] = "Buffalo"  
4  
5 - if "Pittsburgh" in d.values():  
6     del d["PA"]  
7  
8 print(d)
```

PYTHON

Explanation:

The `if` statement checks whether the string `Pittsburgh` is present as a value in the dictionary `d`. The `del` keyword deletes the key-value pair where the key is `PA`.

Question 1 of 3

What is the source of error in the code given below?

```
1 d = {"PA": "Pittsburgh", "NY": "New  
2 d["WA"] = "Seattle"  
3 d["NY"] = "Buffalo"  
4  
5 - if "PA" in d:  
6     d.remove("PA")
```

PYTHON

The `remove` method is not applicable to dictionaries.
 The `if` condition is checking for the wrong key.

Complete the MCQs to proceed Continue

SENSEX -0.64%

Search

12:14 14-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fc58bfa6-14ab-4a80-aed6-d23dc8286f

NXT
wAVG™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPS Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. Write code for the occurrence of each letter in Python?

PYTHON

```
1 def count_letters(text):
2     letter_count = {}
3     for letter in text:
4         # Check if the character is a letter
5         if letter.isalpha():
6             letter = letter.lower() # Convert to lowercase to count case-insensitively
7             if letter in letter_count:
8                 letter_count[letter] += 1
9             else:
10                 letter_count[letter] = 1
```

Iterates over input string, check if the current character is a letter, converting each letter to lowercase and adding it to a dictionary with a count of 1, or incrementing the count if the letter already exists.

Question 1 of 2

what will be the output after executing the following code?

PYTHON

```
1 lst = ["a", "a", "b", "c", "c"]
2 count = {}
3 for i in lst:
4     if i.isalpha():
5         if i in count:
6             count[i] += 1
7         else:
8             count[i] = 1
9 print(count)
```

(a) {'a': 1, 'b': 1, 'c': 1}

(b) {'a': 2, 'b': 1, 'c': 2}

(c) Type error

Complete the MCQs to proceed Continue

1 29°C Partly sunny

Search

12:26 14-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fc58bfa6-14ab-4a80-aed6-d23dc8286f

NXT
wAVe™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPS Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. Write code for the occurrence of each letter in Python?

PYTHON

```
7-         if letter in letter_count:  
8-             letter_count[letter] += 1  
9-         else:  
10-             letter_count[letter] = 1  
11-     return letter_count  
12-  
13- # Example usage  
14- text = "Hello, World!"  
15- result = count_letters(text)  
16- print(result)
```

Iterates over input string, check if the current character is a letter, converting each letter to lowercase and adding it to a dictionary with a count of 1, or incrementing the count if the letter already exists.

Question 1 of 2

what will be the output after executing the following code?

PYTHON

```
1 lst = ["a", "a", "b", "c", "c"]  
2 count = {}  
3 for i in lst:  
4     if i.isalpha():  
5         if i in count:  
6             count[i] += 1  
7         else:  
8             count[i] = 1  
9 print(count)
```

{'a': 1, 'b': 1, 'c': 1}
 {'a': 2, 'b': 1, 'c': 2} Type error

Complete the MCQs to proceed Continue

1 29°C Partly sunny

Search

12:26 14-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fc58bfa6-14ab-4a80-aed6-d23dc8286f

NXT
wAVG™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

It checks if the character is a special character (e.g., !, @, #, etc.).

Q. You have to write a program to get the frequency of every letter in Python. In addition, print the letters that have the top 5 frequencies.

PYTHON

```
1- def sort_by_frequency(item):
2-     return item[1]
3-
4- def letter_frequency(paragraph):
5-     all_freq = {}
6-
7-     for char in paragraph:
8-         if char.isalpha():
9-             char_lower = char.lower()
10-            if char_lower in all_freq:
```

Explanation:

The code counts the frequency of each letter in a paragraph and prints the top 5 most frequent letters. It first converts the paragraph to lowercase to ensure case-insensitivity.

Question 1 of 3

What does the `lower()` method in Python do?

- Converts all characters in a string to uppercase.
- Converts all characters in a string to lowercase.
- Removes all leading whitespaces from a string.
- Checks if a string is entirely made up of alphabetic characters.

Complete the MCQs to proceed Continue

1 29°C Partly sunny

Search

12:27 14-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fc58bfa6-14ab-4a80-aed6-d23dc8286f

NXT
wAVG™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions (2 hours, 10 mins)

- Introduction to Python Reading Material (25 mins)
- Conditional Statements and Loops Reading Material (15 mins)
- List and Functions Reading Material (25 mins)
- Tuples, Sets and Dictionaries Reading Material (25 mins)
- OOPs Reading Material (20 mins)
- Miscellaneous Topics Reading Material (20 mins)
- More Interview Questions (20 mins)

It checks if the character is a special character (e.g., !, @, #, etc.).

Q. You have to write a program to get the frequency of every letter in Python. In addition, print the letters that have the top 5 frequencies.

PYTHON

```
11         all_freq[char_lower] += 1
12     else:
13         all_freq[char_lower] = 1
14
15     sorted_freq = sorted(all_freq.items(), key=sort_by_frequency, reverse=True)
16    for i in range(min(5, len(sorted_freq))):
17        print(sorted_freq[i][0], ":", sorted_freq[i][1])
18
19 paragraph = input()
20 letter_frequency(paragraph)
```

Explanation:

The code counts the frequency of each letter in a paragraph and prints the top 5 most frequent letters. It first converts the paragraph to lowercase to ensure case-insensitivity.

Question 1 of 3

What does the `lower()` method in Python do?

- Converts all characters in a string to uppercase.
- Converts all characters in a string to lowercase.
- Removes all leading whitespaces from a string.
- Checks if a string is entirely made up of alphabetic characters.

Complete the MCQs to proceed Continue

1 29°C Partly sunny

Search

12:27 14-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fc58bfa6-14ab-4a80-aed6-d23dc8286f

NXT
wAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions (2 hours, 10 mins)

- Introduction to Python Reading Material (25 mins)
- Conditional Statements and Loops Reading Material (15 mins)
- List and Functions Reading Material (25 mins)
- Tuples, Sets and Dictionaries Reading Material (25 mins)
- OOPS Reading Material (20 mins)
- Miscellaneous Topics Reading Material (20 mins)
- More Interview Questions (20 mins)

Q. Write a function that takes a string as input and returns a dictionary. The keys will be the different alphabets in the string, and the values will represent the frequency of each alphabet. From this dictionary, give me the top 5 most frequent letters.

```
input: s="asish"
output: {a:1,s:2,i:1,h:1}
```

Intent of the interviewer: The interviewer expects you to remove all occurrences of comma (,), space, and special characters and write a code which is case sensitive (uppercase and lowercase characters are considered to be different).

PYTHON

```
1- def letter_frequency(paragraph):
2-     all_freq = {}
3-
4-     for char in paragraph:
5-         if char.isalpha():
6-             if char in all_freq:
7-                 all_freq[char] += 1
8-             else:
9-                 all_freq[char] = 1
10-
```

Explanation:

The function iterates through each character in the paragraph. If the character is already present, its frequency is incremented, otherwise, a new entry is created with a frequency of 1. After counting the frequencies, the code sorts the `all_freq` dictionary based on the frequencies in descending order. At last top 5 frequencies will get printed.

Continue

1 29°C Partly sunny

Search

12:28 14-02-2024 ENG IN



Welcome to use Lenovo Smart Key !

Try Smart key to get quick access to Lenovo applications and service.

Next

Revolutionizing the Job Market - X

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fc58bfa6-14ab-4a80-aed6-d23dc8286f Download Search Star Code Playground Jangipally Ranjith

**NXT
wAVES™**

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python** Reading Material 25 mins
- Conditional Statements and Loops** Reading Material 15 mins
- List and Functions** Reading Material 25 mins
- Tuples, Sets and Dictionaries** Reading Material 25 mins
- OOPs** Reading Material 20 mins
- Miscellaneous Topics** Reading Material 20 mins
- More Interview Questions** 20 mins

Q. Write a function that takes a string as input and returns a dictionary. The keys will be the different alphabets in the string, and the values will represent the frequency of each alphabet. From this dictionary, give me the top 5 most frequent letters.

```
input: s="asish"
output: {a:1,s:2,i:1,h:1}
```

Intent of the interviewer: The interviewer expects you to remove all occurrences of comma (,), space, and special characters and write a code which is case sensitive (uppercase and lowercase characters are considered to be different).

PYTHON

```
10
11     sorted_freq = dict(sorted(all_freq.items(), key=lambda x: x[1], reverse=True))
12     top_5freq = dict(list(sorted_freq.items())[:5])
13
14     return top_5freq
15
16 paragraph = "Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation"
17 result = letter_frequency(paragraph)
18
19 print(result)
```

Explanation:

The function iterates through each character in the paragraph. If the character is already present, its frequency is incremented, otherwise, a new entry is created with a frequency of 1. After counting the frequencies, the code sorts the `all_freq` dictionary based on the frequencies in descending order. At last top 5 frequencies will get printed.

Continue

Revolutionizing the Job Market - X

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fc58bfa6-14ab-4a80-aed6-d23dc8286f Download Search Star Code Playground Jangipally Ranjith

NXT wAVE™

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

The function will not return any output.

Q. Create a variable inventory of data type as a list. Create 2 to 3 product dictionaries and add these products to the list.

Intent of the interviewer: The interviewer is expecting to create 2 to 3 dictionary and then update the value of one key from the dictionary.

PYTHON

```
1 product1 = {  
2     "productID": 1,  
3     "prodName": "Laptop",  
4     "quantity": 1,  
5     "prodPrice": 1200.00,  
6     "category": "Electronics"  
7 }  
8  
9 product2 = {  
10    "productID": 2,
```

Three product dictionaries are created: `product1`, `product2`, and `product3`. The quantity of `product2` is updated to `4`. An empty list named `inventory` is created and the three product dictionaries are appended to it.

Question 1 of 2

How to update the value of `a` from `dict_1` dictionary?

PYTHON

```
1 dict_1 = {"a":2}
```

`dict_1["a"] = 5` (radio)

`dict_1.update("a":5)` (radio)

Code Playground

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fc58bfa6-14ab-4a80-aed6-d23dc8286f

NXT
wAVES™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions (2 hours, 10 mins)

- Introduction to Python (25 mins)
- Conditional Statements and Loops (15 mins)
- List and Functions (25 mins)
- Tuples, Sets and Dictionaries (25 mins)
- OOPs (20 mins)
- Miscellaneous Topics (20 mins)
- More Interview Questions (20 mins)

Q. Create a variable inventory of data type as a list. Create 2 to 3 product dictionaries and add these products to the list.

Intent of the interviewer: The interviewer is expecting to create 2 to 3 dictionary and then update the value of one key from the dictionary.

PYTHON

```
9 - product2 = {  
10 -     "productID": 2,  
11 -     "prodName": "Mouse",  
12 -     "quantity": 2,  
13 -     "prodPrice": 100.00,  
14 -     "category": "Electronics"  
15 - }  
16 -  
17 - product3 = {  
18 -     "productID": 3,
```

Three product dictionaries are created: `product1`, `product2`, and `product3`. The quantity of `product2` is updated to 4. An empty list named `inventory` is created and the three product dictionaries are appended to it.

Question 1 of 2

How to update the value of `a` from `dict_1` dictionary?

PYTHON

```
1 dict_1 = {"a":2}
```

`dict_1["a"] = 5`

`dict_1.update("a":5)`

Revolutionizing the Job Market - X

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fc58bfa6-14ab-4a80-aed6-d23dc8286f Download Search Star Code Playground Jangipally Ranjith

**NXT
wAVES™**

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. Create a variable inventory of data type as a list. Create 2 to 3 product dictionaries and add these products to the list.

Intent of the interviewer: The interviewer is expecting to create 2 to 3 dictionary and then update the value of one key from the dictionary.

PYTHON

```
17 product3 = {  
18     "prodID": 3,  
19     "prodName": "Keyboard",  
20     "quantity": 3,  
21     "prodPrice": 50.00,  
22     "category": "Electronics"  
23 }  
24  
25 product2["quantity"] = 4  
26
```

Three product dictionaries are created: `product1`, `product2`, and `product3`. The quantity of `product2` is updated to `4`. An empty list named `inventory` is created and the three product dictionaries are appended to it.

Question 1 of 2

How to update the value of `a` from `dict_1` dictionary?

PYTHON

```
1 dict_1 = {"a":2}
```

`dict_1["a"] = 5` (Correct)

`dict_1.update("a":5)`

?

Revolutionizing the Job Market - X

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fc58bfa6-14ab-4a80-aed6-d23dc8286f Download Search Star Code Playground Jangipally Ranjith

**NXT
wAVES™**

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. Create a variable inventory of data type as a list. Create 2 to 3 product dictionaries and add these products to the list.

Intent of the interviewer: The interviewer is expecting to create 2 to 3 dictionary and then update the value of one key from the dictionary.

PYTHON

```
24
25 product2["quantity"] = 4
26
27 inventory = []
28 inventory.append(product1)
29 inventory.append(product2)
30 inventory.append(product3)
31
32 for product in inventory:
33     print(product)
```

Three product dictionaries are created: `product1`, `product2`, and `product3`. The quantity of `product2` is updated to `4`. An empty list named `inventory` is created and the three product dictionaries are appended to it.

Question 1 of 2

How to update the value of `a` from `dict_1` dictionary?

PYTHON

```
1 dict_1 = {"a":2}
```

`dict_1["a"] = 5` (1)

`dict_1.update("a":5)`

`dict_1.replace("a":5)`


Snipping Tool
Screenshot copied to clipboard and saved
Select here to mark up and share the image

Revolutionizing the Job Market - X

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fe98dc9f-bc9d-4ed9-b04d-765905c34a41 Download Search Star Print R More

**NXT
wAVES™** Search for 'Arrays' Profile Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Interview Questions

Recently Asked Questions and Answers

Q. What is OOPs?

Companies Asked: Uniserved

Object-Oriented Programming is a way of approaching, designing and developing software, so that the components of the software and the interactions between them resemble real-life objects and their interactions.

Proper usage of OOPs concepts helps us build well-organized systems that are easy to use and extend.

← Question 1 of 2

Given the object-oriented programming (OOP) concept, which of the following is an advantage of properly using OOP concepts?

- Reducing code length
- Making software resemble real-life interactions
- Increasing the number of classes used
- Making the software harder to extend

[Next](#)

Complete the MCQs to proceed Continue

Revolutionizing the Job Market - X

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fe98dc9f-bc9d-4ed9-b04d-765905c34a41 Download Search Star Print R More

**NXT
wAVES™** Search for 'Arrays' Profile Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Interview Questions

Recently Asked Questions and Answers

Q. What is OOPs?

Companies Asked: Uniserved

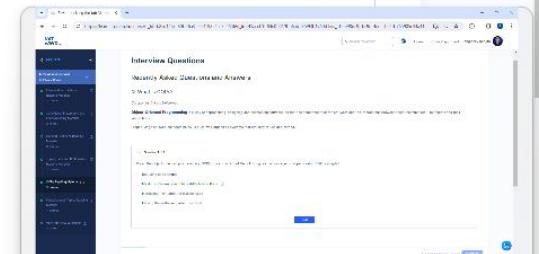
Object-Oriented Programming is a way of approaching, designing and developing software, so that the components of the software and the interactions between them resemble real-life objects and their interactions.

Proper usage of OOPs concepts helps us build well-organized systems that are easy to use and extend.

← Question 2 of 2

Which of the following best describes the benefits of using OOP concepts in software development?

- OOP helps create disorganized and hard-to-maintain systems.
- OOP makes it challenging to model real-life objects in software.
- OOP results in well-organized and easily extendable systems. (✓)
- OOP is not suitable for modern software development.



Snipping Tool

Screenshot copied to clipboard and saved
Select here to mark up and share the image

Revolutionizing the Job Market - X

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fe98dc9f-bc9d-4ed9-b04d-765905c34a41 Download Search Star Print R More

**NXT
wAVE™** Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

2. What are the advantages of OOPs?

The advantages of OOPs:

- Easier way to analyse and solve bugs
- Reusability of code through inheritance
- Effective problem solving
- Elimination of code redundancy

3. What are the principles of OOPs?

The principles of OOPs involve,

- Abstraction
- Encapsulation
- Inheritance
- Polymorphism

Abstraction in python is defined as a process of handling complexity by hiding unnecessary information from the user. This is one of the core concepts of object-oriented programming (OOP) languages.

Polymorphism is an OOPs concept. The word polymorphism means having many forms. It refers to the use of a single type entity (method, operator, or object) to represent different types in different scenarios.

Continue ...

Revolutionizing the Job Market - X

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fe98dc9f-bc9d-4ed9-b04d-765905c34a41 Download Search Star Copy Print R More

**NXT
wAVe™** Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Polymorphism is an OOPs concept. The word polymorphism means having many forms. It refers to the use of a single type entity (method, operator, or object) to represent different types in different scenarios.

Q. Where do we use OOP concepts?

Some of the applications of OOPs:

- **Software Design and Architecture:** OOP provides a clear structure for the software, making it easier to manage and understand complex systems.
- **Software Maintenance and Updates:** OOP encapsulates data and behavior within objects, simplifying code maintenance. This encapsulation allows for easy modification and extension of the existing code, reducing the risk of disrupting other parts of the software..
- **Object-oriented Databases:** These databases maintain the identity and integrity of real-world entities by establishing a direct correspondence between them and database objects.

← Question 1 of 4

Where do we primarily use Object-Oriented Programming (OOP) concepts?

Writing simple scripts

Designing and structuring software systems (✓)

Debugging code

None of the given options.

Next

Complete the MCQs to proceed Continue

Revolutionizing the Job Market - X

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fe98dc9f-bc9d-4ed9-b04d-765905c34a41 Download Search Star Code Playground Jangipally Ranjith

**NXT
wAVES™**

Search for 'Arrays' Home

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Polymorphism is an OOPs concept. The word polymorphism means having many forms. It refers to the use of a single type entity (method, operator, or object) to represent different types in different scenarios.

Q. Where do we use OOP concepts?

Some of the applications of OOPs:

- Software Design and Architecture:** OOP provides a clear structure for the software, making it easier to manage and understand complex systems.
- Software Maintenance and Updates:** OOP encapsulates data and behavior within objects, simplifying code maintenance. This encapsulation allows for easy modification and extension of the existing code, reducing the risk of disrupting other parts of the software..
- Object-oriented Databases:** These databases maintain the identity and integrity of real-world entities by establishing a direct correspondence between them and database objects.

Question 2 of 4

Which of the following is a benefit of using Object-Oriented Programming (OOP) in the context of software maintenance?

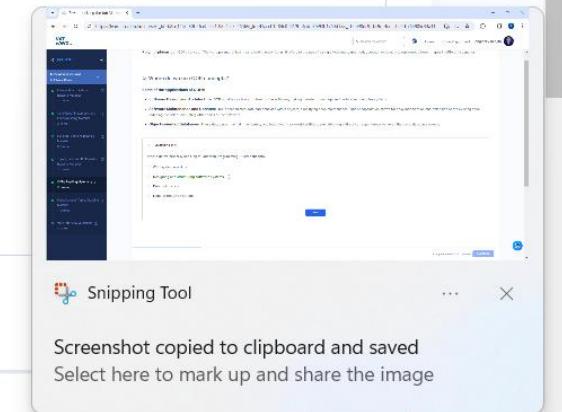
It makes debugging easier.

It reduces the need for comments in code.

It allows code to be organized into classes and objects. ✓

It automates the process of updating software.

Next



Screenshot copied to clipboard and saved
Select here to mark up and share the image

Revolutionizing the Job Market - X

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fe98dc9f-bc9d-4ed9-b04d-765905c34a41 Download  Search  Star  Copy  Refresh  More 

**NXT
wAVES™** Home  Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material  25 mins
- Conditional Statements and Loops Reading Material  15 mins
- List and Functions Reading Material  25 mins
- Tuples, Sets and Dictionaries Reading Material  25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material  20 mins
- More Interview Questions 20 mins

Polymorphism is an OOPs concept. The word polymorphism means having many forms. It refers to the use of a single type entity (method, operator, or object) to represent different types in different scenarios.

Q. Where do we use OOP concepts?

Some of the applications of OOPs:

- Software Design and Architecture:** OOP provides a clear structure for the software, making it easier to manage and understand complex systems.
- Software Maintenance and Updates:** OOP encapsulates data and behavior within objects, simplifying code maintenance. This encapsulation allows for easy modification and extension of the existing code, reducing the risk of disrupting other parts of the software..
- Object-oriented Databases:** These databases maintain the identity and integrity of real-world entities by establishing a direct correspondence between them and database objects.

Question 3 of 4

How does Object-Oriented Programming (OOP) help in Object-oriented database?

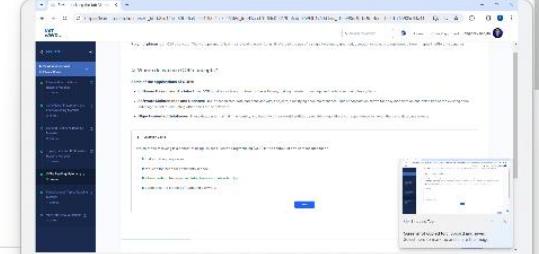
To hide the real-world entities.

To maintain a direct correspondence between the real world and database objects. 

To separate the database from the real world.

To create an entirely new representation of the real world.

Next


Snipping Tool
Screenshot copied to clipboard and saved
Select here to mark up and share the image

Revolutionizing the Job Market - X

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fe98dc9f-bc9d-4ed9-b04d-765905c34a41 Download Search Star Print R More

**NXT
wAVE™** Profile Picture

Search for 'Arrays' Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Polymorphism is an OOPs concept. The word polymorphism means having many forms. It refers to the use of a single type entity (method, operator, or object) to represent different types in different scenarios.

Q. Where do we use OOP concepts?

Some of the applications of OOPs:

- **Software Design and Architecture:** OOP provides a clear structure for the software, making it easier to manage and understand complex systems.
- **Software Maintenance and Updates:** OOP encapsulates data and behavior within objects, simplifying code maintenance. This encapsulation allows for easy modification and extension of the existing code, reducing the risk of disrupting other parts of the software..
- **Object-oriented Databases:** These databases maintain the identity and integrity of real-world entities by establishing a direct correspondence between them and database objects.

Question 4 of 4

How does Object-Oriented Programming (OOP) help in software system design?

It makes the code more complex and harder to manage.

It doesn't have any impact on software design.

It organizes code into classes and objects, making it easier to manage complex systems. ✓

It eliminates the need for code organization.

Continue Feedback

4. What are Classes?

A `class` is a prototype from which objects are created. Classes can be used to bundle related attributes and methods. An instance of a `class` is an Object.

Code

```
1+ class Mobile:
2+     def __init__(self, model, storage):
3         self.model = model
4         self.storage = storage
5
6
7 obj = Mobile("iPhone 12 Pro", "128GB")
8 print(obj.model)
```

Output

```
iPhone 12 Pro
```

Q. What are classes and objects, and what are the differences between them?

Class

Object

Revolutionizing the Job Market X +

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fe98dc9f-bc9d-4ed9-b04d-765905c34a41

Q. What are classes and objects, and what are the differences between them?

Class	Object
A Class is a blueprint or template for creating objects. It defines the attributes and methods that an object will have.	An object is an instance of a class, representing a specific example of the concept defined by the class.
Defining a class does not occupy any memory space.	Creating an object allocates memory space.
A class is defined once, but can be instantiated multiple times.	We can create objects as many times as we want according to our requirements.

▼ More Information

```
1+ class A(object):
2+     def show(self):
3+         print("A class method is called")
4+
5+ def checkmethod(clasmethod):
6+     clasmethod.show()
7+
8+ AObj = A()
9+ checkmethod(AObj)
```

- In the code above, `A` is a class that defines a method called `show()`. `AObj` is an instance (or object) of the class `A`, and we can call the `show()` method on `AObj`.

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fe98dc9f-bc9d-4ed9-b04d-765905c34a41

NXT
wAVES™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions (2 hours, 10 mins)

- Introduction to Python (Reading Material, 25 mins)
- Conditional Statements and Loops (Reading Material, 15 mins)
- List and Functions (Reading Material, 25 mins)
- Tuples, Sets and Dictionaries (Reading Material, 25 mins)
- OOPs (Reading Material, 20 mins)
- Miscellaneous Topics (Reading Material, 20 mins)
- More Interview Questions (20 mins)

Question 1 of 4

What is a class in object-oriented programming (OOP)?

- A class is an instance of an object.
- A class is a blueprint or template for creating objects.** (checked)
- A class is a specific example of an object.
- A class is a method that operates on objects.

Next

Complete the MCQs to proceed

Continue

Feedback icon

Revolutionizing the Job Market - X

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fe98dc9f-bc9d-4ed9-b04d-765905c34a41 Download Search Star Code Playground Jangipally Ranjith

**NXT
wAVe™**

Search for 'Arrays' Home

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Question 2 of 4

How many times can a class be defined in OOP?

A class can be defined multiple times.

A class can be defined only once. ✓

A class can be defined as many times as there are objects created.

A class can be defined as many times as needed for different methods.

Next

Snipping Tool

Screenshot copied to clipboard and saved
Select here to mark up and share the image

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fe98dc9f-bc9d-4ed9-b04d-765905c34a41

NXT
wAVES™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions (2 hours, 10 mins)

- Introduction to Python (Reading Material, 25 mins)
- Conditional Statements and Loops (Reading Material, 15 mins)
- List and Functions (Reading Material, 25 mins)
- Tuples, Sets and Dictionaries (Reading Material, 25 mins)
- OOPs (Reading Material, 20 mins)
- Miscellaneous Topics (Reading Material, 20 mins)
- More Interview Questions (20 mins)

Question 3 of 4

What does creating an object in OOP involve?

Creating an object does not allocate any memory space.

Creating an object defines the attributes and methods of a class.

Creating an object instantiates a specific class and allocates memory space.

Creating an object defines a blueprint for a class.

Next

Screenshot copied to clipboard and saved
Select here to mark up and share the image

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fe98dc9f-bc9d-4ed9-b04d-765905c34a41

NXT WAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions (2 hours, 10 mins)

- Introduction to Python (Reading Material, 25 mins)
- Conditional Statements and Loops (Reading Material, 15 mins)
- List and Functions (Reading Material, 25 mins)
- Tuples, Sets and Dictionaries (Reading Material, 25 mins)
- OOPs (Reading Material, 20 mins)
- Miscellaneous Topics (Reading Material, 20 mins)
- More Interview Questions (20 mins)

Question 4 of 4

Which statement is true regarding the memory allocation of classes and objects in OOP?

- Defining a class allocates memory space.
- Creating an object allocates memory space.
- Both defining a class and creating an object allocate memory space.
- Neither defining a class nor creating an object allocates memory space.

Screenshot copied to clipboard and saved
Select here to mark up and share the image

5. What is `__init__` / constructor in Python?

The `__init__` is a special method used to initialize values to attributes.

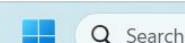
Code

```
1+ class Mobile:
2+     def __init__(self, model, camera):
3+         self.model = model
4+         self.camera = camera
5+     def make_call(self, number):
6+         print("calling..{}".format(number))
7+
8 mobile = Mobile("Nikon", "D850")
9 mobile.make_call("12345")
```

In the above example, the `model` and `camera` attributes are initialized with the values that are passed to the `__init__` method.

6. What is `self` in OOPs?

In Python, the `self` is the first parameter of methods that represents the instance of the class. Therefore, to call attributes and methods of a class, the programmer need to use `self` with `self` is not a keyword and has no special meaning in Python. Writing this parameter as `self` is a convention. We can use other names but it is highly discouraged.



6. What is `self` in OOPs?

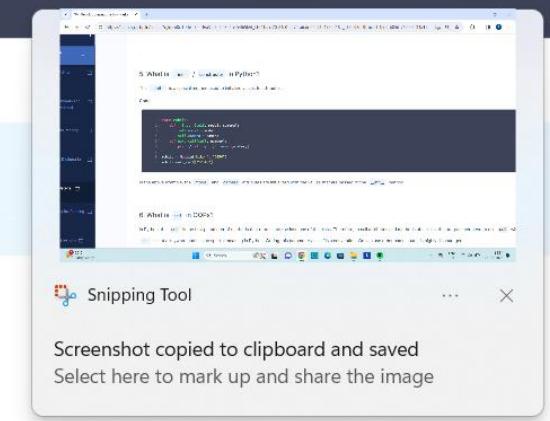
In Python, the `self` is the first parameter of methods that represents the instance of the class. Therefore, to call attributes and methods of a class, the programmer need to use `self` with `self` is not a keyword and has no special meaning in Python. Writing this parameter as self is a convention. We can use other names but it is highly discouraged.

Code

```
1- class Dog:  
2-     def __init__(self, name, age):  
3-         self.name = name  
4-         self.age = age  
5-     def info(self):  
6-         print("My name is {}".format(self.name))  
7-     def make_sound(self):  
8-         print("Bow Wow")
```

Output

```
My name is Rex
```



6. What is `self` in OOPs?

In Python, the `self` is the first parameter of methods that represents the instance of the class. Therefore, to call attributes and methods of a class, the programmer need to use `self` with `self` is not a keyword and has no special meaning in Python. Writing this parameter as self is a convention. We can use other names but it is highly discouraged.

Code

```
5
6+     def info(self):
7+         print("My name is {}".format(self.name))
8
9+     def make_sound(self):
10+        print("Bow Wow")
11
12 dog1 = Dog('Rex', 2)
13
14 dog1.info()
```

Output

```
My name is Rex
```

Revolutionizing the Job Market - +

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fe98dc9f-bc9d-4ed9-b04d-765905c34a41 Download Search Star Print R More

NXT WAVE Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

7. What are class methods?

Methods which need access to class attributes but not instance attributes are marked as class methods.
For class methods, we send `cls` as a parameter indicating we are passing the class.

Cart

```
items
min_bill
flat_discount
add_item(self)
update_flat_discount(cls)
```

Class Method

Code

PYTHON

```
1. class Cart:
2.     flat_discount = 0
3.     min_bill = 100
4.     @classmethod
5.     def update_flat_discount(cls, new_flat_discount):
6.         cls.flat_discount = new_flat_discount
7.
8. Cart.update_flat_discount(25)
9. print(Cart.flat_discount)
```

Output

```
25
```

`@classmethod` decorator marks the method below it as a class method.

Q. What is the difference between a Method and Function?

Continue

Revolutionizing the Job Market - X

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fe98dc9f-bc9d-4ed9-b04d-765905c34a41 Download Search Star Copy Print R More

**NXT
wAVe™** Search for 'Arrays' Profile Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. What is the difference between a Method and Function?

Function	Method
Functions have an independent existence. You can define them outside of the class.	Methods do not have an independent existence. They are always defined within a class.
Functions don't have any reference variables.	Methods are used to manipulate instance variables of a class.
Functions are called independently.	Methods are called using instance or object.

Question 1 of 3

How are methods typically called or invoked?

Methods are called independently.

Methods are called using reference variables. (Correct)

Methods are called using global variables.

Methods are called using local variables.

Next

Complete the MCQs to proceed Continue

Revolutionizing the Job Market - X

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fe98dc9f-bc9d-4ed9-b04d-765905c34a41 Download Search Star Print R More

**NXT
wAVe™** Search for 'Arrays' Profile Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. What is the difference between a Method and Function?

Function	Method
Functions have an independent existence. You can define them outside of the class.	Methods do not have an independent existence. They are always defined within a class.
Functions don't have any reference variables.	Methods are used to manipulate instance variables of a class.
Functions are called independently.	Methods are called using instance or object.

← Question 3 of 3

What is the primary purpose of a function?

- Functions are used to manipulate instance variables of a class.
- Functions are used to define classes.
- Functions are self-describing pieces of code. (✓)
- Functions are used to create objects.

Continue Help

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fe98dc9f-bc9d-4ed9-b04d-765905c34a41

NXT
wAVG™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. Create a class representing a student with properties for age, name, and gender.

PYTHON

```
1 - class Student:
2 -     def __init__(self, name, age, gender):
3 -         self.name = name
4 -         self.age = age
5 -         self.gender = gender
6 -
7 -     def get_name(self):
8 -         return self.name
9 -
10 -    def get_age(self):
```

← Question 1 of 3

The functions within the class are also called ...

- Properties
- Methods
- Operators
- Objects

Next

Complete the MCQs to proceed Continue

1 29°C Partly sunny

Search

12:36 14-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fe98dc9f-bc9d-4ed9-b04d-765905c34a41

NXT wAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions (2 hours, 10 mins)

- Introduction to Python (Reading Material, 25 mins)
- Conditional Statements and Loops (Reading Material, 15 mins)
- List and Functions (Reading Material, 25 mins)
- Tuples, Sets and Dictionaries (Reading Material, 25 mins)
- OOPs (Reading Material, 20 mins)
- Miscellaneous Topics (Reading Material, 20 mins)
- More Interview Questions (20 mins)

Q. Create a class representing a student with properties for age, name, and gender.

PYTHON

```
>
10 -     def get_age(self):
11 -         return self.age
12 -
13 -     def get_gender(self):
14 -         return self.gender
15 -
16 -     def print_student_info(self):
17 -         print(f"Student name: {self.name}")
18 -         print(f"Student age: {self.age}")
19 -         print(f"Student gender: {self.gender}")
```

Question 1 of 3

The functions within the class are also called ...

- Properties
- Methods
- Operators
- Objects

Next

Complete the MCQs to proceed

Continue

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fe98dc9f-bc9d-4ed9-b04d-765905c34a41

NXT wAVES™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions (2 hours, 10 mins)

- Introduction to Python (Reading Material, 25 mins)
- Conditional Statements and Loops (Reading Material, 15 mins)
- List and Functions (Reading Material, 25 mins)
- Tuples, Sets and Dictionaries (Reading Material, 25 mins)
- OOPs (Reading Material, 20 mins)
- Miscellaneous Topics (Reading Material, 20 mins)
- More Interview Questions (20 mins)

Q. Create a class representing a student with properties for age, name, and gender.

PYTHON

```
16 -     def print_student_info(self):
17 -         print(f"Student name: {self.name}")
18 -         print(f"Student age: {self.age}")
19 -         print(f"Student gender: {self.gender}")
20 -
21 student1 = Student("Ram", 16, "Male")
22 student2 = Student("Raheem", 19, "Male")
23 -
24 student1.print_student_info()
25 student2.print_student_info()
```

Question 1 of 3

The functions within the class are also called ...

- Properties
- Methods
- Operators
- Objects

Next

Snipping Tool

Screenshot copied to clipboard and saved
Select here to mark up and share the image

Revolutionizing the Job Market - X

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fe98dc9f-bc9d-4ed9-b04d-765905c34a41 Download Search Star Print R More

**NXT
wAVe™** Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

8. What is Inheritance?

Inheritance is a mechanism by which a class inherits attributes and methods from another class.

The class whose attributes and methods are inherited is known as the Super/Base/Parent Class. And the class that inherits the attributes and methods from the parent class is the Sub/Derived/Child Class.

9. Write an example program to show Inheritance?

Code

PYTHON

```
1- class Product:
2-     def __init__(self, name, price, deal_price, ratings):
3-         self.name = name
4-         self.price = price
5-         self.deal_price = deal_price
6-         self.ratings = ratings
7-         self.you_save = price - deal_price
8-     def display_product_details(self):
9-         print("Product: {}".format(self.name))
10-        print("Price: {}".format(self.price))
```

Output

```
Product: TV
Price: 45000
Deal Price: 40000
You Saved: 5000
Ratings: 3.5
```

In the above example, the child class **ElectronicItem** inherits all the attributes and methods from the parent class **Product**.

Continue Next

Revolutionizing the Job Market - X

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fe98dc9f-bc9d-4ed9-b04d-765905c34a41 Download Search Star Print R More

**NXT
wAVe™** Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

8. What is Inheritance?

Inheritance is a mechanism by which a class inherits attributes and methods from another class.

The class whose attributes and methods are inherited is known as the Super/Base/Parent Class. And the class that inherits the attributes and methods from the parent class is the Sub/Derived/Child Class.

9. Write an example program to show Inheritance?

Code

PYTHON

```
11     print("Deal Price: {}".format(self.deal_price))
12     print("You Saved: {}".format(self.you_save))
13     print("Ratings: {}".format(self.ratings))
14
15- class ElectronicItem(Product):
16-     def set_warranty(self, warranty_in_months):
17         self.warranty_in_months = warranty_in_months
18
19-     def get_warranty(self):
20         return self.warranty_in_months
```

Output

```
Product: TV
Price: 45000
Deal Price: 40000
You Saved: 5000
Ratings: 3.5
```

In the above example, the child class **ElectronicItem** inherits all the attributes and methods from the parent class **Product**.

Continue ...

Revolutionizing the Job Market - X

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fe98dc9f-bc9d-4ed9-b04d-765905c34a41 Download Search Star Code Playground Jangipally Ranjith

**NXT
wAVe™**

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

8. What is Inheritance?

Inheritance is a mechanism by which a class inherits attributes and methods from another class.

The class whose attributes and methods are inherited is known as the Super/Base/Parent Class. And the class that inherits the attributes and methods from the parent class is the Sub/Derived/Child Class.

9. Write an example program to show Inheritance?

Code

PYTHON

```
14
15 - class ElectronicItem(Product):
16 -     def set_warranty(self, warranty_in_months):
17 -         self.warranty_in_months = warranty_in_months
18 -
19 -     def get_warranty(self):
20 -         return self.warranty_in_months
21 -
22 e = ElectronicItem("TV", 45000, 40000, 3.5)
23 e.display_product_details()
```

Output

```
Product: TV
Price: 45000
Deal Price: 40000
You Saved: 5000
Ratings: 3.5
```

In the above example, the child class **ElectronicItem** inherits all the attributes and methods from the parent class **Product**.

Snipping Tool

Screenshot copied to clipboard and saved
Select here to mark up and share the image

Revolutionizing the Job Market - X

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fe98dc9f-bc9d-4ed9-b04d-765905c34a41 Download Search Star Print R More

**NXT
wAVe™** Search for 'Arrays' Profile Home Code Playground Jangipally Ranjith

CONTENTS

- Interview Questions** 2 hours, 10 mins
- Introduction to Python** Reading Material 25 mins
- Conditional Statements and Loops** Reading Material 15 mins
- List and Functions** Reading Material 25 mins
- Tuples, Sets and Dictionaries** Reading Material 25 mins
- OOPs** Reading Material 20 mins
- Miscellaneous Topics** Reading Material 20 mins
- More Interview Questions** 20 mins

10. What are the types of Inheritance?

There are 5 types of Inheritance in Python:

1. Single Inheritance
2. Multiple Inheritance
3. Multilevel Inheritance
4. Hierarchical Inheritance
5. Hybrid Inheritance

Single Inheritance:

```
graph TD; Parent[Parent] --> Child[Child]
```

When a Child class inherits attributes and methods from a single Parent class, it is called Single Inheritance.

Multiple Inheritance:

```
graph TD; Parent1[Parent 1] --- Child[Child]; Parent2[Parent 2] --- Child
```

When a Child class inherits attributes and methods from more than one Parent class, it is called Multiple Inheritance.

Multilevel Inheritance:

Continue

Revolutionizing the Job Market - X

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fe98dc9f-bc9d-4ed9-b04d-765905c34a41 Download Search Star Code Playground Jangipally Ranjith

**NXT
wAVe™**

When a Child class inherits attributes and methods from more than one Parent class, it is called Multiple Inheritance.

Multilevel Inheritance:

```
graph TD; Parent[Parent] --> Child1[Child 1]; Child1 --> Child2[Child 2]
```

When a Child class inherits attributes and methods from another Child class, it is called Multilevel Inheritance.

Hierarchical Inheritance:

```
graph TD; Parent[Parent] --> Child1[Child 1]; Parent --> Child2[Child 2]
```

When more than one Child class inherits attributes and methods from a single Parent class, it is called Hierarchical Inheritance.

Hybrid Inheritance:

```
graph TD; Parent[Parent]
```

Screenshot copied to clipboard and saved
Select here to mark up and share the image



Search



ENG
IN



12:37
14-02-2024



Revolutionizing the Job Market - X

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fe98dc9f-bc9d-4ed9-b04d-765905c34a41 Download Search Star Code Playground Jangipally Ranjith

**NXT
wAVe™**

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. What is meant by multiple inheritance?

When a class can inherit attributes and methods from more than one parent class Multiple inheritance.

▼ More Information

Example:

```
1- class Animal:
2-     def make_sound(self):
3-         pass
4-
5- class Bird(Animal):
6-     def fly(self):
7-         print("Flying")
8-
9- class Mammal(Animal):
10-    def run(self):
```

PYTHON

Explanation:

- Animal Class:**
`Animal` is a base class that has a method `make_sound`.
- Bird Class:**
`Bird` is a subclass of `Animal` and has an additional method `fly` which prints `Flying`. It inherits the `make_sound` method from the `Animal` class.
- Mammal Class:**
`Mammal` is another subclass of `Animal` and has a method `run` which prints `Running`. Like `Bird`, it also inherits the `make_sound` method from the `Animal` class.
- Bat Class:**
`Bat` is a subclass that inherits from both `Bird` and `Mammal`. It implements the `make_sound` method, providing a concrete implementation by printing `Screech`.

Continue

Revolutionizing the Job Market - X

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fe98dc9f-bc9d-4ed9-b04d-765905c34a41 Download Search Star Print R More

**NXT
wAVe™** Search for 'Arrays' Profile Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

11. What is Encapsulation?

While modelling objects with object-oriented programming, we bundle related information together to clearly separate information of different objects.

The bundling of related attributes and methods together is called Encapsulation.

Classes can be used to bundle related attributes and methods.

12. What is Method Overriding?

Method Overriding is an OOPs concept related to Inheritance. When a child class method overrides the parent class method of the same name, parameters and return type, it is known as Method Overriding.

Method Overriding allows us to change the implementation of a function in the child class that is defined in the parent class.

Code

PYTHON

```
1: class Product:
2:
3:     def __init__(self, name, price, deal_price, rating):
4:         self.name = name
5:         self.price = price
6:         self.deal_price = deal_price
7:         self.rating = rating
8:         self.you_save = price - deal_price
9:
10:    def display_product_details(self):
```

Output

```
Product: Laptop
Price: 45000
Deal Price: 40000
You Saved: 5000
Rating: 3.5
```

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fe98dc9f-bc9d-4ed9-b04d-765905c34a41

NXT wAVES™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Code PYTHON

```
9
10 -     def display_product_details(self):
11 -         print("Product: {}".format(self.name))
12 -         print("Price: {}".format(self.price))
13 -         print("Deal Price: {}".format(self.deal_price))
14 -         print("You Saved: {}".format(self.you_save))
15 -         print("Rating: {}".format(self.rating))
16 -
17 -     def get_deal_price(self):
18 -         return self.deal_price
```

Output

```
Product: Laptop
Price: 45000
Deal Price: 40000
You Saved: 5000
Rating: 3.5
Warranty 10 months
```

In the above example, the `display_product_details()` method in the **ElectronicItem** class overrides the `display_product_details()` method of the **Product** class.

Continue

Ask doubt

Notes Discussions

Notes + NEW NOTE

B I U S E M T Take a Note

Take a Note

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fe98dc9f-bc9d-4ed9-b04d-765905c34a41

NXT wAVe™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Code PYTHON

```
18     return self.deal_price
19
20- class ElectronicItem(Product):
21
22-     def display_product_details(self):
23         super().display_product_details()
24         print("Warranty {} months".format(self.warranty_in_months))
25
26-     def set_warranty(self, warranty_in_months):
27         self.warranty_in_months = warranty_in_months
```

Output

```
Product: Laptop
Price: 45000
Deal Price: 40000
You Saved: 5000
Rating: 3.5
Warranty 10 months
```

In the above example, the `display_product_details()` method in the `ElectronicItem` class overrides the `display_product_details()` method of the `Product` class.

Continue

Ask doubt

Notes Discussions

Notes + NEW NOTE

B I U S E ⌂ ⌂ Take a Note

Take a Note

Revolutionizing the Job Market - X

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fe98dc9f-bc9d-4ed9-b04d-765905c34a41 Download Search Star Print R More

**NXT
wAVe™** Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

PYTHON

```
25
26 -     def set_warranty(self, warranty_in_months):
27 -         self.warranty_in_months = warranty_in_months
28
29 -     def get_warranty(self):
30 -         return self.warranty_in_months
31
32 e = ElectronicItem("Laptop", 45000, 40000, 3.5)
33 e.set_warranty(10)
34 e.display_product_details()
```

Output

```
Product: Laptop
Price: 45000
Deal Price: 40000
You Saved: 5000
Rating: 3.5
Warranty: 10 months
```

In the above example, the `display_product_details()` method in the **ElectronicItem** class overrides the `display_product_details()` method of the **Product** class.

Continue

Ask doubt

Notes Discussions

Notes + NEW NOTE

B I U S E ⌂ ⌂ ⌂

Title
Take a Note

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fe98dc9f-bc9d-4ed9-b04d-765905c34a41

NXT wAVES™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. What are Decorators in python?

Companies Asked: Translytics Business Services

In Python, decorators are a flexible way to modify or extend the behavior of functions or methods without changing their code. Decorators let us add extra stuff to functions because functions can be treated just like any other object, like numbers or strings. This means we can pass them as arguments, return them from other functions, or even change them.

A decorator is a function that takes another function (or method) as an argument and returns a new function that usually extends or modifies the behavior of the given function. The original function or method is not altered; instead, its behavior is changed when it's called through its decorated version.

Basic Syntax:

Here's a simple example to illustrate the concept of decorators:

PYTHON

```
1 def my_decorator(func):
2     def wrapper():
3         print("Something is happening before the function is called.")
4         func()
5         print("Something is happening after the function is called.")
6     return wrapper
7
8 @my_decorator
9 def say_hello():
10    print("Hello!")
```

When you run the above code, the output will be:

```
1 Something is happening before the function is called.
2 Hello!
3 Something is happening after the function is called.
```

The `@my_decorator` syntax is just an easier way of saying `say_hello = my_decorator(say_hello)`.

Continue

Revolutionizing the Job Market - X

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fe98dc9f-bc9d-4ed9-b04d-765905c34a41 Download Search Star Print R More

**NXT
wAVe™** Search for 'Arrays' Profile Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. What are Decorators in python?

Companies Asked: Translytics Business Services

In Python, decorators are a flexible way to modify or extend the behavior of functions or methods without changing their code. Decorators let us add extra stuff to functions because functions can be treated just like any other object, like numbers or strings. This means we can pass them as arguments, return them from other functions, or even change them.

A decorator is a function that takes another function (or method) as an argument and returns a new function that usually extends or modifies the behavior of the given function. The original function or method is not altered; instead, its behavior is changed when it's called through its decorated version.

Basic Syntax:

Here's a simple example to illustrate the concept of decorators:

```
3     print("Something is happening before the function is called.")
4     func()
5     print("Something is happening after the function is called.")
6     return wrapper
7
8 @my_decorator
9 def say_hello():
10     print("Hello!")
11
12 say_hello()
```

When you run the above code, the output will be:

```
1 Something is happening before the function is called.
2 Hello!
3 Something is happening after the function is called.
```

The `@my_decorator` syntax is just an easier way of saying `say_hello = my_decorator(say_hello)`.

Screenshot copied to clipboard and saved Select here to mark up and share the image

Revolutionizing the Job Market - X

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fe98dc9f-bc9d-4ed9-b04d-765905c34a41 Download Search Star Print R More

**NXT
wAVES™** Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. What is init function in python?

Companies Asked: Bluecloud

The "init" function in Python refers to the special method called `__init__`. It is also known as the constructor method, and it is used to initialize the attributes (or properties) of an object when an instance of a class is created. This method is automatically called when you create a new object from a class, and it allows you to set up the initial state of the object.

Here's a brief example of how the `__init__` method works in Python:

PYTHON

```
2 -     def __init__(self, parameter1, parameter2):
3 -         self.parameter1 = parameter1
4 -         self.parameter2 = parameter2
5 -
6 -     # Creating an instance of MyClass and passing values to the __init__ method
7 -     my_instance = MyClass("Value1", "Value2")
8 -
9 -     # Accessing the attributes of the object
10 -    print(my_instance.parameter1) # Output: "Value1"
11 -    print(my_instance.parameter2) # Output: "Value2"
```

In this example, the `__init__` method is defined with two parameters, and when an instance of the "MyClass" class is created, these parameters are used to initialize the object's attributes.

◀ **Question 1 of 4**

What is the value of `my_instance.parameter1` immediately after executing the line `my_instance = MyClass("Value1", "Value2")` ?

"Value1" "Value2"

Complete the MCQs to proceed Continue

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fe98dc9f-bc9d-4ed9-b04d-765905c34a41

NXT wAVE™

Search for 'Arrays' Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions (2 hours, 10 mins)

- Introduction to Python (25 mins)
- Conditional Statements and Loops (15 mins)
- List and Functions (25 mins)
- Tuples, Sets and Dictionaries (25 mins)
- OOPs (20 mins)
- Miscellaneous Topics (20 mins)
- More Interview Questions (20 mins)

Q. How do you define a class in Python?

Companies Asked: Uniserved

A class in Python is defined using the `class` keyword, followed by the class name and a colon. Inside the class, attributes and methods can be defined to represent properties and behaviors of the class.

Here's a simple example of a class representing a `Dog`:

```
1+ class Dog:
2+     species = "Canine" # class attribute
3+
4+     def __init__(self, name, age):
5+         self.name = name # instance attribute
6+         self.age = age   # instance attribute
7+
8+     def bark(self):
9+         return f"{self.name} says Woof!"
```

You can create instances of this class and call its methods:

```
1 dog1 = Dog("Buddy", 3)
2 dog2 = Dog("Lucy", 5)
3
4 print(dog1.bark()) # Output: Buddy says Woof!
5 print(dog2.bark()) # Output: Lucy says Woof!
6 print(Dog.species) # Output: Canine
```

The class provides a blueprint for creating objects, and each object can have its own values for the attributes defined in the class.

Continue

Revolutionizing the Job Market - X

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fe98dc9f-bc9d-4ed9-b04d-765905c34a41 Download Search Star Print R More

**NXT
wAVe™** Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- **Introduction to Python** Reading Material 25 mins
- **Conditional Statements and Loops** Reading Material 15 mins
- **List and Functions** Reading Material 25 mins
- **Tuples, Sets and Dictionaries** Reading Material 25 mins
- **OOPs** Reading Material 20 mins
- **Miscellaneous Topics** Reading Material 20 mins
- **More Interview Questions** 20 mins

Q. Given a class in Python, how can you access its data members?

We can access a class's data members using the following approaches:

- **Using self**
- **Using Object**
- **Using Class**

Class:

PYTHON

```
1 - class Cart:
2   flat_discount = 0
3   min_bill = 100
4 - def __init__(self):
5   self.items = {}
6 - def add_item(self, item_name, quantity):
7   self.items[item_name] = quantity
8 - def display_items(self):
9   print(self.items)
10
```

Continue CHAT

Revolutionizing the Job Market - X

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fe98dc9f-bc9d-4ed9-b04d-765905c34a41 Download Search Star Print R More

**NXT
wAVe™** Search for 'Arrays' Profile Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- **Introduction to Python** Reading Material 25 mins
- **Conditional Statements and Loops** Reading Material 15 mins
- **List and Functions** Reading Material 25 mins
- **Tuples, Sets and Dictionaries** Reading Material 25 mins
- **OOPs** Reading Material 20 mins
- **Miscellaneous Topics** Reading Material 20 mins
- **More Interview Questions** 20 mins

Q. Given a class in Python, how can you access its data members?

We can access a class's data members using the following approaches:

- **Using self**
- **Using Object**
- **Using Class**

Class:

PYTHON

```
10
11 a = Cart()
12
13 a.add_item("book", 3)
14 a.display_items()          # Accessing using Self
15                                         # Output : {'book': 3}
16
17 a.add_item("pamplate", 6)   # Accessing using Object
18 print(a.items())           # Output : {'book': 3, 'pamplate': 6}
```

Continue Feedback

NXT WAVE™ Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fe98dc9f-bc9d-4ed9-b04d-765905c34a41

Search for 'Arrays' Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. Given a class in Python, how can you access its data members?

We can access a class's data members using the following approaches:

- Using self
- Using Object
- Using Class

Class:

```
13 a.add_item("book", 3)          # Accessing using Self
14 a.display_items()              # Output : {'book': 3}
15
16
17 a.add_item("pamplate", 6)      # Accessing using Object
18 print(a.items())              # Output : {'book': 3, 'pamplate': 6}
19
20
21 print(Cart.min_bill)          # Accessing using Class
22 # Output : 100
```

PYTHON

Screenshot copied to clipboard and saved Select here to mark up and share the image

Revolutionizing the Job Market - X

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fe98dc9f-bc9d-4ed9-b04d-765905c34a41 Download Search Star Code Playground Jangipally Ranjith

NXT wAVE™

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

{'Pamplete': 7}

Q. What is polymorphism?

Polymorphism, meaning "many forms," is a fundamental concept in object-oriented programming (OOP) that allows a single entity (method, operator, or object) to take on multiple forms. This means that the same operation or method can have different behaviors depending on the object. Polymorphism promotes flexibility and code reusability by allowing developers to write code that can handle different types of objects in a consistent way.

Polymorphism is often used in Class methods, where we can have multiple classes with the same method name.

Example:

We have two classes: A and B with same method named show .

PYTHON

```
1- class Animal:
2-     def make_sound(self):
3-         print("Generic animal sound")
4-
5- class Dog(Animal):
6-     def make_sound(self):
7-         print("Woof!")
8-
9- class Cat(Animal):
10-    def make_sound(self):
```

▼ More Information

There are two primary ways in which polymorphism is implemented in Python:

- **Method Overriding:** This occurs when a subclass defines a method with the same name as a method in its parent class, but with a different implementation.
- **Operator Overloading:** Python allows operators to be overloaded, meaning that their behavior can be customized for different object types. This is commonly used for arithmetic operators, such as +, -, *, and /.

Continue Feedback

Revolutionizing the Job Market - X

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=fe98dc9f-bc9d-4ed9-b04d-765905c34a41 Download Search Star Copy Print R More

**NXT
wAVe™** Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. What is polymorphism?

Polymorphism, meaning "many forms," is a fundamental concept in object-oriented programming (OOP) that allows a single entity (method, operator, or object) to take on multiple forms. This means that the same operation or method can have different behaviors depending on the object. Polymorphism promotes flexibility and code reusability by allowing developers to write code that can handle different types of objects in a consistent way.

Polymorphism is often used in Class methods, where we can have multiple classes with the same method name.

Example:

We have two classes: `A` and `B` with same method named `show`.

PYTHON

```
8
9 - class Cat(Animal):
10 -     def make_sound(self):
11 -         print("Meow!")
12 -
13 dog = Dog()
14 cat = Cat()
15 -
16 dog.make_sound() # Output: Woof!
17 cat.make_sound() # Output: Meow!
```

More Information

There are two primary ways in which polymorphism is implemented in Python:

- **Method Overriding:** This occurs when a subclass defines a method with the same name as a method in its parent class, but with a different implementation.
- **Operator Overloading:** Python allows operators to be overloaded, meaning that their behavior can be customized for different object types. This is commonly used for arithmetic operators, such as `+`, `-`, `*`, and `/`.

Continue

Revolutionizing the Job Market - X

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=0d4d02db-bd07-42c2-9304-86718345a729 Download Search Star Print R More

**NXT
wAVE™** Search for 'Arrays' Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python** Reading Material 25 mins
- Conditional Statements and Loops** Reading Material 15 mins
- List and Functions** Reading Material 25 mins
- Tuples, Sets and Dictionaries** Reading Material 25 mins
- OOPs** Reading Material 20 mins
- Miscellaneous Topics** Reading Material 20 mins
- More Interview Questions** 20 mins

Interview Questions > Miscellaneous Topics Reading Material

1. What is an Object in Python?

In general, anything that can be assigned to a variable in Python is referred to as an **object**.

Strings, Integers, Floats, Lists, Functions, Modules etc... are all objects.

"A"

1.25

1 2 3

Powered by

Identity of an Object

Whenever an object is created in Python, it will be given a **unique identifier (id)**. This unique id can be different each time you run the program.

"A"

Id - 140035229724336

1 2 3

Revolutionizing the Job Market - +

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=0d4d02db-bd07-42c2-9304-86718345a729 Download Search Star Print R More

**NXT
wAVe™** Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python** Reading Material 25 mins
- Conditional Statements and Loops** Reading Material 15 mins
- List and Functions** Reading Material 25 mins
- Tuples, Sets and Dictionaries** Reading Material 25 mins
- OOPs** Reading Material 20 mins
- Miscellaneous Topics** Reading Material 20 mins
- More Interview Questions** 20 mins

wAVe™ IB HUBS

Identity of an Object

Whenever an object is created in Python, it will be given a **unique identifier (id)**. This unique id can be different each time you run the program.

"A"

Id - **140035229724336**

1 2 3

Id - **139630925071104**

Powered by **NXT
wAVe™** IB HUBS

Every object that you use in a Python program will be stored in Computer Memory

The unique id will be related to the location where the object is stored in the Computer Memory.

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=0d4d02db-bd07-42c2-9304-86718345a729

NXT
wAVe™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPS Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Every object that you use in a Python program will be stored in Computer Memory

The unique id will be related to the location where the object is stored in the Computer Memory.

Q. Have you used Python libraries?

I have used Python a few libraries like:

- math:** This library provides a wide range of mathematical functions, including basic operations, exponentiation, logarithms, trigonometric and hyperbolic functions, special functions, and constants.
- random:** This library provides random number generation, random selections from lists and strings, and list shuffling.
- Datetime:** The Python Datetime module provides classes for manipulating dates and times, along with a number of functions for performing operations on these objects.

Question 1 of 4

Considering the given code what is the value of num ?

PYTHON

```
1 import random
2 num = random.randint(1, 10)
```

A number between 1 and 10 (inclusive)

A floating point number between 1 and 10

A random boolean value

None of the above

Complete the MCQs to proceed Continue

1 29°C Sunny

Search

12:45 14-02-2024 ENG IN

Revolutionizing the Job Market - X

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=0d4d02db-bd07-42c2-9304-86718345a729 Download Search Star Print R More

**NXT
wAVES™** Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

2. What are Modules in Python?

In the Python context, any file containing a Python code is called a **Module**. Some examples of modules are collections, random, datetime, math, etc...

3. What are Packages in Python?

In the Python context, any file containing a Python code is called a Module. A Package is a collection of modules.

Q. Can you show me today's date?

We can obtain the current date using Python's built-in `datetime` library. We have used `today()` method defined in the `datetime` class to get a date object containing the current local date.

```
1 import datetime
2
3 today = datetime.date.today()
4 print(today)
```

PYTHON

When we run this code, it'll print the current date in the format of `YYYY-MM-DD`.

Continue Next

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=0d4d02db-bd07-42c2-9304-86718345a729

NXT
wAVe™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. Can you give me the date 1 week from today?

To get the date one week from today, we can use the `timedelta()` method in Python's `datetime` library.

```
1 import datetime
2
3 today = datetime.date.today()
4 one_week_from_today = today + datetime.timedelta(days=7)
5 print(one_week_from_today)
```

PYTHON

When we run this code, it'll print the date one week from the current date.

Question 1 of 4

Which of the following classes from the datetime module is used to represent a duration?

- date
- time
- timedelta
- datetime

Next

Complete the MCQs to proceed Continue

DOW -1.35%

Search

12:46 14-02-2024

Revolutionizing the Job Market - X

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=0d4d02db-bd07-42c2-9304-86718345a729 Download Search Star Copy Print R More

**NXT
wAVES™** Search for 'Arrays' Profile Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. Can you give me the date 1 week from today?

"To get the date one week from today, we can use the `timedelta()` method in Python's `datetime` library.

PYTHON

```
1 import datetime
2
3 today = datetime.date.today()
4 one_week_from_today = today + datetime.timedelta(days=7)
5 print(one_week_from_today)
```

When we run this code, it'll print the date one week from the current date.

Question 2 of 4

Given the code: `future_date = today + datetime.timedelta(days=5)`. What will be the value of `future_date` if `today` is `2023-10-25` ?

PYTHON

```
1 import datetime
2
3 today = datetime.date.today()
4 future_date = today + datetime.timedelta(days=5)
5 print(future_date)
```

2023-10-20

2023-10-30

2023-10-15

2023-10-31

Next Complete the MCQs to proceed Continue

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=0d4d02db-bd07-42c2-9304-86718345a729

NXT wAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions (2 hours, 10 mins)

- Introduction to Python (Reading Material, 25 mins)
- Conditional Statements and Loops (Reading Material, 15 mins)
- List and Functions (Reading Material, 25 mins)
- Tuples, Sets and Dictionaries (Reading Material, 25 mins)
- OOPs (Reading Material, 20 mins)
- Miscellaneous Topics (Reading Material, 20 mins)
- More Interview Questions (20 mins)

Q. Give me a date 1 week before today?

To get the date before one week from today using Python's `datetime` library, we make use of the `timedelta()` method defined in the `datetime` class to get a date.

`PYTHON`

```
1 import datetime
2
3 today = datetime.date.today()
4 one_week_from_today = today - datetime.timedelta(days=7)
5 print(one_week_from_today)
```

When we run this code, it'll print the date before one week from the current date.

Question 1 of 3

What will be the format of the output after executing the following code?

`PYTHON`

```
1 print(datetime.date.today())
```

yyyy-mm-dd
 dd-mm-yyyy
 mm-yyyy-dd
 dd-yyyy-mm

Next

Complete the MCQs to proceed

Continue

Revolutionizing the Job Market - X

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=0d4d02db-bd07-42c2-9304-86718345a729 Download Search Star Copy Print R More

NXT
wAVES™ Home Code Playground Jangipally Ranjith

Search for 'Arrays'

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. What are Namespaces?

Rephrased Question: **What is Namespace in Python?**

A **namespace** is a collection of currently defined names along with information about the object that the name references.

▼ More Information

Namespace ensures that names are **unique** and won't lead to any conflict.

Namespace **a** points to "Hello" and **b** points to "World".
Id - 140589419285168 Id - 140589419285182

Namespaces allow us to have the same name referring different things in **different namespaces**.

Namespace **a** points to "Hey" and **b** points to "Guys".

Namespace **a** points to "Hello" and **b** points to "World".

Continue

Revolutionizing the Job Market - +

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=0d4d02db-bd07-42c2-9304-86718345a729 Download Search Star Code Playground Jangipally Ranjith

**NXT
wAVe™**

Search for 'Arrays'

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Namespaces allow us to have the same name referring different things in **different namespaces**.

Namespace

a b

"Hey" "Guys"

Id - [140584419285168](#)

Namespace

a b

"Hello" "World"

Id - [140589419285168](#)

Code:

```
1 def greet_1():
2     a = "Hello"
3     print(a)
4     print(id(a))
5
6 def greet_2():
7     a = "Hey"
8     print(a)
9     print(id(a))
10
```

Output:

```
Namespace - 1
Hello
140639382368176
```

PYTHON

Powered by IB HUBS

Continue

Revolutionizing the Job Market - X

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=0d4d02db-bd07-42c2-9304-86718345a729 Download Search Star Copy Print R More

NXT WAVE™ Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

"Hey" Id - [140584419285168](#)

"Guys" Id - [140580419285182](#)

"Hello" Id - [140589419285168](#)

"World" Id - [140589419285182](#)

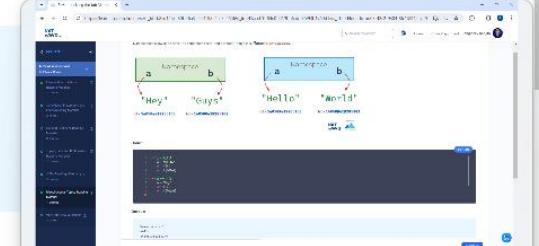
Code:

```
5
6 def greet_2():
7     a = "Hey"
8     print(a)
9     print(id(a))
10
11 print("Namespace - 1")
12 greet_1()
13 print("Namespace - 2")
14 greet_2()
```

PYTHON

Output:

```
Namespace - 1
Hello
140639382368176
Namespace - 2
Hey
140639382570608
```

 Snipping Tool

Screenshot copied to clipboard and saved
Select here to mark up and share the image

Revolutionizing the Job Market - +

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=0d4d02db-bd07-42c2-9304-86718345a729 Download Search Star Print R More

**NXT
wAVe™** Search for 'Arrays' Profile Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

5. What is Scope?

The scope of a variable is the region in which that variable can be accessed.

6. What are Global and Local variables?

Global Variables:

If a variable is declared outside of all functions and conditional statements then that variable is known as **Global variable**.

These can be accessed anywhere in the program. If the value of the global variable is modified inside a function or conditional statement then the changes are reflected in the rest of the program.

Local Variables:

If a variable is declared inside of a function or conditional statement then that variable is known as **Local variable**.

These can be accessed inside a function or conditional statement in which they are declared. If the value of the local variable is modified in one function, then the changes are not reflected in another function.

Continue Next

Revolutionizing the Job Market - X

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=0d4d02db-bd07-42c2-9304-86718345a729 Download Search Star Print R More

**NXT
wAVES™** Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. How to Handle Exceptions in Python?

Rephrased Question: What is a try block in Python?

Python provides **try block** way to catch the exceptions that were raised so that they can be properly handled.

- Exceptions can be handled with **try-except** block.
- Whenever an exception occurs at some line in the **try** block, the execution stops at that line and jumps to except block.

1 - **try:**
2 - # Write the code that might cause exceptions
3 - **except:**
4 - # The code to be run when there is an exception.

PYTHON

Code

```
1 - try:  
2 -     print(x)  
3 - except:  
4 -     print("x is not defined")
```

PYTHON

Output

x is not defined

► **More information**

Continue

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=0d4d02db-bd07-42c2-9304-86718345a729

NXT
wAVES™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

x is not defined

More information

Handling Specific Exceptions

We can specifically mention the **name of the exception** to catch all exceptions of that specific type.

Syntax

```
try:  
    # Write code that might cause exceptions  
except Exception:  
    # The code to be run when there is an exception
```

PYTHON

Example

Code

```
try:  
    print(x)  
except NameError:  
    print("x is not defined")  
except:  
    print("Uncaught Exception")
```

PYTHON

Output

```
x is not defined
```

Handling Multiple Exceptions

Continue

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=0d4d02db-bd07-42c2-9304-86718345a729

NXT
wAVe™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Handling Multiple Exceptions

We can write **multiple exception blocks** to handle different types of exceptions differently.

Syntax

```
try:  
    # Write code that might cause exceptions  
except Exception1:  
    # The code to be run when there is an exception1  
except Exception2:  
    # The code to be run when there is an exception2
```

PYTHON

Example_1

Code

```
try:  
    a = int(input())  
    b = int(input())  
    c = a/b  
    print(c)  
except ZeroDivisionError:  
    print("Denominator can't be 0")  
except ValueError:  
    print("Input should be an integer")  
except:
```

PYTHON

Input

Rahul
Teja

Output

Continue

Revolutionizing the Job Market - X

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=0d4d02db-bd07-42c2-9304-86718345a729 Download Search Star Print R More

**NXT
wAVe™** Search for 'Arrays' Profile Home Code Playground Jangipally Ranjith

5- except Exception2:
6- # The code to be run when there is an exception

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Example_1

Code PYTHON

```
2-     a = int(input())
3-     b = int(input())
4-     c = a/b
5-     print(c)
6- except ZeroDivisionError:
7-     print("Denominator can't be 0")
8- except ValueError:
9-     print("Input should be an integer")
10- except:
11-     print("Something went wrong")
```

Input

```
Rahul
Teja
```

Output

```
Input should be an integer
```

Example_2

Code PYTHON

Continue

Revolutionizing the Job Market - X

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=0d4d02db-bd07-42c2-9304-86718345a729 Download Search Star Print R More

**NXT
wAVe™** Search for 'Arrays' Profile Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions Reading Material 20 mins

Input should be an integer

Example_2

Code

PYTHON

```
1 try:
2     a = int(input())
3     b = int(input())
4     c = a/b
5     print(c)
6 except ZeroDivisionError:
7     print("Denominator can't be 0")
8 except ValueError:
9     print("Input should be an integer")
10 except:
```

Input

```
1
0
```

Output

```
Denominator can't be 0
```

Continue

Revolutionizing the Job Market - X

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=0d4d02db-bd07-42c2-9304-86718345a729 Download Search Star Print R More

**NXT
wAVES™** Search for 'Arrays' Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Input should be an integer

Example_2

Code

PY PYTHON

```
2     a = int(input())
3     b = int(input())
4     c = a/b
5     print(c)
6 except ZeroDivisionError:
7     print("Denominator can't be 0")
8 except ValueError:
9     print("Input should be an integer")
10 except:
11     print("Something went wrong")
```

Input

```
1
0
```

Output

```
Denominator can't be 0
```

Sn Snipping Tool X

Sc Screenshot copied to clipboard and saved Select here to mark up and share the image

Revolutionizing the Job Market - X

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=0d4d02db-bd07-42c2-9304-86718345a729 Download Search Star Print R More

**NXT
wAVe™** Search for 'Arrays' Profile Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. What are errors and exceptions in Python? Write a logic to handle exceptions.

In Python, errors and exceptions are issues that arise during the execution of a program that disrupt the normal flow of the program's instructions.

- Errors:** These are problems in the program that the interpreter identifies before execution starts. They are typically syntax errors where the code does not follow to the rules of the Python language.
- Exceptions:** These are errors detected during execution that can be handled by the program, allowing the program to continue running.

To handle the logic exceptions we use `try` and `except` blocks.

PYTHON

```
1- def divide(x, y):
2-     try:
3-         result = x / y
4-     except ZeroDivisionError:
5-         return "Error: Division by zero is not allowed."
6-     except TypeError:
7-         return "Error: Only numbers can be divided."
8-     else:
9-         return result
10-
```

The `try` block attempts division. If successful, the result is stored in `result`.
Except blocks handle errors:

- ZeroDivisionError for division by zero, returning `Error: Division by zero is not allowed.`
- TypeError for non-numeric operands, returning `Error: Only numbers can be divided`.

Continue

Revolutionizing the Job Market - X

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=0d4d02db-bd07-42c2-9304-86718345a729 Download Search Star Print R More

**NXT
wAVe™** Search for 'Arrays' Profile Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. What are errors and exceptions in Python? Write a logic to handle exceptions.

In Python, errors and exceptions are issues that arise during the execution of a program that disrupt the normal flow of the program's instructions.

- **Errors:** These are problems in the program that the interpreter identifies before execution starts. They are typically syntax errors where the code does not follow to the rules of the Python language.
- **Exceptions:** These are errors detected during execution that can be handled by the program, allowing the program to continue running.

To handle the logic exceptions we use `try` and `except` blocks.

PYTHON

```
4-     except ZeroDivisionError:
5-         return "Error: Division by zero is not allowed."
6-     except TypeError:
7-         return "Error: Only numbers can be divided."
8-     else:
9-         return result
10-
11    print(divide(10, 2))
12    print(divide(10, 0)) # Error: Division by zero is not allowed.
13    print(divide("10", 2)) # Error: Only numbers can be divided.
```

The try block attempts division. If successful, the result is stored in `result`.
Except blocks handle errors:

- `ZeroDivisionError` for division by zero, returning `Error: Division by zero is not allowed.`
- `TypeError` for non-numeric operands, returning `Error: Only numbers can be divided`.

Continue CHAT

Revolutionizing the Job Market - X

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=0d4d02db-bd07-42c2-9304-86718345a729 Download Search Star Print R More

**NXT
wAVe™** Search for 'Arrays' Profile Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. What is the difference between Shallow Copy and Deep Copy?

Companies Asked: Translytics Business Services

Shallow Copy and Deep Copy are terms used to copy objects, like lists or dictionaries in Python.

Shallow Copy:

A shallow copy creates a new object (like a list or dictionary), but it doesn't create new copies of the items inside. Instead, it just points to the same items that the original container has. If you change one of these items, it will change in both the original and the copied container.

Example with Lists:

```
1 import copy
2
3 original_list = [[1, 2, 3], [4, 5, 6]]
4 shallow_copied_list = copy.copy(original_list)
5
6 # Modify an element in the shallow copied list
7 shallow_copied_list[0][0] = 99
8
9 print(original_list) # Output: [[99, 2, 3], [4, 5, 6]]
```

In this example, modifying `shallow_copied_list` also modified `original_list`, because the shallow copy only copied the references to the sublists, not the sublists themselves. The `copy` module, which provides methods to create shallow and deep copies of objects.

Deep Copy:

A deep copy creates a new object (like a list or dictionary) and also recursively creates entirely new objects for each of the elements inside this object. This means the copied object and its elements are entirely independent of the original. Any modifications made to the elements of the copied object won't affect the original object's elements, and vice versa.

Example with Lists:

```
1 import copy
2
```

Python Java C/C++ JavaScript C# Go Ruby PHP Python Java C/C++ JavaScript C# Go Ruby <span style="border: 1px solid blue; border-radius: 50%; padding: 2px 5px; margin-right: 1

Revolutionizing the Job Market X +

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=0d4d02db-bd07-42c2-9304-86718345a729

Deep Copy:

A deep copy creates a new object (like a list or dictionary) and also recursively creates entirely new objects for each of the elements inside this object. This means the copied object and its elements are entirely independent of the original. Any modifications made to the elements of the copied object won't affect the original object's elements, and vice versa.

Example with Lists:

```
1 import copy
2
3 original_list = [[1, 2, 3], [4, 5, 6]]
4 deep_copied_list = copy.deepcopy(original_list)
5
6 # Modify an element in the deep copied list
7 deep_copied_list[0][0] = 99
8
9 print(original_list) # Output: [[1, 2, 3], [4, 5, 6]]
```

Here, modifying `deep_copied_list` did not affect `original_list`.

In Summary:

Attribute	Shallow Copy	Deep Copy
Definition	Copies the outer object and references to the inner objects.	Copies both the outer and inner objects.
Inner Object Changes	Changes in inner objects reflect in both the original and the copy.	Changes in inner objects don't affect the other.
Independence	The copy is not fully independent from the original.	The copy is entirely independent from the original.
Usage in Python	<code>copy.copy(original)</code>	<code>copy.deepcopy(original)</code>
Example Impact	Modifying a sublist in the copied list can affect the original list.	Modifying a sublist in the copied list has no impact on the original list.

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=0d4d02db-bd07-42c2-9304-86718345a729

NXT wAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions

2 hours, 10 mins

- Introduction to Python
- Conditional Statements and Loops
- List and Functions
- Tuples, Sets and Dictionaries
- OOPs
- Miscellaneous Topics
- More Interview Questions

25 mins 15 mins 25 mins 25 mins 20 mins 20 mins 20 mins

Q. Write the program to check if the given string is a Palindrome?

Companies Asked: Translytics Business Services

Below is the program to check if the given string is a Palindrome:

```
1- def is_palindrome(string):
2-     reversed_string = string[::-1]
3-
4-     if string == reversed_string:
5-         return True
6-     else:
7-         return False
8-
9- string = input()
10- if is_palindrome(string):
```

PYTHON

Question 1 of 5

What is the correct code to reverse a string using slicing in Python?

string[::-1] string[::1] string[-1::-1] string[1:-1]

Next

Complete the MCQs to proceed Continue

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=0d4d02db-bd07-42c2-9304-86718345a729

NXT WAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPS Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. Write the program to check if the given string is a Palindrome?

Companies Asked: Translytics Business Services

Below is the program to check if the given string is a Palindrome:

```
4 -     if string == reversed_string:
5 -         return True
6 -     else:
7 -         return False
8 -
9 - string = input()
10 - if is_palindrome(string):
11 -     print("The string is a palindrome.")
12 - else:
13 -     print("The string is not a palindrome.")
```

PYTHON

Question 1 of 5

What is the correct code to reverse a string using slicing in Python?

- string[::-1]
- string[:::-1]
- string[-1::-1]
- string[1::-1]

Next

Screenshot copied to clipboard and saved Select here to mark up and share the image

Revolutionizing the Job Market - X

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=0d4d02db-bd07-42c2-9304-86718345a729 Download Search Star Print R More

**NXT
wAVe™** Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. Write a code to group the data by method and calculate the sum of the values using pandas.

Given data:

```
1 data = []
2   'period' :['31-03-2023','30-04-2023','31-05-2023','30-06-2023','31-07-2023','30-08-2023']
3   'method':['pr1','pr1','pr2','pr1','pr2','pr2'],
4   'value':[2125,150,6400,1000,4000,2749]
5 
```

PYTHON

Companies Asked: Translytics Business Services

To group the data by the 'method' column and calculate the sum of the values in the 'value' column, one can use the pandas library in Python, a powerful tool for data analysis. Here is how to go about it using the `groupby` and `sum` functions:

```
1 import pandas as pd
2
3 data = {
4   'period' :['31-03-2023','30-04-2023','31-05-2023','30-06-2023','31-07-2023','30-08-2023']
5   'method':['pr1','pr1','pr2','pr1','pr2','pr2'],
6   'value':[2125,150,6400,1000,4000,2749]
7 }
8
9 df = pd.DataFrame(data)
10 result = df.groupby('method')['value'].sum().reset_index()
```

PYTHON

In this code:

- Pandas is imported as `pd`.
- The given data is structured into a dictionary.
- A DataFrame is created from the dictionary using `pd.DataFrame`.
- The `groupby` function is used to group the data by the 'method' column, and the sum of the 'value' column is then calculated using the `sum` function.
- The `reset_index` function is used to reset the index of the DataFrame to make it more presentable.
- The result is printed to display the summed values for each method group.

Continue

Revolutionizing the Job Market - X

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=0d4d02db-bd07-42c2-9304-86718345a729 Download Search Star Code Playground Jangipally Ranjith

NXT wAVe™

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions Reading Material 20 mins

Companies Asked: Translytics Business Services

To group the data by the 'method' column and calculate the sum of the values in the 'value' column, one can use the pandas library in Python, a powerful tool for data analysis. Here is how to go about it using the `groupby` and `sum` functions:

PYTHON

```
1 - data = [
2 -     'period' :['31-03-2023','30-04-2023','31-05-2023','30-06-2023','31-07-2023','30-08-2023'],
3 -     'method':['pr1','pr1','pr2','pr1','pr2','pr2'],
4 -     'value':[2125,150,6400,1000,4000,2749]
5 - ]
```

```
3 - data = {
4 -     'period' :['31-03-2023','30-04-2023','31-05-2023','30-06-2023','31-07-2023','30-08-2023']
5 -     'method':['pr1','pr1','pr2','pr1','pr2','pr2'],
6 -     'value':[2125,150,6400,1000,4000,2749]
7 - }
8 -
9 - df = pd.DataFrame(data)
10 result = df.groupby('method')['value'].sum().reset_index()
11
12 print(result)
```

In this code:

- Pandas is imported as `pd`.
- The given data is structured into a dictionary.
- A DataFrame is created from the dictionary using `pd.DataFrame`.
- The `groupby` function is used to group the data by the 'method' column, and the sum of the 'value' column is then calculated using the `sum` function.
- The `reset_index` function is used to reset the index of the DataFrame to make it more presentable.
- The result is printed to display the summed values for each method group.

Upon running the code, one will get the sum of the 'value' grouped by the 'method' column.

Continue

Revolutionizing the Job Market - X

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=0d4d02db-bd07-42c2-9304-86718345a729 Download Search Star Print R More

**NXT
wAVe™** Search for 'Arrays' Profile Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. What are Pandas and Numpy?

Companies Asked: Translytics Business Services

NumPy is a powerful Python library for mathematical and logical operations. It provides a large variety of useful features for multi-dimensional arrays, along with creating arrays of random numbers, variety of linear algebra functions with a huge community and has been extensively documented. It is widely used in scientific computing, engineering, and data analysis.

Pandas is a powerful Python library for data manipulation and analysis. It allows us to import data from a variety of formats (like CSV, Excel, SQL, etc.) and convert data into many formats. Its features enable tasks such as cleaning, transforming, and analyzing data efficiently.

Q. How much do you rate yourself in Python?

Companies Asked: Translytics Business Services

I'd rate myself **6** out of **10** in Python. I have grasped fundamental concepts such as variables, data types, and data structures, including lists, tuples, sets, and dictionaries, along with a few concepts from object-oriented programming (OOP).

I have improved my problem-solving skills through consistent practice and solving numerous problems. Currently, I am focusing on further developing these skills by working with various libraries such as Pandas and NumPy.

Q. What is Debugging?

Debugging is the process of identifying, analyzing, and resolving issues or errors in a software program. It's a critical part in the development process, as it helps to ensure the program runs as expected. Debugging typically involves finding and resolving `logical errors`, `syntax errors`, or `runtime errors`.

Continue

Revolutionizing the Job Market - X

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=0d4d02db-bd07-42c2-9304-86718345a729 Download Search Star Print R More

**NXT
wAVe™** Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. What is the difference between map and filter?

- map():** The map() function in Python takes in a function and an iterable as arguments. It iterates over the entire iterable, applies the provided function to each item, and returns a new iterable with the results.
- filter():** The filter() function in Python also takes a function and an iterable as arguments. However, it constructs a new iterable by applying the provided function to each item and including it only if the function returns `True`.

More Information

Using map() function:

```
1 - def newFunc(a):
2 -     return a*a
3 - x = map(newFunc, (1,2,3,4))
4 - print(x)
5 - print(list(x)) #Output: <map object at 0x7f42f4faa170> \n [1, 4, 9, 16]
```

Using filter() function:

```
1 - def func(x):
2 -     if x>=3:
3 -         return x
4 - y = filter(func, (1,2,3,4))
5 - print(y)
6 - print(list(y)) #Output: <filter object at 0x7fc51276b60> \n [3, 4]
```

PYTHON PYTHON

Continue

Revolutionizing the Job Market - X

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=c3d44d4a-c0e7-4160-818a-4be81541d132 Download Search Star Print R More

**NXT
wAVES™** Search for 'Arrays' Profile Home Code Playground Jangipally Ranjith 

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. What is list comprehension in python?

Intent of the interviewer: Interviewer is expecting the candidate to know about list comprehension and should have done some hands-on on that.

List comprehension is a fast, short, and simpler way to create lists compared to other iterative methods, like `for` loops.

list comprehension in Python includes three elements:

1. **expression** is the member itself, a call to a method, or any other valid expression that returns a value.
2. **member** is the object or value in the list or iterable.
3. **iterable** is a list, set, sequence, generator, or any other object that can return its elements one at a time.

Example:

PYTHON

```
1 squares = [i * i for i in range(10)]
2 squares
```

Output:

```
[0, 1, 4, 9, 16, 25, 36, 49, 64, 81]
```

Here

- `i * i` is the expression
- `i` is the member
- `range(10)` is the iterable

Continue ...

Revolutionizing the Job Market X +

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=c3d44d4a-c0e7-4160-818a-4be81541d132

1
2 
3

Q. Write the code for list comprehension, for getting even numbers from 1 to 10 in Python.

PYTHON

```
1 even_numbers = [x for x in range(1, 11) if x % 2 == 0]
2 print(even_numbers)
```

Explanation:

This list comprehension creates a list of even numbers from 1 to 10. It iterates through the range, and checks if the numbers divisible by 2. If they are divisible by 2, they are added to the even_numbers list.

← Question 2 of 2

Which of the following code snippets correctly prints even numbers?

PYTHON

```
1 numbers = [x for x in range(1,6) if x%2 != 0]
2 print(numbers)
```

PYTHON

Revolutionizing the Job Market - X

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=c3d44d4a-c0e7-4160-818a-4be81541d132 Download Search Star Print R More

**NXT
wAVES™** Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. Imagine we have a list containing 200 numbers and 200 alphanumeric values. How can we create a list in Python that contains only the alphabetical values?

Intent of the interviewer: Interviewer wants you to solve this problem with a list comprehension.

Solution:

PYTHON

```
1 original_list = ["abc123", "def456", "ghi789", "jklmno", "pqr456", "stu789", "vwx123", "yz"]
2
3 alphabet_only_list = [value for value in original_list if value.isalpha()]
4
5 print(alphabet_only_list)
```

For each 'value' in `original_list`, check if `value` contains only alphabetic characters. If the condition satisfied add `value` to the `alphabet_only_list`.

- `isalpha()` method is used to checks if a string contains only alphabetic characters.

Another approach

PYTHON

```
1
2 original_list = ["abc123", "def456", "ghi789", "jklmno", "pqr456", "stu789", "vwx123", "yz"]
3
4 alphabet_only_list = []
5
6 for value in original_list:
7     is_alpha = True
8     for char in value:
9         if not char.isalpha():
10             is_alpha = False
11
12     if is_alpha:
13         alphabet_only_list.append(value)
14
15 print(alphabet_only_list)
```

For each string `value` in the `original_list`, assume the string is made up of alphabets only. For each character `char` in the string `value`, if `char` is not an alphabetic character, Set `is_alpha` to False. Break out of the inner loop (no need to check remaining characters), and If `is_alpha` is still True after checking all characters, add the value to the `alphabet_only_list`.

- `isalpha()` method is used to checks if a string contains only alphabetic characters.

Continue ...

Revolutionizing the Job Market - X

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=c3d44d4a-c0e7-4160-818a-4be81541d132 Download Search Star Code Playground Jangipally Ranjith

**NXT
wAVES™**

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. Imagine we have a list containing 200 numbers and 200 alphanumeric values. How can we create a list in Python that contains only the alphabetical values?

Intent of the interviewer: Interviewer wants you to solve this problem with a list comprehension.

Solution:

```
1 original_list = ["abc123", "def456", "ghi789", "jklmno", "pqr456", "stu789", "vwxyz123", "yz"]  
2  
3 alphabet_only_list = [value for value in original_list if value.isalpha()]  
4  
5 print(alphabet_only_list)
```

For each 'value' in `original_list`, check if `value` contains only alphabetic characters. If the condition satisfied add `value` to the `alphabet_only_list`.

- `isalpha()` method is used to checks if a string contains only alphabetic characters.

Another approach

```
6 for value in original_list:  
7     is_alpha = True  
8     for char in value:  
9         if not char.isalpha():  
10             is_alpha = False  
11             break  
12     if is_alpha:  
13         alphabet_only_list.append(value)  
14  
15 print(alphabet_only_list)
```

For each string `value` in the `original_list`, assume the string is made up of alphabets only. For each character `char` in the string `value`, if `char` is not an alphabetic character, set `is_alpha` to False and break out of the inner loop (no need to check remaining characters), and if `is_alpha` is still True after checking all characters, add the value to the `alphabet_only_list`.

- `isalpha()` method is used to checks if a string contains only alphabetic characters.

Screenshot copied to clipboard and saved Select here to mark up and share the image

Revolutionizing the Job Market - X

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=c3d44d4a-c0e7-4160-818a-4be81541d132 Download Search Star Print R More

**NXT
wAVES™** Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. Write a program to create an array?

- By importing array module.

The array module defines the following type:

```
array.array(typecode['datatype', initializer])
```

```
1 import array
2 numbers = array.array('i',[10,20,30])
3 print(numbers) # Output: array([10, 20, 30])
```

Another Approach

- In Python we can also use list to create array

```
1 array = [1, 2, 3, 4, 5]
2
3 print(array) #Output: [1, 2, 3, 4, 5]
```

Question 1 of 2

What will be the output after executing the following code?

```
PYTHON
1 import array
2 numbers = array.array('i',[10,30,20])
3 print(numbers)
```

array('i', [10,30,20]) ✓

Complete the MCQs to proceed Continue

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=c3d44d4a-c0e7-4160-818a-4be81541d132

NXT
wAVES™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions

2 hours, 10 mins

- Introduction to Python
- Conditional Statements and Loops
- List and Functions
- Tuples, Sets and Dictionaries
- OOPs
- Miscellaneous Topics
- More Interview Questions

Q. Write a program to find sum of even number from a list of numbers.

Intent of the interviewer: The interviewer is expecting you to do this question using list comprehension and with an alternative approach which is by loops.

Using List Comprehension:

```
1 input_numbers = input()
2
3 numbers = [int(num) for num in input_numbers.split()]
4 even_numbers = [num for num in numbers if num % 2 == 0]
5
6 sum_of_numbers = sum(even_numbers)
7 print(sum_of_numbers)
```

Explanation:

The code takes a string of numbers as input and convert it to a list of integers and stores them in `numbers`. Another list `even_numbers` is created which has only even numbers based on the condition `num % 2 == 0`. The `sum()` function is then applied to `even_numbers` to calculate the sum of these even numbers.

Alternative Approach

Using Loops:

```
1 input_numbers = input()
2
3 numbers = [int(num) for num in input_numbers.split()]
4 even_numbers = 0
5
6 for num in numbers:
7     if num % 2 == 0:
8         even_numbers += num
9
10 print(even_numbers)
```

Explanation:

Continue

Revolutionizing the Job Market - X

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=c3d44d4a-c0e7-4160-818a-4be81541d132 Download Search Star Code Playground Jangipally Ranjith

**NXT
wAVe™**

Search for 'Arrays' Home

CONTENTS

Interview Questions < 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Interview Questions > More Interview Questions

Using List Comprehension: PYTHON

```
1 input_numbers = input()
2
3 numbers = [int(num) for num in input_numbers.split()]
4 even_numbers = [num for num in numbers if num % 2 == 0]
5
6 sum_of_numbers = sum(even_numbers)
7 print(sum_of_numbers)
```

Explanation:

The code takes a string of numbers as input and converts it to a list of integers and stores them in `numbers`. Another list `even_numbers` is created which has only even numbers based on the condition `num % 2 == 0`. The `sum()` function is then applied to `even_numbers` to calculate the sum of these even numbers.

Alternative Approach

Using Loops: PYTHON

```
1 input_numbers = input()
2
3 numbers = [int(num) for num in input_numbers.split()]
4 even_numbers = 0
5
6 for num in numbers:
7     if num % 2 == 0:
8         even_numbers += num
9
10 print(even_numbers)
```

Explanation:

The code takes a string of numbers as input and converts it to a list of integers and stores them in `numbers`. The code initializes a variable `even_numbers` to `0`. This variable will be used to store the sum of even numbers. It then iterates through the elements in the `numbers` list using a for loop. For each number, it checks if it's even. If it is, the number is added to the `even_numbers` variable.

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=c3d44d4a-c0e7-4160-818a-4be81541d132

NXT wAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions (2 hours, 10 mins)

- Introduction to Python (Reading Material, 25 mins)
- Conditional Statements and Loops (Reading Material, 15 mins)
- List and Functions (Reading Material, 25 mins)
- Tuples, Sets and Dictionaries (Reading Material, 25 mins)
- OOPs (Reading Material, 20 mins)
- Miscellaneous Topics (Reading Material, 20 mins)
- More Interview Questions (20 mins)

Q. what is queue and stack in python?

Both stacks and queues are linear data structures.

Stacks follow the LIFO (Last In First Out) principle, This means that the most recently added element is always at the top of the stack and is the first element to be removed.

There are multiple operations that are performed on the stack elements like insertion, deletion, finding the peek element, and determining whether the stack is empty or not. The insertion in a stack is called a push , whereas the deletion in a stack is called a pop .

Queues follow the FIFO (First In First Out) principle, meaning that the first element added to the queue is the first element removed.

Similar to Stacks, they are also used to perform insertions, and deletions, find the peak element, and check whether the queue is empty or not. However, insertion into a queue is called enqueue , whereas deletion from a queue is called dequeue .

Question 1 of 4

What does the pop method without any arguments denote when applied on a list used as a stack?

- Remove and return the last added item.
- Remove and return the first added item.
- Remove all items.
- Remove and return a random item.

Next

Complete the MCQs to proceed

Continue

Revolutionizing the Job Market - X

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=c3d44d4a-c0e7-4160-818a-4be81541d132 Download Search Star Print R More

**NXT
wAVES™** Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. Can we implement a queue using Python?

Yes we can implement queue in Python using list.

Python lists can be used to implement a basic queue. You can use the `append()` method to enqueue elements at the back of the list and the `pop(0)` method to dequeue elements from the front of the list.

PYTHON

```
1 - class Queue:
2 -     def __init__(self):
3 -         self.items = []
4 -
5 -     def isEmpty(self):
6 -         return len(self.items) == 0
7 -
8 -     def enqueue(self, item):
9 -         self.items.append(item)
10 -
```

▼ More Information

Operations on Queue:

- **isEmpty()** method determines whether the list is empty or not.
- **enqueue()** method adds an element to the end of the list.
- **dequeue()** method removes and returns the first element of the list.
- **peek()** Returns the first element or None if empty.
- **printQueue()** method prints the elements of the list.

Continue

Revolutionizing the Job Market - X

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=c3d44d4a-c0e7-4160-818a-4be81541d132 Download Search Star Print R More

**NXT
wAVES™** Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. Can you write a program to implement Stack in Python?

Python lists can be used to implement a basic stack. We can use the `append()` method to push the elements inside the list and the `pop()` method to remove the elements from the end of the list.

PYTHON

```
1 class Stack:
2     def __init__(self):
3         self.stack = []
4
5     def push(self, item):
6         self.stack.append(item)
7
8     def pop(self):
9         if not self.stack:
10             return None
```

▼ More Information

Operations on Stack:

- **push:** Adds an item to the top of the stack.
- **pop:** Removes the topmost element from the stack.
- **peek:** Returns the topmost element of the stack without removing it.
- **is_empty:** Checks whether the stack is empty.
- **size:** Returns the number of elements in the stack.

Continue Next

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=c3d44d4a-c0e7-4160-818a-4be81541d132

NXT
wAVe™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions (2 hours, 10 mins)

- Introduction to Python Reading Material (25 mins)
- Conditional Statements and Loops Reading Material (15 mins)
- List and Functions Reading Material (25 mins)
- Tuples, Sets and Dictionaries Reading Material (25 mins)
- OOPs Reading Material (20 mins)
- Miscellaneous Topics Reading Material (20 mins)
- More Interview Questions (20 mins)

Q. Where stack is important?

A stack is a data structure that stores and retrieves elements in a Last In First Out (LIFO) order. Stacks are used in many different applications, including:

- Function calls and recursion:** When a function is called, its information is pushed onto a stack called the call stack. When the function returns, its information is popped off the stack. This allows the computer to keep track of which functions are currently running and to return to the correct caller when each function finishes.
- Undo and redo features:** Stacks can be used to implement undo and redo features in software programs. When an action is performed, the state of the program is pushed onto a stack. If the user wants to undo the action, the previous state can be popped off the stack and restored.
- String reversal:** A stack can be used to reverse a string by pushing each character of the string onto the stack and then popping them off in reverse order.

Question 1 of 2

When you wish to implement the "Undo" functionality in a software program, which data structure would be most suitable?

- Array
- Graph
- Stack
- Hashtable

Next

Complete the MCQs to proceed Continue

Breaking news Unfolding now

Search

13:18 14-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=c3d44d4a-c0e7-4160-818a-4be81541d132

NXT
wAVe™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions (2 hours, 10 mins)

- Introduction to Python Reading Material (25 mins)
- Conditional Statements and Loops Reading Material (15 mins)
- List and Functions Reading Material (25 mins)
- Tuples, Sets and Dictionaries Reading Material (25 mins)
- OOPs Reading Material (20 mins)
- Miscellaneous Topics Reading Material (20 mins)
- More Interview Questions (20 mins)

Q. What are the Data Structures?

Rephrased Question: Are you familiar with data structures in Python?

Data Structures allow us to store and organize data efficiently. This will allow us to easily access and perform operations on the data.

In Python, there are four built-in data structures

- List*
- Tuple*
- Set*
- Dictionary*

Q. What are non linear data structure?

Intent: Interviewer is expecting detailed information regarding non-linear data structures.

Non-linear data structures in Python are used to organize data in a hierarchical way, where each element can have multiple children. This is unlike linear data structures, which organize data in a sequential order.

Some most commonly used non linear data structures are:

- Tree
- Graph

Some uses of non-linear datastructure:

1. Files management: Tree data structures organize and manage file systems by representing each file and directory as a node in a tree, with parent-child relationships indicating the hierarchy.
2. Decision making: In a decision tree, each internal node represents a test on a predictive variable, and each edge represents a possible outcome of that test. Leaf nodes represent the final outcome of all tests on a path.

Continue

Breaking news Unfolding now

Search

13:18 14-02-2024 ENG IN

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=c3d44d4a-c0e7-4160-818a-4be81541d132

NXT
wAV™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions (2 hours, 10 mins)

- Introduction to Python Reading Material (25 mins)
- Conditional Statements and Loops Reading Material (15 mins)
- List and Functions Reading Material (25 mins)
- Tuples, Sets and Dictionaries Reading Material (25 mins)
- OOPs Reading Material (20 mins)
- Miscellaneous Topics Reading Material (20 mins)
- More Interview Questions (20 mins)

Q. What is a Tree data structure?
Companies Asked: Fretron

A tree data structure is a non-linear data structure that is used to represent hierarchical relationships. It starts with a root node and branches out into child nodes, forming a tree-like structure. Each node can have zero or more child nodes, and it is used to represent hierarchical relationships and organize data efficiently.

A binary search tree (BST) is a common example of a tree data structure. In a BST, each node has at most two children, and it is used to efficiently store and search for data. The left child of a node contains values smaller than the node, and the right child contains values larger than the node, making it suitable for fast searching and sorting operations.

Question 1 of 4

What does the left child of a node in a binary search tree (BST) typically contain?

- Values larger than the node
- Values smaller than the node
- Values equal to the node
- No values

Next

Complete the MCQs to proceed Continue

DOW -1.35%

Search

13:19 14-02-2024 ENG IN

Revolutionizing the Job Market - X

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=c3d44d4a-c0e7-4160-818a-4be81541d132 Download Search Star Print R More

**NXT
wAVe™** Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. What is the purpose of tree data structure?

Rephrase question: Why tree data structure used?

A tree data structure helps us to organise data hierarchically, with each node having root node (top element of the tree) and one or more child nodes.

Trees are used in various areas because they provide efficient solutions for storing, searching, and sorting data.

▼ More information

- 1. Storing Hierarchical Data:** Trees are great for storing data that naturally forms a hierarchy. For example, the file system on a computer is a tree structure, with directories (folders) that contain files or other directories.
- 2. Searching:** Trees, especially binary search trees, are used when fast searching of data is required. They allow for efficient searching algorithms because on each step, they reduce the search space by half.

The tree data structure is used in various areas where hierarchical data representation, efficient searching, sorting, or networking is required.

Continue

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=c3d44d4a-c0e7-4160-818a-4be81541d132

NXT
wAVES™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions

2 hours, 10 mins

- Introduction to Python
- Conditional Statements and Loops
- List and Functions
- Tuples, Sets and Dictionaries
- OOPs
- Miscellaneous Topics
- More Interview Questions

25 mins 15 mins 25 mins 25 mins 20 mins 20 mins 20 mins

Q. Implement tree in Python?

A tree in Python is a hierarchical data structure where `nodes` are connected by `edges`. Each node contains a `value` and `zero` or more `child nodes`, if there has no child then it will be considered as `Leaf node`.

PYTHON

```
1- class TreeNode:
2-     def __init__(self, value):
3-         self.value = value
4-         self.left = None
5-         self.right = None
6-
7-     def inorder_traversal(node):
8-         if node:
9-             inorder_traversal(node.left)
10            print(node.value, end=' ')
```

Representation:

```
1
/\ 
2 3
/\ 
4 5
```

Continue

Revolutionizing the Job Market - X

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=c3d44d4a-c0e7-4160-818a-4be81541d132 Download Search Star Code Playground Jangipally Ranjith

**NXT
wAVe™**

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. Implement tree in Python?

A tree in Python is a hierarchical data structure where `nodes` are connected by `edges`. Each node contains a `value` and `zero` or more `child nodes`, if there has no child then it will be considered as `Leaf node`.

PYTHON

```
10     print(node.value, end=' ')
11     inorder_traversal(node.right)
12
13 root = TreeNode(1)
14 root.left = TreeNode(2)
15 root.right = TreeNode(3)
16 root.left.left = TreeNode(4)
17 root.left.right = TreeNode(5)
18
19 inorder_traversal(root) # Output: 4 2 5 1 3
```

Representation:

```
1
/\ 
2 3
/\ 
4 5
```

Continue

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=c3d44d4a-c0e7-4160-818a-4be81541d132

NXT
wAVES™

CONTENTS

Interview Questions

2 hours, 10 mins

- Introduction to Python
- Conditional Statements and Loops
- List and Functions
- Tuples, Sets and Dictionaries
- OOPs
- Miscellaneous Topics
- More Interview Questions

25 mins

15 mins

25 mins

25 mins

20 mins

20 mins

20 mins

Q. What is the binary search algorithm and how does it work?

Companies Asked: Fretron

Binary search is an efficient algorithm used to find a specific value in a sorted list or array. It works by repeatedly dividing the search range in half and comparing the middle element to the target value. If they match, the search is successful. If not, it narrows down the search to the appropriate half. This process continues until the target is found or the search range is empty, making it a highly effective way to locate items in sorted data sets.

Code:

PYTHON

```
1 def binary_search(arr, target):
2     left, right = 0, len(arr) - 1
3
4     while left <= right:
5         mid = (left + right) // 2 # Calculate the middle index
6         mid_val = arr[mid] # Get the middle element
7
8         if mid_val == target:
9             return mid # Target found, return its index
10        elif mid_val < target:
```

In this code, the `binary_search` function takes a sorted array `arr` and a `target` value as input. It uses a while loop to repeatedly divide the search range and compare the middle element with the target value until the target is found or the search range becomes empty. If the target is found, it returns the index of the target; otherwise, it returns `-1` to indicate that the target is not in the list.

Continue

Revolutionizing the Job Market +

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=c3d44d4a-c0e7-4160-818a-4be81541d132

NXT wAVES™

Search for 'Arrays' Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. What is the binary search algorithm and how does it work?

Companies Asked: Fretron

Binary search is an efficient algorithm used to find a specific value in a sorted list or array. It works by repeatedly dividing the search range in half and comparing the middle element to the target value. If they match, the search is successful. If not, it narrows down the search to the appropriate half. This process continues until the target is found or the search range is empty, making it a highly effective way to locate items in sorted data sets.

Code:

```
def binary_search(arr, target):  
    left = 0  
    right = len(arr) - 1  
  
    while left <= right:  
        mid = (left + right) // 2  
        mid_val = arr[mid]  
  
        if mid_val == target:  
            return mid  
        elif mid_val < target:  
            left = mid + 1 # Target is in the right half  
        else:  
            right = mid - 1 # Target is in the left half  
  
    return -1 # Target not found in the array  
  
# Example usage:  
my_list = [1, 3, 5, 7, 9, 11, 13, 15]  
target_value = 7
```

In this code, the `binary_search` function takes a sorted array `arr` and a `target` value as input. It uses a while loop to repeatedly divide the search range and compare the target value. If the target is found or the search range becomes empty, it returns the index of the target; otherwise, it returns -1 to indicate that the target is not in the array.

Screenshot copied to clipboard and saved
Select here to mark up and share the image

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=c3d44d4a-c0e7-4160-818a-4be81541d132

NXT wAVES™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions

2 hours, 10 mins

- Introduction to Python
- Conditional Statements and Loops
- List and Functions
- Tuples, Sets and Dictionaries
- OOPs
- Miscellaneous Topics
- More Interview Questions

25 mins 15 mins 25 mins 25 mins 20 mins 20 mins 20 mins

Q. What is the binary search algorithm and how does it work?

Companies Asked: Fretron

Binary search is an efficient algorithm used to find a specific value in a sorted list or array. It works by repeatedly dividing the search range in half and comparing the middle element to the target value. If they match, the search is successful. If not, it narrows down the search to the appropriate half. This process continues until the target is found or the search range is empty, making it a highly effective way to locate items in sorted data sets.

Code:

```
16
17 # Example usage:
18 my_list = [1, 3, 5, 7, 9, 11, 13, 15]
19 target_value = 7
20 result = binary_search(my_list, target_value)
21
22 if result != -1:
23     print(f"Target {target_value} found at index {result}")
24 else:
25     print(f"Target {target_value} not found in the list.")
```

In this code, the `binary_search` function takes a sorted array `arr` and a `target` value as input. It uses a while loop to repeatedly divide the search range and compare the target is found or the search range becomes empty. If the target is found, it returns the index of the target; otherwise, it returns -1 to indicate that the target is not in the list.

Screenshot copied to clipboard and saved
Select here to mark up and share the image

Revolutionizing the Job Market - X

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=c3d44d4a-c0e7-4160-818a-4be81541d132 Download Search Star Print R More

**NXT
wAVES™** Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. Explain the date and time modules in Python.

Companies Asked: Uniserved

The `date` and `time` modules in Python provide functionalities to work with dates, times, and intervals.

- `date` module: Helps in working with dates (year, month, day).

Example:

```
1 from datetime import date
2 today = date.today()
3 print(today) # Outputs current date, e.g., 2023-09-26
```

`PYTHON`

- `time` module: Deals with time and provides functionalities to work with time intervals.

Example:

```
1 import time
2 current_time = time.localtime()
3 formatted_time = time.strftime("%H:%M:%S", current_time)
4 print(formatted_time) # Outputs current time, e.g., 14:05:30
```

`PYTHON`

Both modules are useful for various date and time operations in Python.

Continue

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=c3d44d4a-c0e7-4160-818a-4be81541d132

NXT
wAVES™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. Explain Data Encapsulation in Python?

The process of binding up variables and methods into a single unit is known as Encapsulation. Moreover, it acts as a protective shield that puts restrictions on accessing variables and methods directly and can prevent accidental or unauthorized modification of data.

▼ More information

We use three different types of access modifiers to restrict access to the variables and methods of the class.

1. Public modifier
2. Private modifier
3. Protected modifier

Public modifier:

By default, the member variables and methods are public which means they can be accessed from anywhere outside or inside the class.

Example:

```
1: class Student:
2:     def __init__(self, name, age):
3:         self.name = name
4:         self.age = age
5:
6:     def display(self):
7:         print("Name:", self.name)
8:         print("Age:", self.age)
9:
10: s = Student("John", 20)
```

Private modifier:

Private access modifier can only be accessed within the class where they are defined and cannot be accessed outside the class.

In Python private properties and methods are declared by adding a prefix with two underscores (__) before their declaration.

Example:

PYTHON Continue

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=c3d44d4a-c0e7-4160-818a-4be81541d132

NXT
wAVES™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions (2 hours, 10 mins)

- Introduction to Python (25 mins)
- Conditional Statements and Loops (15 mins)
- List and Functions (25 mins)
- Tuples, Sets and Dictionaries (25 mins)
- OOPs (20 mins)
- Miscellaneous Topics (20 mins)
- More Interview Questions (20 mins)

By default, the member variables and methods are public which means they can be accessed from anywhere outside or inside the class.

Example:

```
2- def __init__(self, name, age):
3-     self.name = name
4-     self.age = age
5-
6- def display(self):
7-     print("Name:", self.name)
8-     print("Age:", self.age)
9-
10 s = Student("John", 20)
11 s.display()
```

Private modifier:

Private access modifier can only be accessed within the class where they are defined and cannot be accessed outside the class.

In Python private properties and methods are declared by adding a prefix with two underscores (`_`) before their declaration.

Example:

```
1- class MyClass:
2-     def __init__(self):
3-         self.__private_var = 42
4-
5-     def __private_method(self):
6-         return "This is a private method."
7-
8-     def public_method(self):
9-         return f"Accessing private variable: {self.__private_var}"
10
```

Protected modifier:

Protected access modifier can be accessed within the class and from the class that inherits the protected class.

In python, protected members and methods are declared using a single underscore (`_`) as prefix before their names.

Continue

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=c3d44d4a-c0e7-4160-818a-4be81541d132

NXT wAVES™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions

2 hours, 10 mins

- Introduction to Python
- Conditional Statements and Loops
- List and Functions
- Tuples, Sets and Dictionaries
- OOPs
- Miscellaneous Topics
- More Interview Questions

Private modifier:

Private access modifier can only be accessed within the class where they are defined and cannot be accessed outside the class.

In Python private properties and methods are declared by adding a prefix with two underscores (__) before their declaration.

Example:

```
1- class MyClass:
2-     def __init__(self):
3-         self.__private_var = 42
4-
5-     def __private_method(self):
6-         return "This is a private method."
7-
8-     def public_method(self):
9-         return f"Accessing private variable: {self.__private_var}"
10-
```

Protected modifier:

Protected access modifier can be accessed within the class and from the class that inherits the protected class.

In python, protected members and methods are declared using a single underscore (_) as prefix before their names.

Example:

```
1- class MyClass:
2-     def __init__(self):
3-         self._protected_var = 10
4-
5-     def __protected_method(self):
6-         print("This is a protected method")
7-
8- class MySubclass(MyClass):
9-     def access_protected(self):
10-         print(self._protected_var)
```

Sure, let's generate questions based on the provided format and instructions.

Continue

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=c3d44d4a-c0e7-4160-818a-4be81541d132

NXT wAVES™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Private access modifier can only be accessed within the class where they are defined and cannot be accessed outside the class.

In Python private properties and methods are declared by adding a prefix with two underscores (__) before their declaration.

Example:

```
3     self.__private_var = 42
4
5     def __private_method(self):
6         return "This is a private method."
7
8     def public_method(self):
9         return f"Accessing private variable: {self.__private_var}"
10
11 obj = MyClass()
12 print(obj.public_method())
```

PYTHON

Protected modifier:

Protected access modifier can be accessed within the class and from the class that inherits the protected class.

In python, protected members and methods are declared using a single underscore (_) as prefix before their names.

Example:

```
1- class MyClass:
2-     def __init__(self):
3-         self._protected_var = 10
4-
5-     def __protected_method(self):
6-         print("This is a protected method")
7-
8- class MySubclass(MyClass):
9-     def access_protected(self):
10-         print(self._protected_var)
```

Sure, let's generate questions based on the provided format and instructions.

Screenshot copied to clipboard and saved Select here to mark up and share the image

Revolutionizing the Job Market - X

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=c3d44d4a-c0e7-4160-818a-4be81541d132 Download Search Star Code Playground Jangipally Ranjith

**NXT
wAVe™**

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Protected modifier:

Protected access modifier can be accessed within the class and from the class that inherits the protected class.

In python, protected members and methods are declared using a single underscore (_) as prefix before their names.

Example:

```
1 - class MyClass:
2 -     def __init__(self):
3 -         self._protected_var = 10
4 -
5 -     def __protected_method(self):
6 -         print("This is a protected method")
7 -
8 - class MySubclass(MyClass):
9 -     def access_protected(self):
10 -         print(self._protected_var)
```

PYTHON

Sure, let's generate questions based on the provided format and instructions.

Practical Examples:

Bank Account Management: Imagine a `BankAccount` class where you encapsulate the balance attribute. You don't want users to directly change the balance. Instead, they should use deposit and withdraw methods. This ensures that the balance is always updated in a controlled manner.

```
1 - class BankAccount:
2 -     def __init__(self, initial_balance):
3 -         self.__balance = initial_balance
4 -
5 -     def deposit(self, amount):
6 -         if amount > 0:
7 -             self.__balance += amount
8 -             return True
9 -         return False
10 -
```

PYTHON

Continue

Revolutionizing the Job Market - X

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=c3d44d4a-c0e7-4160-818a-4be81541d132 Download Search Star Code Playground Jangipally Ranjith

**NXT
wAVE™**

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Protected modifier:

Protected access modifier can be accessed within the class and from the class that inherits the protected class.

In python, protected members and methods are declared using a single underscore (`_`) as prefix before their names.

Example:

```
5 - def __protected_method(self):
6 -     print("This is a protected method")
7 -
8 - class MySubclass(MyClass):
9 -     def access_protected(self):
10 -         print(self._protected_var)
11 -         self.__protected_method()
12 -
13 - obj = MySubclass()
14 - obj.access_protected()
```

Sure, let's generate questions based on the provided format and instructions.

Practical Examples:

Bank Account Management: Imagine a `BankAccount` class where you encapsulate the balance attribute. You don't want users to directly change the balance. Instead, they should use deposit and withdraw methods. This ensures that the balance is always updated in a controlled manner.

```
1 - class BankAccount:
2 -     def __init__(self, initial_balance):
3 -         self.__balance = initial_balance
4 -
5 -     def deposit(self, amount):
6 -         if amount > 0:
7 -             self.__balance += amount
8 -             return True
9 -         return False
10 -
```

PYTHON

Continue

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=c3d44d4a-c0e7-4160-818a-4be81541d132

NXT wAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions (2 hours, 10 mins)

- Introduction to Python (Reading Material, 25 mins)
- Conditional Statements and Loops (Reading Material, 15 mins)
- List and Functions (Reading Material, 25 mins)
- Tuples, Sets and Dictionaries (Reading Material, 25 mins)
- OOPs (Reading Material, 20 mins)
- Miscellaneous Topics (Reading Material, 20 mins)
- More Interview Questions (20 mins)

9 - def access_protected(self):
10 - print(self._protected_var)
11 - self._protected_method()
12 -
13 - obj = MySubclass()
14 - obj.access_protected()

Sure, let's generate questions based on the provided format and instructions.

Practical Examples:

Bank Account Management: Imagine a `BankAccount` class where you encapsulate the balance attribute. You don't want users to directly change the balance. Instead, they should use deposit and withdraw methods. This ensures that the balance is always updated in a controlled manner.

PYTHON

```
1 - class BankAccount:  
2 -     def __init__(self, initial_balance):  
3 -         self.__balance = initial_balance  
4 -  
5 -     def deposit(self, amount):  
6 -         if amount > 0:  
7 -             self.__balance += amount  
8 -             return True  
9 -         return False  
10 -
```

Continue

Ask doubt

Notes Discussions

Notes + NEW NOTE

B I U S E ≡ %

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=c3d44d4a-c0e7-4160-818a-4be81541d132

NXT WAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

9. def access_protected(self):
10. print(self._protected_var)
11. self._protected_method()
12.
13. obj = MySubclass()
14. obj.access_protected()

Sure, let's generate questions based on the provided format and instructions.

Practical Examples:

Bank Account Management: Imagine a `BankAccount` class where you encapsulate the balance attribute. You don't want users to directly change the balance. Instead, they should use deposit and withdraw methods. This ensures that the balance is always updated in a controlled manner.

PYTHON

```
    1.     def deposit(amount):  
2.         self._balance += amount  
3.         return True  
4.  
5.     def withdraw(amount):  
6.         if 0 < amount <= self._balance:  
7.             self._balance -= amount  
8.             return True  
9.         return False  
10.  
11.    def get_balance(self):  
12.        return self._balance
```

Continue

Ask doubt

Notes Discussions

Notes + NEW NOTE

31°C Partly sunny

Search

13:23 14-02-2024

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=c3d44d4a-c0e7-4160-818a-4be81541d132

NXT WAVE™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Sure, let's generate questions based on the provided format and instructions.

Practical Examples:

Bank Account Management: Imagine a `BankAccount` class where you encapsulate the balance attribute. You don't want users to directly change the balance. Instead, they should use deposit and withdraw methods. This ensures that the balance is always updated in a controlled manner.

```
12-     if 0 < amount <= self.__balance:  
13         self.__balance -= amount  
14         return True  
15     return False  
16  
17-     def get_balance(self):  
18         return self.__balance  
19  
20 my_balance = BankAccount("1000")  
21 print(my_balance.balance)
```

PYTHON

Ask doubt

Notes Discussions

Title Take a Note

31°C Partly sunny

Search

Snipping Tool

Screenshot copied to clipboard and saved Select here to mark up and share the image

13:23 14-02-2024 ENG IN

Revolutionizing the Job Market - X

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=c3d44d4a-c0e7-4160-818a-4be81541d132 Download Search Star Print R More

**NXT
wAVe™** Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python** Reading Material 25 mins
- Conditional Statements and Loops** Reading Material 15 mins
- List and Functions** Reading Material 25 mins
- Tuples, Sets and Dictionaries** Reading Material 25 mins
- OOPs** Reading Material 20 mins
- Miscellaneous Topics** Reading Material 20 mins
- More Interview Questions** 20 mins

Q. What does the single underscore mean?

Intent: Interviewer wants to know about protected access modifiers.

A single underscore is used to indicate that a variable is internal and should not be accessed directly from outside the class. The members of a class that are declared internal are only accessible to the class itself and its subclasses.

Some use cases of protected variable:

1. Enforcing Data Consistency: Protecting data members in a base class allows subclasses to interact with them in a controlled manner, maintaining data consistency and avoiding unintentional misuse.
2. Resource Management: To manage resource state in subclasses, protected variables can store relevant information.

More Information

Example

Here is a small sample of a protected variable:

```
1 - class dog:
2 -     def __init__(self):
3 -         self._dog_name = "Max"
4 -
5 - t1 = dog()
6 - print(t1._dog_name)
```

PYTHON

In the example above, the `_dog_name` variable is protected because it is prefixed with a single underscore. This means that it can only be accessed from within the `dog` class or its subclasses.

Continue CHAT

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=c3d44d4a-c0e7-4160-818a-4be81541d132

NXT
wAVES™

CONTENTS

Interview Questions (2 hours, 10 mins)

- Introduction to Python (Reading Material, 25 mins)
- Conditional Statements and Loops (Reading Material, 15 mins)
- List and Functions (Reading Material, 25 mins)
- Tuples, Sets and Dictionaries (Reading Material, 25 mins)
- OOPs (Reading Material, 20 mins)
- Miscellaneous Topics (Reading Material, 20 mins)
- More Interview Questions (20 mins)

Protected variable (Static variable)

Q. What is the difference between a .py file and a .pyc file?

A .py file is a Python source file with human-readable code. When this file is imported, the Python interpreter generates a .pyc file, which is a compiled version containing Python bytecode. This bytecode is faster to execute, improving performance. However, .pyc files are only created for imported modules, not for the main script that's directly executed.

Question 1 of 4

What does a .pyc file contain?

- A Python file that is currently being edited.
- Machine-readable Python bytecode. (Correct)
- A backup copy of a Python source file.
- A special format for Python configuration files.

Next

Complete the MCQs to proceed

Continue

Code Playground Jangipally Ranjith

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=c3d44d4a-c0e7-4160-818a-4be81541d132

NXT wAVES™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions (2 hours, 10 mins)

- Introduction to Python (Reading Material, 25 mins)
- Conditional Statements and Loops (Reading Material, 15 mins)
- List and Functions (Reading Material, 25 mins)
- Tuples, Sets and Dictionaries (Reading Material, 25 mins)
- OOPs (Reading Material, 20 mins)
- Miscellaneous Topics (Reading Material, 20 mins)
- More Interview Questions (20 mins)

Q. What is method overriding in Python?

Method overriding in object-oriented programming enables a subclass to replace or extend the behavior of a method inherited from its parent class with its own implementation. This helps us to customize the functionality of inherited methods in a more specific manner to suit the requirements of the subclass.

```
1. class Animal:
2.     def speak(self):
3.         print("Animal sound")
4.
5. class Dog(Animal):
6.     def speak(self):
7.         print("Woof!")
8.
9. class Cat(Animal):
10.    def speak(self):
```

PYTHON

▼ More Information

Benefits of method overriding:

- Increased flexibility:** Method overriding allows a subclass to reuse methods from its superclass.
- Reduced code duplication:** Method overriding reduces code duplication by letting subclasses reuse superclass methods, making code more maintainable and understandable.
- Improved polymorphism:** Method overriding, a key feature of polymorphism, allows objects of different types to respond to the same method differently, making code more powerful and expressive.

Continue

Revolutionizing the Job Market - X

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=c3d44d4a-c0e7-4160-818a-4be81541d132 Download Search Star Print R More

**NXT
wAVES™** Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. What is method overriding in Python?

Method overriding in object-oriented programming enables a subclass to replace or extend the behavior of a method inherited from its parent class with its own implementation. This helps us to customize the functionality of inherited methods in a more specific manner to suit the requirements of the subclass.

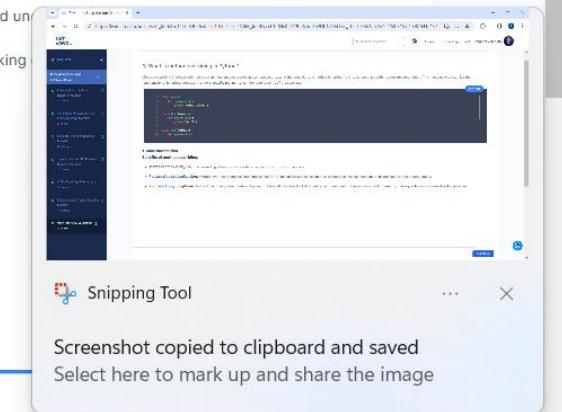
PYTHON

```
8
9 - class Cat(Animal):
10 -     def speak(self):
11         print("Meow!")
12
13 dog = Dog()
14 dog.speak() # Output: Woof!
15
16 cat = Cat()
17 cat.speak() # Output: Meow!
```

▼ More Information

Benefits of method overriding:

- Increased flexibility:** Method overriding allows a subclass to reuse methods from its superclass
- Reduced code duplication:** Method overriding reduces code duplication by letting subclasses reuse superclass methods, making code more maintainable and understandable
- Improved polymorphism:** Method overriding, a key feature of polymorphism, allows objects of different types to respond to the same method differently, making the code more flexible and extensible

 A screenshot of a Windows desktop showing a browser window with a course page. The page title is "What is method overriding in Python?". It contains text about method overriding and a code snippet. Below the code, there's a section titled "More Information" with a bulleted list of benefits. A "Snipping Tool" window is overlaid on the bottom right, showing a message: "Screenshot copied to clipboard and saved Select here to mark up and share the image".

Revolutionizing the Job Market - X

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=c3d44d4a-c0e7-4160-818a-4be81541d132 Download Search Star Print R More

**NXT
wAVe™** Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. What is method overriding in Python?

Method overriding in object-oriented programming enables a subclass to replace or extend the behavior of a method inherited from its parent class with its own implementation. This helps us to customize the functionality of inherited methods in a more specific manner to suit the requirements of the subclass.

PYTHON

```
8
9 - class Cat(Animal):
10 -     def speak(self):
11         print("Meow!")
12
13 dog = Dog()
14 dog.speak() # Output: Woof!
15
16 cat = Cat()
17 cat.speak() # Output: Meow!
```

▼ More Information

Benefits of method overriding:

- **Increased flexibility:** Method overriding allows a subclass to reuse methods from its superclass
- **Reduced code duplication:** Method overriding reduces code duplication by letting subclasses reuse superclass methods, making code more maintainable and understandable.
- **Improved polymorphism:** Method overriding, a key feature of polymorphism, allows objects of different types to respond to the same method differently, making code more powerful and expressive.

Continue

Revolutionizing the Job Market - X

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=c3d44d4a-c0e7-4160-818a-4be81541d132 Download Search Star Print R More

**NXT
wAVE™** Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. What is a linked list?

A linked list is a linear data structure where each element is a separate object, known as a node. Each node consists of two items: the data and a reference to the next node. The last node has a reference to null, indicating the end of the chain. The entry point into a linked list is called the head of the list.

More Information

Creating a Node class:

```
1- class Node:
2-     def __init__(self, data):
3-         self.data = data
4-         self.next = None
```

Creating LinkedList class:

```
1- class LinkedList:
2-     def __init__(self):
3-         self.head = None
4-
5-     def add(self, data):
6-         new_node = Node(data)
7-         new_node.next = self.head
8-         self.head = new_node
9-
10-    def print_list(self):
```

There are three types of linked lists: [Singly linked lists](#), [Doubly linked lists](#), and [Circular linked lists](#). In a singly linked list, each node points to the next node in the list. In a doubly linked list, each node points to both the next node and the previous node. Circular linked lists are a variation of a linked list in which the last node points back to the first node, forming a loop.

Continue

Revolutionizing the Job Market - X

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=c3d44d4a-c0e7-4160-818a-4be81541d132 Download Search Star Print R More

**NXT
wAVe™** Home Code Playground Jangipally Ranjith

CONTENTS

Interview Questions 2 hours, 10 mins

- Introduction to Python Reading Material 25 mins
- Conditional Statements and Loops Reading Material 15 mins
- List and Functions Reading Material 25 mins
- Tuples, Sets and Dictionaries Reading Material 25 mins
- OOPs Reading Material 20 mins
- Miscellaneous Topics Reading Material 20 mins
- More Interview Questions 20 mins

Q. What is a linked list?

A linked list is a linear data structure where each element is a separate object, known as a node. Each node consists of two items: the data and a reference to the next node. The last node has a reference to null, indicating the end of the chain. The entry point into a linked list is called the head of the list.

▼ More Information

Creating a Node class:

```
1 - class Node:
2 -     def __init__(self, data):
3 -         self.data = data
4 -         self.next = None
```

Creating LinkedList class:

```
9
10 - def print_list(self):
11     current_node = self.head
12 -     while current_node is not None:
13         print(current_node.data)
14         current_node = current_node.next
15
16 my_linked_list = LinkedList()
17 my_linked_list.add(1)
18 my_linked_list.add(2)
```

There are three types of linked lists: [Singly linked lists](#), [Doubly linked lists](#), and [Circular linked lists](#). In a singly linked list, each node points to the next node in the list. In a doubly linked list, each node points to both the next node and the previous node. Circular linked lists are a variation of a linked list in which the last node points back to the first node, forming a loop.

Continue

Revolutionizing the Job Market

https://learning.ccbp.in/course?c_id=b8cd14fe-f80f-46a3-95c1-98e4ce7e465f&t_id=41accf70-84e0-4279-a6ab-02e93b1760d1&s_id=c3d44d4a-c0e7-4160-818a-4be81541d132

NXT
wAVe™

Search for 'Arrays'

Home Code Playground Jangipally Ranjith

A linked list is a linear data structure where each element is a separate object, known as a node. Each node consists of two items: the data and a reference to the next node. The last node has a reference to null, indicating the end of the chain. The entry point into a linked list is called the head of the list.

▼ More Information

Creating a Node class:

```
1 - class Node:
2 -     def __init__(self, data):
3 -         self.data = data
4 -         self.next = None
```

Creating LinkedList class:

```
11      current_node = self.head
12      while current_node is not None:
13          print(current_node.data)
14          current_node = current_node.next
15
16 my_linked_list = LinkedList()
17 my_linked_list.add(1)
18 my_linked_list.add(2)
19
20 my_linked_list.print_list()
```

There are three types of linked lists: [Singly linked lists](#), [Doubly linked lists](#), and [Circular linked lists](#). In a singly linked list, each node points to the next node in the list. In a doubly linked list, each node points to both the next node and the previous node. Circular linked lists are a variation of a linked list in which the last node points back to the first node, forming a loop.

Ask doubt

Continue

31°C Partly sunny

Search

13:26 14-02-2024