Python Tutorial - Learn Python Programming Language

Why to Learn Python?

Unknown • 4 min read • 2024-03-01
https://www.geeksforgeeks.org/python/python-programming-language-tutorial/

Python is one of the most popular programming languages. It's simple to use, packed with features and supported by a wide range of libraries and frameworks. Its clean syntax makes it beginner-friendly.

- A high-level language, used in web development, data science, automation, AI and more.
- Known for its readability, which means code is easier to write, understand and maintain.
- Backed by library support, so we don't have to build everything from scratch, there's probably a library that already does what we need.

Why to Learn Python?

- Requires fewer lines of code compared to other programming languages like Java.
- Provides Libraries / Frameworks like Django, Flask and many more for Web Development, and Pandas, Tensorflow, Scikit-learn and many more for, AI/ML, Data Science and Data Analysis
- Cross-platform, works on Windows, Mac and Linux without major changes.
- Used by top tech companies like Google, Netflix and NASA.
- Many Python coding job opportunities in Software Development, Data Science and AI/ML.

> Try our ongoing free course Python Skillup with weekly topic coverage, notes, daily quizzes and coding problems.

First Python Program

Here is a simple Python code, printing a string. We recommend you to edit the code and try to print your own name.

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

```
Hello World
```

1. Python Basics

In this section, we'll cover the basics of Python programming, including installing Python, writing first program, understanding comments and working with variables, keywords and operators.

> Before starting to learn python we need to install python on our system.

• Introduction

• Applications

• Input and Output

• Variables

• Operators

• **Quiz:** Basics, I/O

• Keywords

• Data Types

• **Quiz:** Data Types, Numbers, Boolean

• Conditional Statements

• Loops

• **Quiz:** Control Flow, Loops

2. Python Functions

In this section of Python 3 tutorial we'll explore Python function syntax, parameter handling, return values and variable scope. Along the way, we'll also introduce versatile functions like range(), map, filter and lambda functions.

• Functions

• Pass Statement in Function

• Global and Local Variables

• Recursion

• *args and **kwargs in Function

• 'Self' as Default Argument

• First Class Function

• Lambda Function

• Map, Reduce and Filter Function

• Inner Function

• Decorators

• **Quiz:** Functions

3. Python Data Structures

Python offers versatile collections of data types, including lists, string, tuples, sets, dictionaries and arrays. In this section, we will learn about each data types in detail.

• Strings

• List

• **Quiz:** List, String

• Tuples

• Dictionary

• **Quiz:** Tuples, Dictionary

• Sets

• Arrays

• List Comprehension

• **Quiz:** Sets, Arrays, List Comprehension

Python's collections module offers essential data structures, including the following:

• Counters

• Heapq

• Deque

• OrderedDict

• Defaultdict

• **Quiz:** Counters, Heapq, Deque, OrderedDict

> To learn data structure and algorithm with python in detail, you can refer to our DSA with Python Tutorial.

4. Python OOP Concepts

In this section, we'll explore the core principles of object-oriented programming (OOP) in Python. From encapsulation to inheritance, polymorphism, abstract classes and iterators, we'll cover the essential concepts that helps you to build modular, reusable and scalable code.

• Python OOP

• Classes and Objects

• Polymorphism

• Inheritance

• Abstraction

• Encapsulation

• Iterators

• **Quiz:** OOP

5. Python Exception Handling

In this section, we'll explore Python Exception Handling that how Python deals with unexpected errors, enabling us to write fault-tolerant code. We'll cover file handling, including reading from and writing to files.

• Exception Handling

• Built-in Exception

• User defined Exception

• **Quiz:** Exception Handling

6. File Handling

In this section, we will cover file handling, including reading from and writing to files.

• File Handling

• Read Files

• Write/Create Files

• OS Module

• pathlib Module

• Directory Management

• **Quiz:** File Handling

7. Python Database Handling

In this section we will learn how to access and work with MySQL and MongoDB databases

• Python MongoDB Tutorial

• Python MySQL Tutorial

8. Python Packages or Libraries

Python is a huge collection of Python Packages standard libraries that make development easier. These libraries help with a wide range of tasks and can save you a lot of time by providing ready-to-use tools.

Some commonly used types of libraries in Python include:

• Packages

• Built-in Modules

• DSA Libraries

• GUI Libraries

9. Data Science with Python

1. Foundational Libraries: These are the libraries that form the base for all data science work. Start here to build a strong foundation.

• NumPy

• Pandas

• Matplotlib

2. Advanced Visualization and Statistical Tools: Once you're comfortable with basic data handling and visualization, move to creating cleaner visuals and performing statistical analysis.

• Seaborn

• Statsmodel

3. Machine Learning Libraries: After data manipulation and visualization, learn machine learning, starting with simpler models and moving to advanced ones.

• Scikit-learn

• XGBoost /LightGBM

4. **Deep Learning Frameworks:** If you're interested in AI and deep learning, these libraries will allow you to build and train neural networks.

• TensorFlow and Keras

• PyTorch

> To learn more, you can refer to Python for Data Science.

10. Web Development with Python

1. Core Web Frameworks (Backend Development with Python): These are the tools for building Python-based web applications.

• Flask

• Django

2. Database Integration: Learn how to connect Python web frameworks to databases for storing and retrieving data.

• SQLite

• SQLAlchemy

• Django ORM

3. Front-End and Backend Integration: Learn how to connect Python backends with front-end technologies to create dynamic, full-stack web applications.

• Jinja2 (Flask)

• Django Templates

4. API Development: Learn to build APIs (Application Programming Interfaces) for connecting your backend with front-end apps or other services.

• Flask-RESTful

• Django REST Framework (DRF)

> To learn more, you can refer to Python for Web Development.

Python Practice

Python quiz page covers topics including variables, data types, input, output, lists, tuples,

dictionaries and sets. The Python Coding Practice Problems page offers exercises on loops, functions, lists, strings, dictionaries, sets and advanced structures like heaps and deques.

• Quizzes

• Python Coding Problems

> This Python tutorial is updated based on latest Python 3.13.1 version.

Python Introduction

How Python Programs are Executed

Comments in Python

Variables in Python

List Introduction

Set in Python

K

K