knowledgehut
upGrad

Introduction to Python for Data Science

Pre-requisites

Hope you have gone through the self-learning content for this session on the PRISM portal.

What have we learned so far?

- Introduction to SQL, MySQL, and MySQL Workbench.
- Types of SQL commands – DDL, DML, DQL, and DCL.
- Fundamentals of SELECT query.
- Use of WHERE, GROUP BY, ORDER BY clauses
- Built-in functions for text, numerical and date columns.
- Use of temporary tables and views.
- Improving efficiency of SQL queries using INDEX.

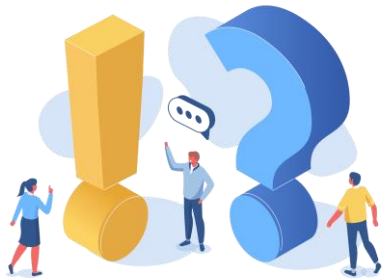
By the End of this Session, You Will be Able to:

- Install Python and essential libraries for data science.
- Understand how to work with code and data in Python.
- Learn the lexical structure of Python.
- Categorize the data into it’s respective datatype.
- Practice Python through hands-on examples and exercises.

Pop Quiz

Q. What is the first step to setting up Python on your computer?

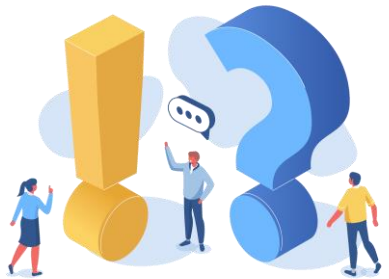
a. Installing Python
b. Choosing an Integrated Development Environment (IDE)
c. Downloading Python packages
d. Setting up a virtual environment



Pop Quiz

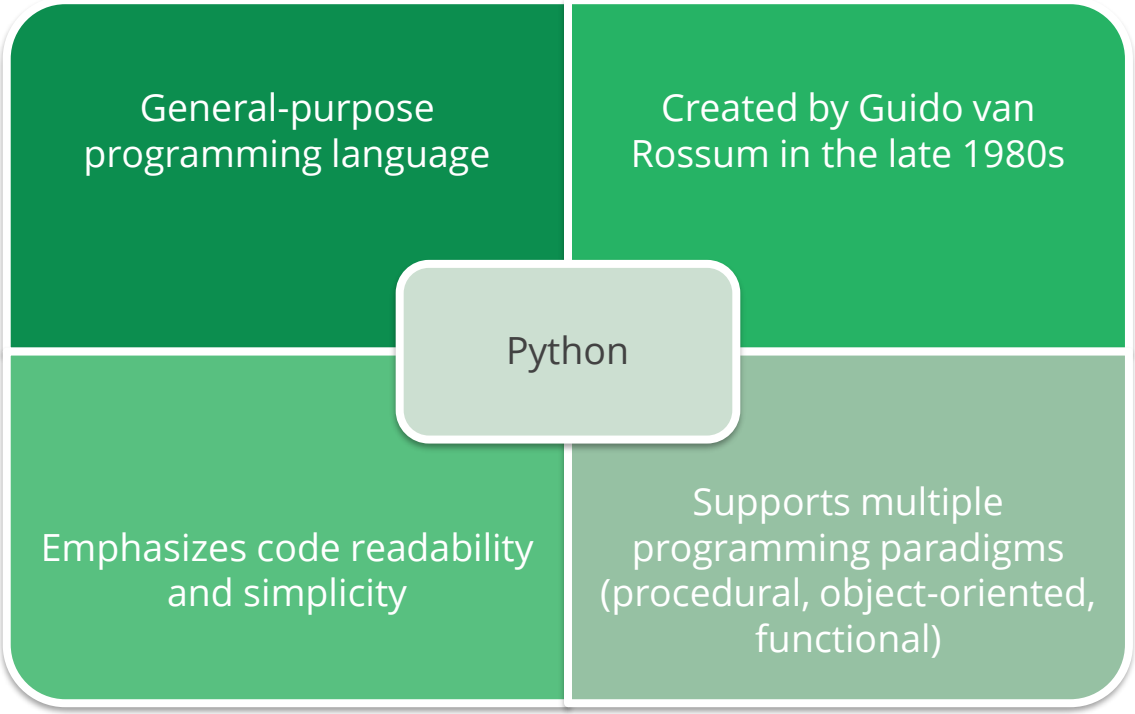
Q. What is the first step to setting up Python on your computer?

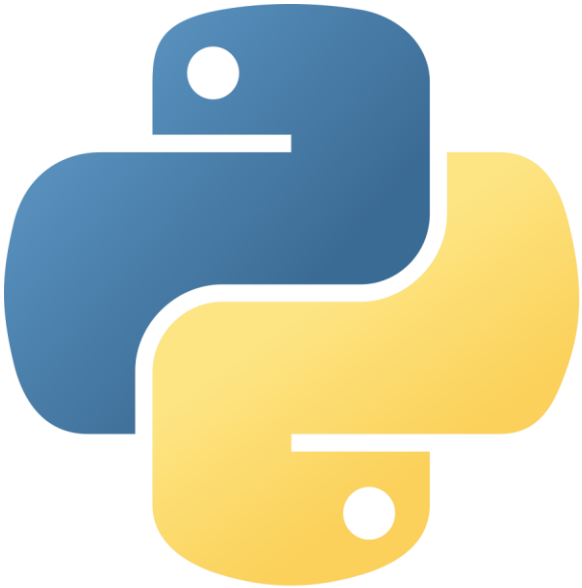
a. **Installing Python**
b. Choosing an Integrated Development Environment (IDE)
c. Downloading Python packages
d. Setting up a virtual environment



Introduction to Python

Python: An Overview

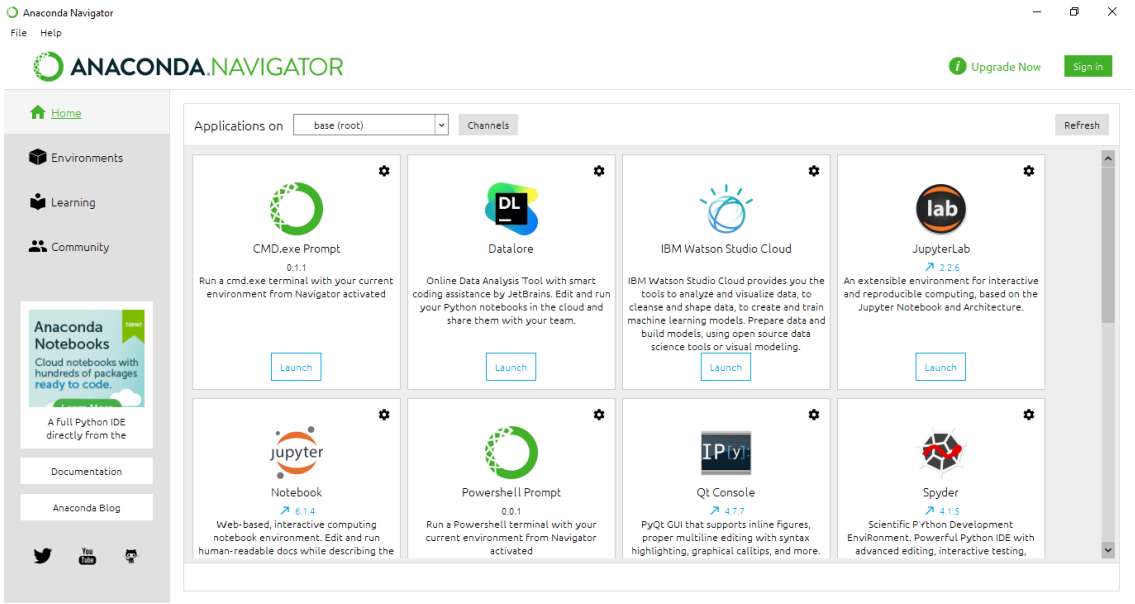




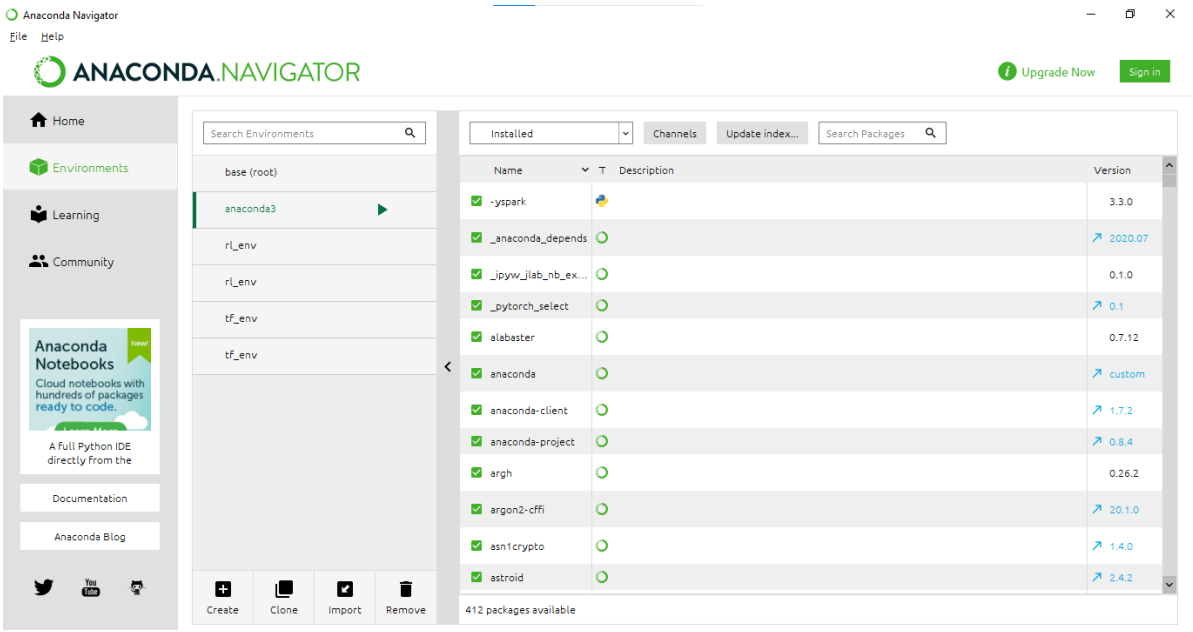
Python Installation

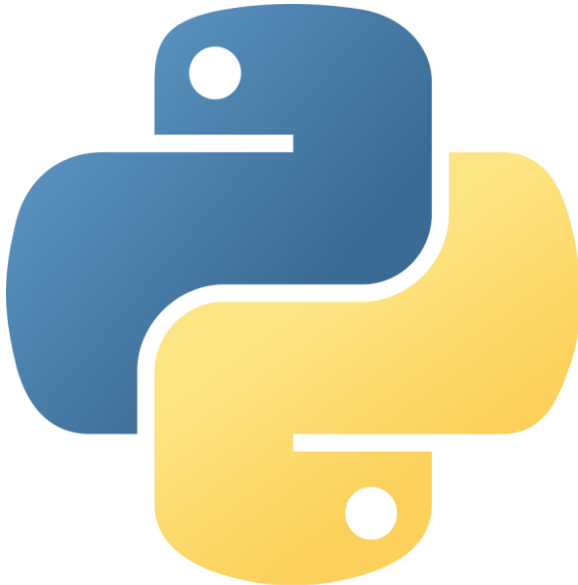
Installation using Anaconda

Anaconda is a popular open-source distribution of the Python programming language. It is designed to simplify package management and deployment of scientific computing libraries and tools.



Installation using Anaconda





Demo – Starting Jupyter Notebook

Poll Time

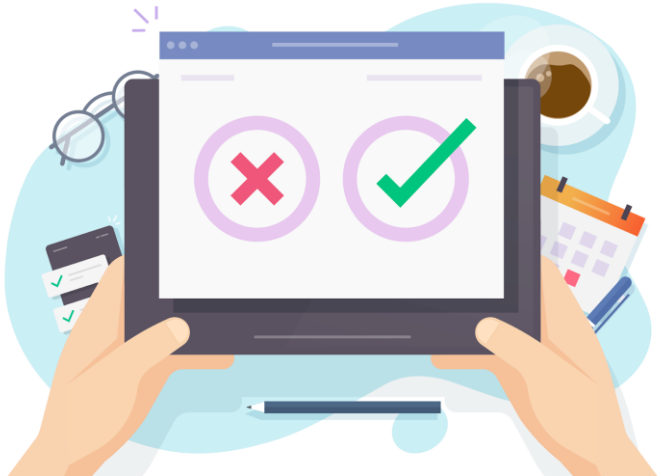
Q. Which operating systems can you install Python on?

a. Windows only
b. macOS only
c. Linux only
d. Windows, macOS, and Linux

Poll Time

Q. Which operating systems can you install Python on?

a. Windows only
b. macOS only
c. Linux only
d. Windows, macOS, and Linux



Code and Data

Introduction to Code and Data

The term "code" typically refers to a set of instructions written in a programming language that is understandable by both humans and computers. It is a way of communicating with a computer to perform specific tasks or operations.

python Copy code

```python
import math

# Get the radius from the user
radius = float(input("Enter the radius of the circle: "))

# Calculate the area of the circle
area = math.pi * (radius ** 2)

# Display the result
print("The area of the circle is:", area)
```

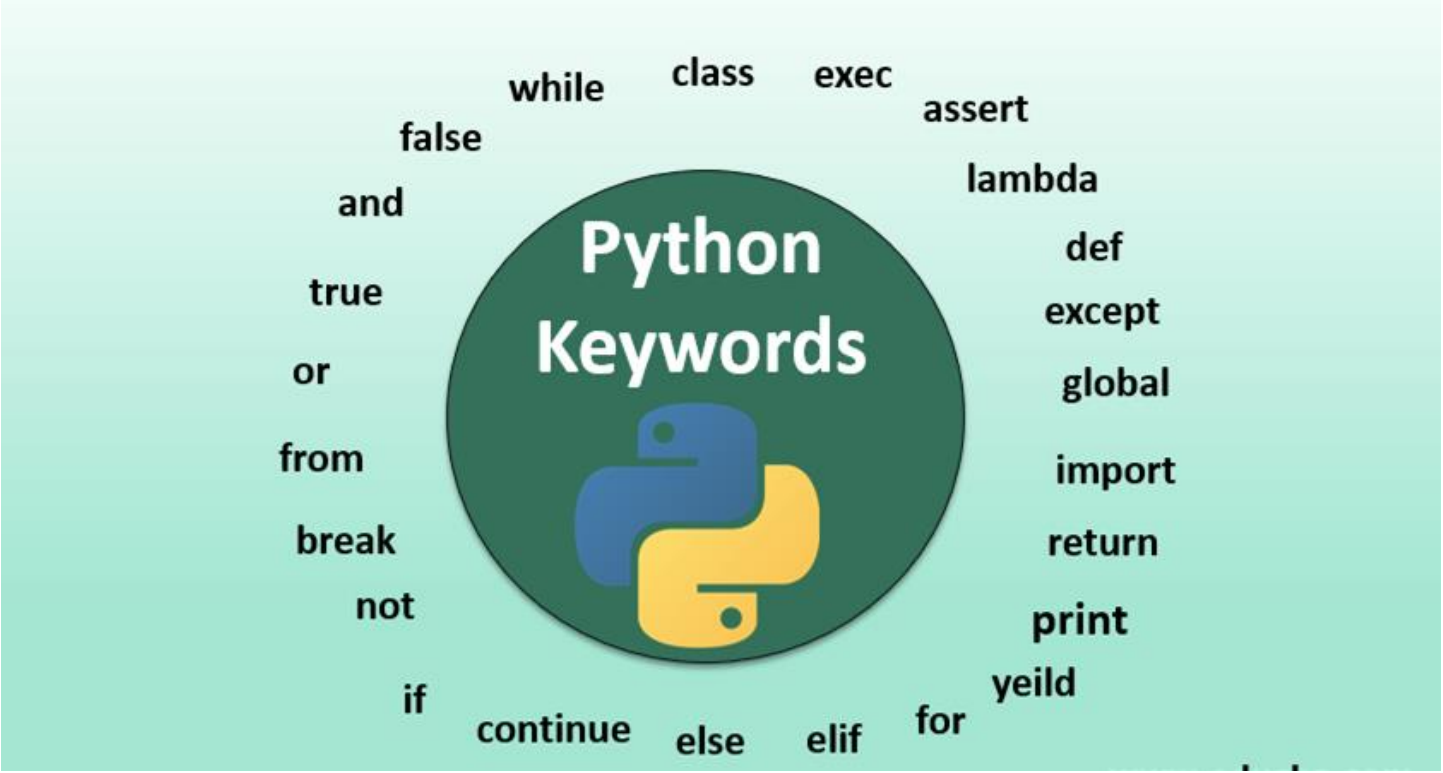
Lexical Structure of Python

Lexical Structure of Python

The lexical structure of Python refers to the set of rules that govern how the Python programming language is written and structured at the basic level.


Q&A

Keywords in Python



Pop Quiz

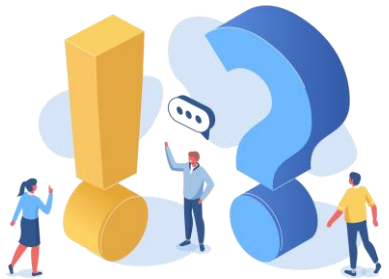
Q. Which of the following options is true regarding the lexical structure of Python?

a. Whitespace is significant and affects program execution.

b. Identifiers can start with a digit.

c. Comments are executed by the Python interpreter.

d. Keywords can be used as variable names.



Pop Quiz

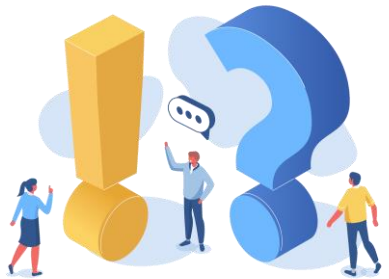
Q. Which of the following options is true regarding the lexical structure of Python?

a. Whitespace is always significant and affects program execution.

b. Identifiers can start with a digit.

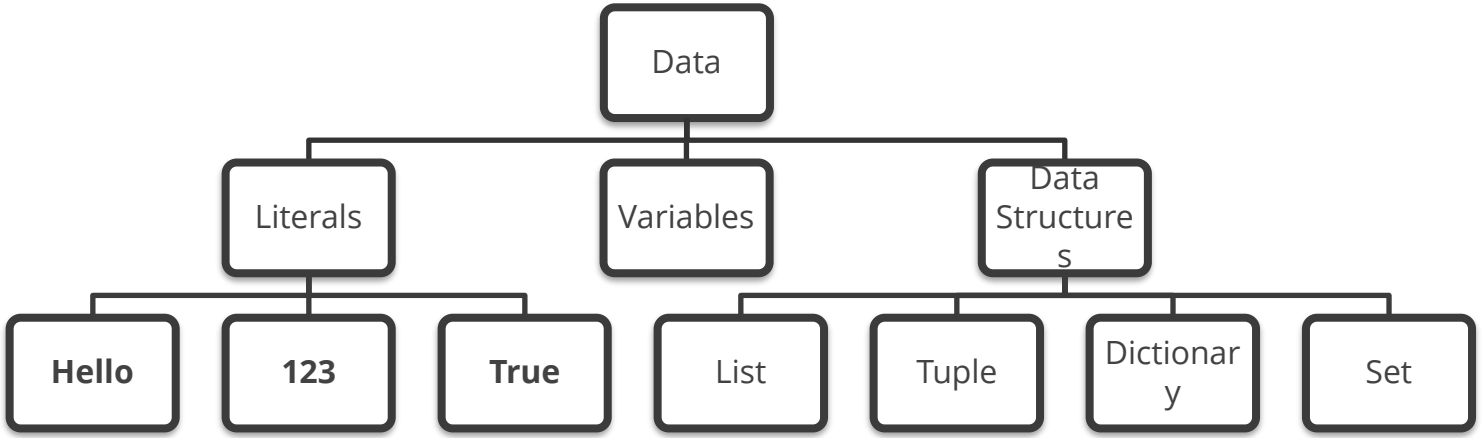
c. Comments are not executed by the Python interpreter.

d. Keywords can be used as variable names.



Creating Data

In programming languages, data refers to the values or information that a program manipulates, stores, and processes.



Introduction to Variables in Python

Variables are containers to store data values. In python, we need to follow the below mentioned naming conventions when creating variables:

- Can contain letters (a-z, A-Z), digits (0-9), and underscores (_)
- Must start with a letter or underscore (cannot start with a digit)
- Are case-sensitive (e.g., "count" and "Count" are different variables)
- Should use descriptive names for clarity

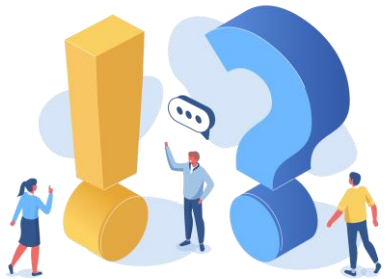
python Copy code

```python
# Integer Variable
age = 25

# String Variable
name = "John Doe"

# Floating-Point Variable
pi = 3.14

# Boolean Variable
is_active = True

# Reassigning a Variable
age = age + 1

# Multiple Assignments
x, y, z = 10, 20, 30
```

Pop Quiz

Q. Which of the following options correctly follow the naming conventions for variables in Python?

a. 1_var
b. variableName
c. variable-name
d. var 1



Pop Quiz

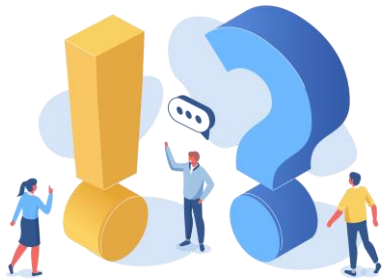
Q. Which of the following options correctly follow the naming conventions for variables in Python?

a. 1_var

b. variableName

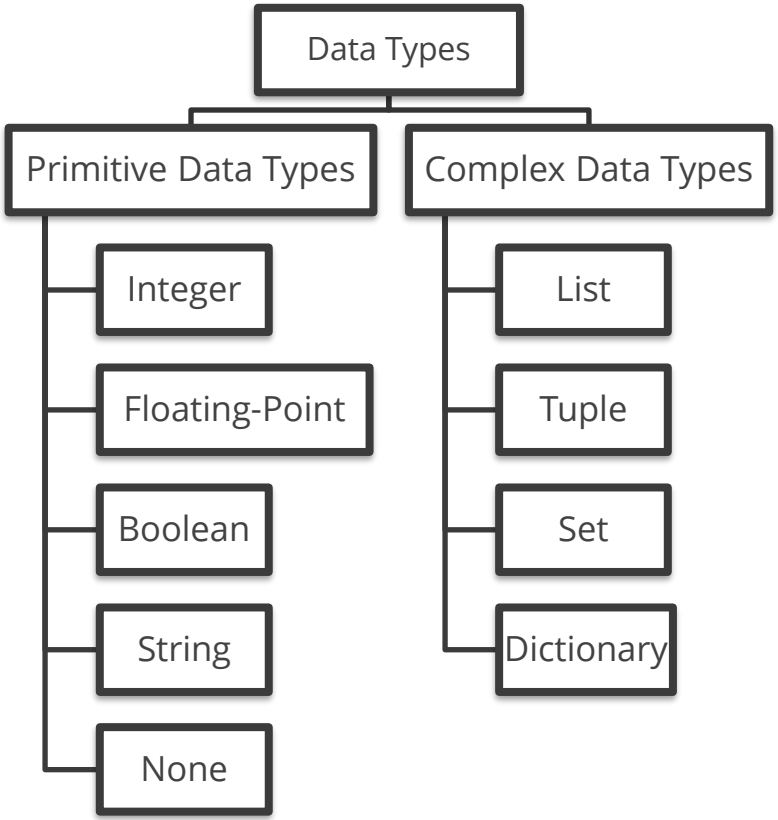
c. variable-name

d. var 1



Demo – Variables and Literals

Python Data Types



Demo – Primitive Data Types

Poll Time

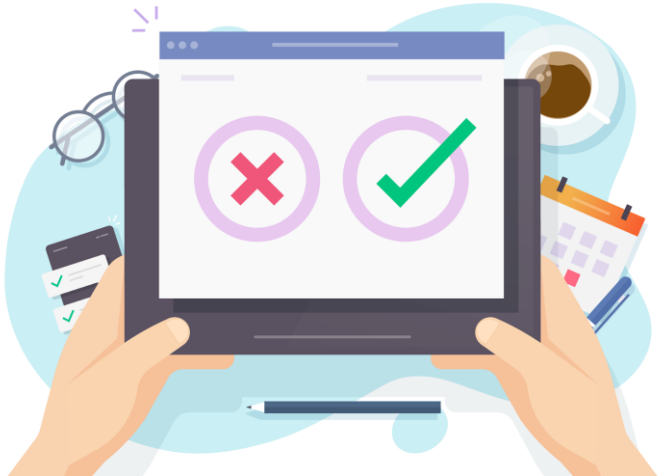
Q. Which of the following options correctly represents the primitive data types in Python?

a. Integer, Float, List, String

b. Boolean, Tuple, Dictionary, Set

c. Integer, Float, Boolean, String

d. List, Tuple, Dictionary, Set



Poll Time

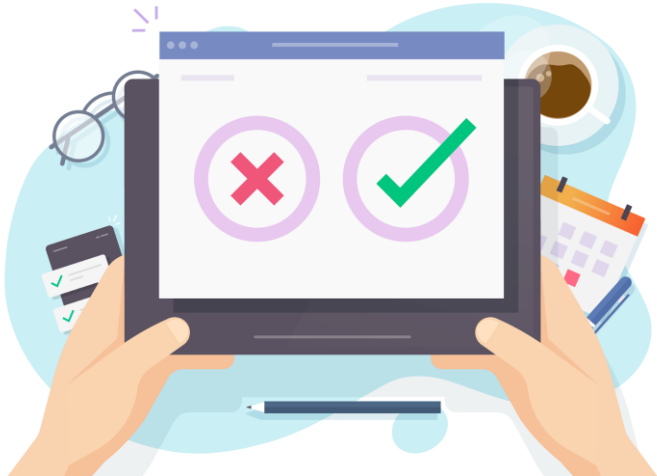
Q. Which of the following options correctly represents the primitive data types in Python?

a. Integer, Float, List, String

b. Boolean, Tuple, Dictionary, Set

c. Integer, Float, Boolean, String

d. List, Tuple, Dictionary, Set




Q&A

Summary

- Python is a high level programming language.
- Code refers to the set of instructions which are executed by the computer.
- Data can be present in different formats in python – literals, variables and data structures.
- Data types in python are categorized as Simple and Complex.

Activity 1

Pre-requisites:

- Python 3.x – preferably python 3.8
- Jupyter Notebook

Scenario:

As you are getting started with python, as a first activity start jupyter notebook in your working directory and perform the below tasks:

- Navigate to jupyter notebook homepage and create a new notebook.
- Rename the notebook to “Python_Practice.ipynb”.
- Create variable to store your name, age and gender.
- Use print function as shown below to print the details on the screen.

```
In [5]: print(name, age, gender)

        Huzefa 28 M
```

Next Session:
Python – Building Blocks and Strings

THANK YOU!

Please complete your assessments and review the self-learning content for this session on the **PRISM** portal.

knowledgehut
upGrad

knowledgehut
upGrad

Python – Building Blocks and Strings

Pre-requisites

Hope you have gone through the self-learning content for this session on the PRISM portal.

By the End of this Session, You Will:

- Explore string datatype in python.
- Learn to manipulate python strings using indexing and slicing.
- Understand the different functions which are available for each string object.
- Use data structures – lists, tuples, and sets, to store a collection of items in a single variable.
- Understand the differences between these data structures and the use case for each one.

Recap

Poll Time

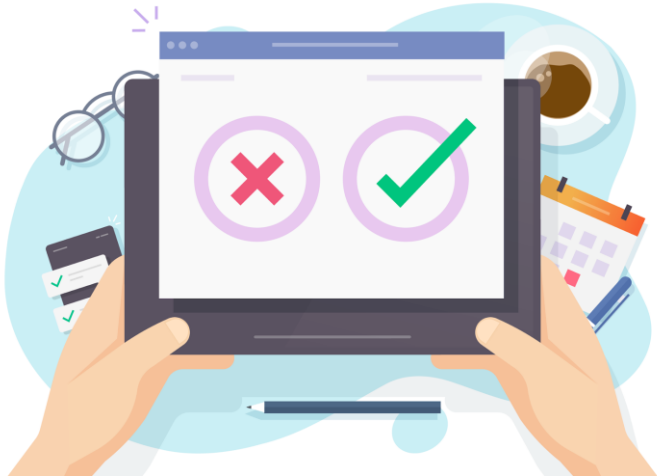
Q. Which of the following options accurately describes the purpose of Python's "print" function?

a. It is used to perform mathematical calculations and operations

b. It is used to accept user input from the keyboard

c. It is used to display output or information on the console

d. It is used to define loops for iterative execution



Poll Time

Q. Which of the following options accurately describes the purpose of Python's "print" function?

a. It is used to perform mathematical calculations and operations

b. It is used to accept user input from the keyboard

c. It is used to display output or information on the console

d. It is used to define loops for iterative execution

Introduction to Strings

Introduction to Strings

Strings are defined as a sequence of characters. They are enclosed in single quotes (‘ ') or double quotes (“ ").

python Copy code

```python
name = 'Alice'
```

python Copy code

```python
message = "Hello, World!"
```

Indexing and Slicing

String indexing and slicing are techniques used to access specific characters or substrings within a string in Python.

Example: String Indexing

```python
message = "Hello, World!"
print(message[0])  # Output: 'H'
print(message[7])  # Output: 'W'
```

Example: String Slicing

```python
message = "Hello, World!"
print(message[7:12])  # Output: 'World'
```

Negative Indexing, Slicing and Stepping

Negative indexing in Python allows you to access elements or characters in a sequence (such as a string or list) starting from the end rather than the beginning.

Example: Negative Indexing

```python
message = "Hello, World!"

print(message[-1])  # Output: '!'
print(message[-2])  # Output: 'd'
print(message[-6])  # Output: 'W'
```

Example: Negative Indexing

```python
message = "Hello, World!"

print(message[-6:-1])  # Output: 'World'
```

The Concept of Immutability

In Python, strings are immutable, which means that once a string is created, its contents cannot be changed.

Example: Immutability

python Copy code

```python
name = "Alice"
print("Original String:", name)

# Attempting to change a character in the string
name[0] = 'B'
print("Modified String:", name)
```

typescript Copy code

```
Original String: Alice
TypeError: 'str' object does not support item assignment
```

Pop Quiz

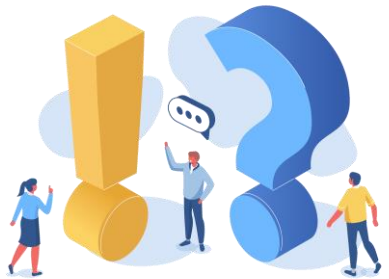
Q. Which of the following statements about strings in Python is true?

a. Strings are mutable and can be modified after creation

b. Strings can be enclosed in single quotes (‘ ’)

c. Strings don’t support negative indexing

d. String indexing starts from 1 in Python



Pop Quiz

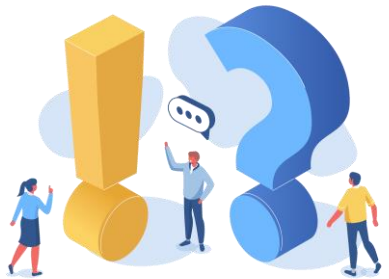
Q. Which of the following statements about strings in Python is true?

a. Strings are mutable and can be modified after creation

b. Strings can be enclosed in single quotes (‘ ’)

c. Strings don’t support negative indexing

d. String indexing starts from 1 in Python



Important String Functions

Important String Functions

len()

- Returns the length (number of characters) of a string.

str()

- Converts an object into a string representation.

lower()

- Converts all characters in a string to lowercase.

strip()

- Removes leading and trailing whitespace characters from a string.

replace()

- Replaces all occurrences of a specified substring with another substring in a string.

split()

- Splits a string into a list of substrings based on a specified delimiter.

join()

- Joins a list of strings into a single string using a specified separator.

Demo – Python Strings

Pop Quiz

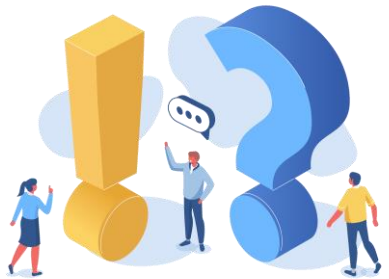
Q. Which of the following Python string functions is used to remove leading and trailing whitespace characters from a string?

a. len()

b. strip()

c. replace()

d. split()



Pop Quiz

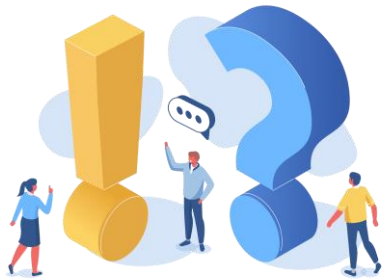
Q. Which of the following Python string functions is used to remove leading and trailing whitespace characters from a string?

a. len()

b. strip()

c. replace()

d. split()

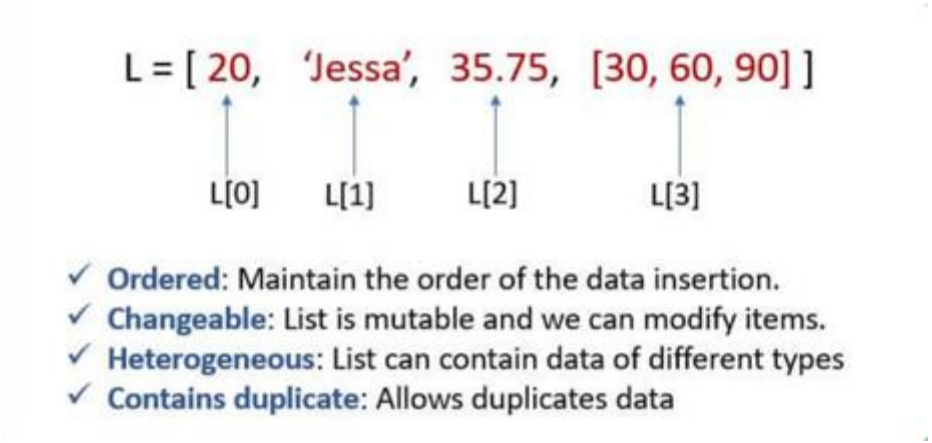



Q&A

Data Structures in Python

Introduction to Lists

In Python, a list is a versatile data structure that can store a collection of elements. It is a mutable, ordered sequence, allowing you to store multiple items in a single variable.



List Functions

append()
- Adds an element to the end of a list.

insert()
- Inserts an element at a specified index position in a list.

remove()
- Removes the first occurrence of a specified element from a list.

pop()
- Removes and returns the element at a specified index position in a list.

sort()
- Sorts the elements in a list in ascending order.

reverse()
- Reverses the order of elements in a list.

count()
- Returns the number of occurrences of a specified element in a list.

Pop Quiz

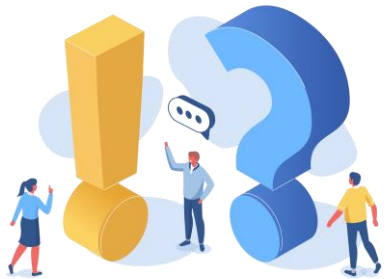
Q. Which of the following options accurately describes the characteristics of Python lists?

a. Lists are immutable and cannot be modified after creation

b. Lists can contain elements of different data types

c. Lists preserve the order of elements but do not allow duplicate values

d. Lists can only store a fixed number of elements



Pop Quiz

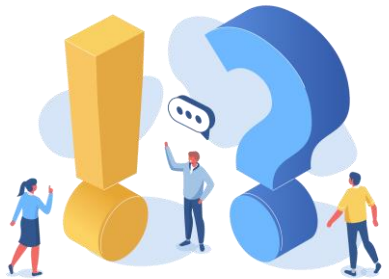
Q. Which of the following options accurately describes the characteristics of Python lists?

a. Lists are immutable and cannot be modified after creation

b. Lists can contain elements of different data types

c. Lists preserve the order of elements but do not allow duplicate values

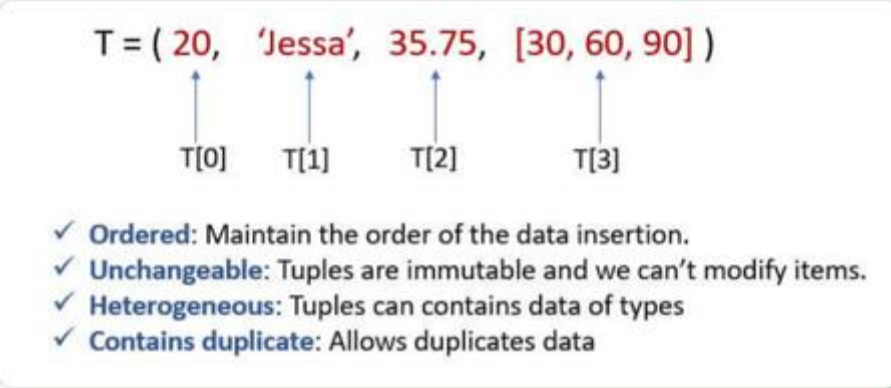
d. Lists can only store a fixed number of elements



Demo – Python Lists

Introduction to Tuples

A tuple is an ordered collection of elements in Python that is similar to a list. However, unlike lists, tuples are immutable, meaning they cannot be modified once created.



Tuple Functions

len()

- Returns the number of elements in a tuple.

sorted()

- Returns a new list containing the sorted elements of a tuple.

count()

- Returns the number of times element appears in the tuple.

index()

- Returns index of the passed element.

Pop Quiz

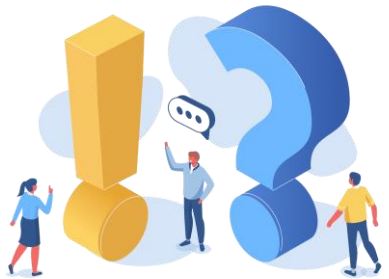
Q. Which of the following options accurately describes the difference between tuples and lists in Python?

a. Tuples are immutable, while lists are mutable

b. Tuples preserve the order of elements, while lists do not

c. Tuples allow duplicate values, while lists do not

d. Tuples are created using square brackets ([]), while lists are created using parentheses ().



Pop Quiz

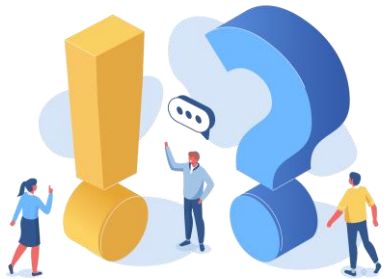
Q. Which of the following options accurately describes the difference between tuples and lists in Python?

a. **Tuples are immutable, while lists are mutable**

b. Tuples preserve the order of elements, while lists do not

c. Tuples allow duplicate values, while lists do not

d. Tuples are created using square brackets ([]), while lists are created using parentheses ()




Q&A

Summary

- Strings are a collection of characters in python.
- String functions allow us to manipulate strings.
- Python data structures allow you to store a collection of values in a single object.
- Lists are an ordered, mutable collection of Python objects.
- Tuples are an ordered, immutable collection of Python objects.

Activity 1

Pre-requisites:

- Python 3.x – preferably python 3.8
- Jupyter Notebook

Scenario:

As you are getting started with Python, as a first activity, start jupyter notebook in your working directory and perform the below tasks:

- Create a variable to store your name and calculate the length of it.
- Also convert your name to UPPER case using string function.
- Create a python list to store your hobbies.
- Manipulate the list by adding hobbies, inserting them at specific indices and dropping few of them.

Session Feedback

Next Session:
Operators in Python

THANK YOU!

Please complete your assessments and review the self-learning content for this session on the **PRISM** portal.

knowledgehut
upGrad