

Tokens in Python

Data Science and A.I. Lecture Series

Bindeshwar Singh Kushwaha

PostNetwork Academy

Introduction to Python Tokens

Tokens are the smallest individual units in a Python program.

Introduction to Python Tokens

Tokens are the smallest individual units in a Python program. Everything in a Python program is built using tokens.

Introduction to Python Tokens

Tokens are the smallest individual units in a Python program. Everything in a Python program is built using tokens. Python has five types of tokens:

Introduction to Python Tokens

Tokens are the smallest individual units in a Python program. Everything in a Python program is built using tokens. Python has five types of tokens:

Introduction to Python Tokens

Tokens are the smallest individual units in a Python program. Everything in a Python program is built using tokens. Python has five types of tokens:

- Keywords are reserved words in Python.

Introduction to Python Tokens

Tokens are the smallest individual units in a Python program. Everything in a Python program is built using tokens. Python has five types of tokens:

- Keywords are reserved words in Python.
- Identifiers are names given to variables, functions, and classes.

Introduction to Python Tokens

Tokens are the smallest individual units in a Python program. Everything in a Python program is built using tokens. Python has five types of tokens:

- Keywords are reserved words in Python.
- Identifiers are names given to variables, functions, and classes.
- Literals are fixed values such as numbers, strings, and boolean values.

Introduction to Python Tokens

Tokens are the smallest individual units in a Python program. Everything in a Python program is built using tokens. Python has five types of tokens:

- Keywords are reserved words in Python.
- Identifiers are names given to variables, functions, and classes.
- Literals are fixed values such as numbers, strings, and boolean values.
- Operators are symbols used for mathematical and logical operations.

Introduction to Python Tokens

Tokens are the smallest individual units in a Python program. Everything in a Python program is built using tokens. Python has five types of tokens:

- Keywords are reserved words in Python.
- Identifiers are names given to variables, functions, and classes.
- Literals are fixed values such as numbers, strings, and boolean values.
- Operators are symbols used for mathematical and logical operations.
- Punctuators are special symbols that help structure the code.

Keywords in Python

Keywords are predefined words in Python that have a specific meaning.

Keywords in Python

Keywords are predefined words in Python that have a specific meaning. They cannot be used as variable names, function names, or any other identifiers.

Keywords in Python

Keywords are predefined words in Python that have a specific meaning. They cannot be used as variable names, function names, or any other identifiers. Some common keywords in Python are:

Keywords in Python

Keywords are predefined words in Python that have a specific meaning. They cannot be used as variable names, function names, or any other identifiers. Some common keywords in Python are:

- if, else, while, for, def, return

Keywords in Python

Keywords are predefined words in Python that have a specific meaning. They cannot be used as variable names, function names, or any other identifiers. Some common keywords in Python are:

- if, else, while, for, def, return
- import, class, try, except, break, continue

Identifiers in Python

Identifiers are the names used to represent variables, functions, classes, and other objects in Python.

Identifiers in Python

Identifiers are the names used to represent variables, functions, classes, and other objects in Python. Rules for defining identifiers:

Identifiers in Python

Identifiers are the names used to represent variables, functions, classes, and other objects in Python. Rules for defining identifiers:

- An identifier must start with a letter or an underscore.

Identifiers in Python

Identifiers are the names used to represent variables, functions, classes, and other objects in Python. Rules for defining identifiers:

- An identifier must start with a letter or an underscore.
- It can contain letters, digits, and underscores but cannot start with a digit.

Identifiers in Python

Identifiers are the names used to represent variables, functions, classes, and other objects in Python. Rules for defining identifiers:

- An identifier must start with a letter or an underscore.
- It can contain letters, digits, and underscores but cannot start with a digit.
- Identifiers are case-sensitive, meaning name and Name are different.

Identifiers in Python

Identifiers are the names used to represent variables, functions, classes, and other objects in Python. Rules for defining identifiers:

- An identifier must start with a letter or an underscore.
- It can contain letters, digits, and underscores but cannot start with a digit.
- Identifiers are case-sensitive, meaning name and Name are different.
- Python keywords cannot be used as identifiers.

Literals in Python

A literal is a constant value assigned to a variable or used directly in the code.

Literals in Python

A literal is a constant value assigned to a variable or used directly in the code. Python has different types of literals:

Literals in Python

A literal is a constant value assigned to a variable or used directly in the code. Python has different types of literals:

- Numeric literals, such as integers and floating-point numbers.

Literals in Python

A literal is a constant value assigned to a variable or used directly in the code. Python has different types of literals:

- Numeric literals, such as integers and floating-point numbers.
- String literals, which are enclosed in quotes.

Literals in Python

A literal is a constant value assigned to a variable or used directly in the code. Python has different types of literals:

- Numeric literals, such as integers and floating-point numbers.
- String literals, which are enclosed in quotes.
- Boolean literals, which are either True or False.

Operators in Python

Operators are symbols used to perform operations on variables and values.

Operators in Python

Operators are symbols used to perform operations on variables and values. Different types of operators in Python include:

Operators in Python

Operators are symbols used to perform operations on variables and values. Different types of operators in Python include:

- Arithmetic operators, such as addition, subtraction, multiplication, and division.

Operators in Python

Operators are symbols used to perform operations on variables and values. Different types of operators in Python include:

- Arithmetic operators, such as addition, subtraction, multiplication, and division.
- Relational operators, which compare values and return a boolean result.

Operators in Python

Operators are symbols used to perform operations on variables and values. Different types of operators in Python include:

- Arithmetic operators, such as addition, subtraction, multiplication, and division.
- Relational operators, which compare values and return a boolean result.
- Logical operators, which combine multiple boolean expressions.

Punctuators in Python

Punctuators are special symbols used to structure Python code.

Punctuators in Python

Punctuators are special symbols used to structure Python code. Examples of punctuators include:

Punctuators in Python

Punctuators are special symbols used to structure Python code. Examples of punctuators include:

- Parentheses, which are used in function calls.

Punctuators in Python

Punctuators are special symbols used to structure Python code. Examples of punctuators include:

- Parentheses, which are used in function calls.
- Brackets, which are used for lists and indexing.

Punctuators in Python

Punctuators are special symbols used to structure Python code. Examples of punctuators include:

- Parentheses, which are used in function calls.
- Brackets, which are used for lists and indexing.
- Braces, which are used in dictionaries.

Punctuators in Python

Punctuators are special symbols used to structure Python code. Examples of punctuators include:

- Parentheses, which are used in function calls.
- Brackets, which are used for lists and indexing.
- Braces, which are used in dictionaries.
- Comma, colon, and semicolon, which help in separating values and statements.

Website

www.postnetwork.co

Website

www.postnetwork.co

YouTube Channel

www.youtube.com/@postnetworkacademy

Website

www.postnetwork.co

YouTube Channel

www.youtube.com/@postnetworkacademy

Facebook Page

www.facebook.com/postnetworkacademy

Reach PostNetwork Academy

Website

www.postnetwork.co

YouTube Channel

www.youtube.com/@postnetworkacademy

Facebook Page

www.facebook.com/postnetworkacademy

LinkedIn Page

www.linkedin.com/company/postnetworkacademy

Thank You!