THIS COURSE HAS FOUR MODULS THESE ARE THE TEXTUAL VERSIONS OF THE COURSE CONTENT. EACH MODULE VIDEO CONTENTS TRANSCRIPTS (THE TIME STAMPS CAN BE IGNORED HERE) , BOOK CHAPTERS AND ONLINE ARTICLE, SAMPLE QUESTIONS AND ASNWERS

THE COURSES NAME IS "PYTHON PROGRAMMING 1"

MODULE1

IF YOU ARE THINKING LEARNING PYTHON OR IF YOU ARE LEARNING IT RECENTLY YOU MIGHT BE WONDERING WHAT CAN I USE PYTHON FOR EXACTLY WELL THAT'S ACTUALLY A TRICKY QUESTION TO ANSWER BECAUSE THERE ARE SO MANY APPLICATIONS OF PYTHON BUT OVERTIME I HAVE SEEN THAT THERE ARE THREE MAIN APPLICATIONS FOR PYTHON WEB DEVELOPMENT DATA SCIENCE INCLUDING MACHINE LEARNING DATA SCIENCE AND DATA VISUALIZATION AND SCRIPTING I AM GONNA TALK ABOUT EACH OF THEM IN END OF THIS VIDEO I AM ALSO GONNA MENTION GAME DEVELOPMENT EMBEDDED APPLICATIONS AND DESKTOP APPLICATIONS IN THE END OF THIS VIDEO AND AS USUAL I AM GONNA

Top 10 Python Applications in Real World
Last Updated : 23 Jul, 2025

We are living in a digital world that is completely driven by chunks of code. Every industry depends on software for its proper functioning be it healthcare, military, banking, research, and the list goes on. We have a huge list of programming languages that facilitate the software development process. One of these is Python which has emerged as the most lucrative and exciting programming language. As per a survey it is observed that Python is the main coding language for more than 80% of developers. The main reason behind this is its extensive libraries and frameworks that fuel up the process.
Top-10-Python-Applications-in-the-Real-World
Table of Content

1. Web Development
2. Machine Learning and Artificial Intelligence
3. Data Science
4. Game Development
5. Audio and Visual Applications
6. Software Development
7. CAD Applications
8. Business Applications
9. Desktop GUI
10. Web Scraping Application

Top Python Applications in Real World

Python has been at the forefront of Machine learning, Data Science, and Artificial Intelligence innovation. Further, Python applications offers provides ease in building a plethora of apps, web development processes, and a lot more. In this blog, we will discuss the top 10 Python applications in the real world in a detailed manner. So let's get started:

1. Web Development

It is one of the most astonishing applications of Python. This is because Python comes up with a wide range of frameworks like Django, Flask, Bottle, and a lot more that provide ease to developers. Furthermore, Python has inbuilt libraries and tools which make the web development process completely effortless. The use of Python for web development also offers:

Amazing visualization
Convenience in development
Enhanced security
Fast development process

2. Machine Learning and Artificial Intelligence

Machine Learning and Artificial Intelligence are the hottest subjects right now. Python along with its inbuilt libraries and tools facilitate the development of AI and ML algorithms. Further, it offers simple, concise, and readable code which makes it easier for developers to write complex algorithms and provide a versatile flow. Some of the inbuilt libraries and tools that enhance AI and ML processes are:

Numpy for complex data analysis
Keras for Machine learning
SciPy for technical computing
Seaborn for data visualization

3. Data Science

Data science involves data collection, data sorting, data analysis, and data visualization. Python provides amazing functionality to tackle statistics and complex mathematical calculations. The presence of in-built libraries provides convenience to data science professionals. Some of the popular libraries that provide ease in the data science process are TensorFlow, Pandas, and Socket Learning. These libraries provide an ecosystem for fine-tuning data models, data preprocessing, and performing complex data analysis.

4. Game Development

With the rapidly growing gaming industry, Python has proved to be an exceptional option for game development. Popular games like Pirates of the Caribbean, Bridge Commander, and Battlefield 2 use Python programming for a wide range of functionalities and add-ons.

The presence of popular 2D and 3D gaming libraries like pygame, panda3D, and Cocos2D makes the game development process completely effortless.

5. Audio and Visual Applications

Audio and video applications are undoubtedly the most amazing feature of Python. Python is equipped with a lot of tools and libraries to accomplish your task flawlessly. Applications that are coded in Python include popular ones like Netflix, Spotify, and YouTube. This can be handled by libraries like

Dejavu
Pyo
Mingus
SciPy
OpenCV

6. Software Development

Python is just the perfect option for software development. Popular applications like Google, Netflix, and Reddit all use Python. This language offers amazing features like:

Platform independence
Inbuilt libraries and frameworks to provide ease of development.
Enhanced code reusability and readability
High compatibility

Apart from these Python offers enhanced features to work with rapidly growing technologies like Machine learning and Artificial intelligence. All these embedded features make it a popular choice for software development.

7. CAD Applications

CAD refers to computer-aided design; it is the process of creating 3D and 2D models digitally. This application has replaced manual drift and is used by architects, product designers, and construction managers to design products with extremely high consistency. Python is embedded with amazing applications like Blender, FreeCAD, open cascade, and a lot more to efficiently design products. These provide enhanced features like technical drawing, dynamic system development, recordings, file export, and import.

8. Business Applications

Python offers excellent security and scalability features that make it perfect for delivering high-performance business applications. It has inbuilt libraries and tools like:

Odoo is business management software that provides you with an automated solution for your business process.
Tryton is easy-to-use open-source business software. It has fully integrated features like

financial accounting, sales, CRM, purchasing, shipping, and the list goes on.

All these distinguishing features make it fit for creating business applications.
9. Desktop GUI

Python is an interactive programming language that helps developers to create GUIs easily and efficiently. It has a huge list of inbuilt tools for Python usage are PyQT, kivy, wxWidgets, and many other libraries like them to build a fully functional GUI in an extremely secure and efficient manner.
10. Web Scraping Application

Web scraping is an automated process used to extract information from websites in an easier and faster way. The information is used by researchers, organizations, and analysts for a wide variety of tasks. Python has a wide range of features that make it suitable for web scraping some of them are:

A concise syntax enhances readability and saves you time.
A wide range of libraries and tools like pandas, matplotlib, and Selenium makes the web scraping process easy and efficient.
Easy to use and understand

Some other real-world applications of Python are:

Robotics and automation by the use of inbuilt libraries and tools like PyDy, Dart, PyRobot, and Pyro.
Image processing: some of the amazing libraries and tools for image processing are Blender, OpenCV, Houdini, and PIL.
Scientific applications are facilitated by popular libraries like Pandas, Matplotlib, SciPy, and many more

Also Read

Python Libraries
How to learn Python from Scratch?
Python Projects - Beginners to Advanced

Conclusion

Python is a concise and extremely powerful language & that's why the application of Python is rapidly gaining popularity. It has been the epicentre of the most amazing technologies like AI, automation, and machine learning. Further, it is used to facilitate hot subjects like data analysis and data visualization. In this blog, we have tried to give you a basic idea about the Top 10 Python applications in the real world. We hope that you found this helpful!

Top 10 Uses of Python in Real World with Examples
By Great Learning Editorial Team Updated on Dec 17, 2024
uses of python

Python has become the go-to programming language in 2024, dominating the market with a 28.11% share. Its simplicity and versatility make it ideal for a wide range of applications.

In this blog, we'll explore the top 10 real-world applications on uses of Python, illustrating why it remains the most popular choice in the tech community.
Academy Pro

Python Programming Course

In this course, you will learn the fundamentals of Python: from basic syntax to mastering data structures, loops, and functions. You will also explore OOP concepts and objects to build robust programs.
11.5 Hrs
51 Coding Exercises
Start Free Trial

Let's dive into the list

Web Development
Data Science
Artificial Intelligence and Machine Learning
Enterprise Applications
Education Sector
Web Scraping Applications
Game Development
Software Development
Desktop GUI
Operating Systems

Top 10 Uses Of Python In The Real World

Let's examine some real-world applications of Python programming. These examples will also provide amazing python project ideas, helping you explore diverse language possibilities.
1. Web Development

Web development is a widely known python usage example for creating dynamic, scalable, and maintainable web applications. It is known for its readability and efficiency, making it a popular choice for backend and full-stack development.

Key Frameworks and Tools

Django: A high-level Python web framework that encourages rapid development and clean, pragmatic design. It includes an ORM, authentication, and other essential tools.

Flask: A micro-framework that provides the essentials to get an application up and running, offering flexibility to developers.

Real-World Example

Instagram, one of the most popular apps in Python's real-time applications, uses Django to handle its massive user base and large amounts of data. Django's ability to manage large-scale applications is ideal for such high-traffic websites.

Join the thousands of learners already benefiting from our Free Web Development Courses.
2. Data Science

Python is a dominant language in data science due to its simplicity, extensive libraries, and active community. It is used for data analysis, visualization, and predictive modeling.

Key Libraries and Tools

Pandas: Provides data structures and data analysis tools.

NumPy: Supports large, multi-dimensional arrays and matrices and a collection of mathematical functions.

Matplotlib: A plotting library for creating static, interactive, and animated visualizations.

SciPy: Provides scientific and technical computing capabilities, including optimization and integration.

Real-World Example

Netflix uses Python for its recommendation algorithm and data analytics, leveraging its powerful libraries to enhance user experience by providing personalized content suggestions.
Academy Pro

Master Data Science Using Python

Learn Data Science with Python in this comprehensive course! From data wrangling to machine learning, gain the expertise to turn raw data into actionable insights with hands-

on practice.
12.5 Hrs
1 Project
Learn Data Science with Python
3. Artificial Intelligence and Machine Learning

Python is extensively used in AI and machine learning due to its simplicity and robust libraries. It supports the development of algorithms that allow machines to learn and make decisions.

Key Libraries and Tools

TensorFlow: An open-source library developed by Google for machine learning and deep learning applications.

Keras: An API designed for human beings, not machines, which allows for easy and fast prototyping.

Scikit-learn: A library for machine learning built on NumPy, SciPy, and Matplotlib.

PyTorch: An open-source machine learning library developed by Facebook's AI Research lab.

Real-World Example

Uber uses machine learning for ETA predictions, fraud detection, and dynamic pricing, which is another Python application in the real world that takes advantage of Python's robust libraries to improve service efficiency and customer satisfaction.
Academy Pro

Master Machine Learning with Python

Learn machine learning with Python! Master the basics, build models, and unlock the power of data to solve real-world challenges.
12 Hrs
1 Coding Exercise
Learn Machine Learning with Python
4. Enterprise Applications

Python is increasingly adopted for building enterprise-level applications due to its versatility, ease of integration, and a large ecosystem of libraries. It allows businesses to develop scalable solutions for various needs.

Key Frameworks and Tools

Tryton: A high-level, open-source application platform that provides a set of modules for various business needs, including accounting, inventory, and project management.

Odoo: A suite of open-source business applications, including CRM, eCommerce, and accounting, offering customization options and integration capabilities.

Pyramid: Known for its flexibility and scalability, Pyramid is suitable for easily developing complex enterprise applications.

Real-World Example

Dropbox utilizes Python for its backend infrastructure and to develop internal tools. Python's simplicity and robustness make it ideal for managing the platform's large-scale data storage and synchronization needs.

5. Education Sector

Python is widely adopted in education as an introductory programming language due to its simplicity and readability. Its English-like syntax, extensive standard library, and abundant learning resources help beginners grasp programming concepts easily.

Key Resources

Python Tutor: An online tool for visualizing Python code and its execution, helping students understand programming concepts.

Python.org Documentation: The official Python documentation provides comprehensive guides, tutorials, and reference materials for learners to explore Python's features and capabilities.

Abundance of Learning Resources: Free online tutorials, documentation, and interactive platforms make it easy for beginners to find diverse learning materials and practice coding in a supportive environment.

Real-World Example

The University of Texas at Austin incorporates Python into its introductory computer science courses, leveraging its clear syntax and extensive libraries to teach fundamental programming concepts effectively. Python's versatility allows students to transition smoothly from basic programming tasks to more complex projects.

Start learning Python for free with our expertly crafted "Free Python Courses."

6. Web Scraping Applications

Python's rich ecosystem of libraries makes it an excellent choice for web scraping, the process of extracting data from websites. It allows developers to retrieve and analyze information from the web efficiently.

Key Libraries and Tools

Beautiful Soup: A Python library for parsing HTML and XML documents, making web scraping easier and more manageable.

Scrapy: An open-source and collaborative web crawling framework for Python, providing robust features for extracting data from websites.

Selenium: A portable framework for automating web browsers, useful for scenarios where dynamic interaction with web pages is required.

Real-World Example

Price monitoring services often utilize Python for web scraping to gather pricing information from various e-commerce websites. This enables businesses to make data-driven decisions and stay competitive in the market.

Gain expertise in extracting data from the web with our expert-led "Free Course On Web Scraping with Python."

7. Game Development

Python's versatility extends to game development. Several frameworks and libraries cater to game creation, including graphics rendering, physics simulations, and user interface design.

Key Libraries and Tools

Pygame: A set of Python modules designed for writing video games, providing functionality for graphics, sound, and input handling.

PyKyra: A fast and versatile game development framework for Python, focusing on 2D games.

Pyglet: A cross-platform windowing and multimedia library for Python, suitable for creating games and multimedia applications.

PyOpenGL: A Python binding to OpenGL, allowing developers to create 3D graphics applications.

Kivy: An open-source Python library for developing multitouch applications with a natural

user interface, including games.

Panda3D: A game engine and framework for 3D rendering and game development in Python and C++.

Cocos2d: A framework for building 2D games, interactive demos, and other graphical/ interactive applications.

Real-World Example

Battlefield 2, a popular first-person shooter game, utilizes Python scripting for various game mechanics and functionalities, demonstrating Python's capability to contribute to developing and enhancing gameplay experiences.

8. Software Development & Engineering

Python's versatility extends to software development, which is used to build various applications, from command-line tools to desktop applications.

Its clean syntax and robust standard library make it suitable for rapid prototyping and production-grade software development.

Key Frameworks and Tools

PyQt: A set of Python bindings for the Qt application framework, enabling the development of cross-platform GUI applications.

Tkinter: Python's standard GUI toolkit, providing a simple and easy-to-use interface for building desktop applications.

Click: A Python package for creating command-line interfaces, allowing developers to build robust and user-friendly command-line tools.

Real-World Example

Dropbox Paper, a collaborative document-editing service, utilizes Python for its backend services and desktop application development, demonstrating Python's capability to support software development across different domains.

Join our "Software Engineering Courses" and become proficient in software design, development, and deployment.

9. Desktop GUI

Python is used to develop desktop graphical user interface (GUI) applications. It offers libraries and frameworks that simplify the creation of interactive and visually appealing

interfaces, making it suitable for building desktop applications across various platforms.

Key Libraries and Tools

Kivy: A Python framework for developing multitouch applications supporting various input methods and platforms.

wxPython: A cross-platform GUI toolkit for Python based on the wxWidgets library, offering native-looking interfaces on each platform.

PyGTK: Python bindings for the GTK+ toolkit, enabling developers to create graphical user interfaces for Linux and other Unix-like systems.

Real-World Example

Blender, a popular open-source 3D creation suite, utilizes Python for its GUI development, allowing users to create, animate, and render 3D models and animations with a powerful and customizable interface.

10. Operating Systems

Python's object-oriented design makes it suitable for building entire operating systems. Its compatibility with most platforms, such as Windows and Mac, and ease of use for native application development contribute to its effectiveness in this domain.

Key Tools and Techniques

Python OS Module: This module provides a way to interact with the operating system, allowing tasks such as file management, process management, and system information retrieval.

Python Shell Scripting: Python scripts can be used for system administration tasks, such as managing services, deploying updates, and configuring network settings.

PyInstaller: Converts Python scripts into standalone executables, which are useful for packaging and distributing Python applications on various operating systems.

Real-World Example

Red Hat Enterprise Linux (RHEL), a leading enterprise Linux distribution, incorporates Python for system administration and automation tasks.

Don't miss out on our 'Free Operating System Courses'—start learning today!

Conclusion

Python is one of the most versatile programming languages, evident in its top 10 real-world applications spanning various industries.

Given its prevalence in data science and AI, exploring Python's capabilities can open doors to lucrative career opportunities in these domains.

If you want to join this rewarding journey, consider enrolling in the Great Learning PG Program in Data Science.

With dual certification from UT Austin & Great Lakes, flexible learning options, weekly mentorship, and dedicated program support, you'll be equipped to excel in the dynamic field of data science.

FAQs

How does Python compare to other programming languages for web development?

Python's frameworks, like Django and Flask, offer rapid development, scalability, and maintainability, making it competitive with languages like JavaScript and PHP for web development. Its ease of use and robust ecosystem contribute to its popularity in this domain.

How does Python contribute to scientific research?

Python's libraries, such as SciPy, Matplotlib, and NumPy, are real time applications of Python that are widely used in scientific computing for tasks such as data analysis, visualization, and simulation. Its ease of use and robust ecosystem make it a preferred tool for researchers across various scientific disciplines.

How does Python handle big data processing and analytics?

Python's libraries, such as PySpark, Dask, and Apache Beam, enable distributed computing and parallel processing, making it suitable for big data processing and analytics tasks. These libraries leverage frameworks like Apache Spark and Hadoop for scalable data processing

MCQs for Python in Web Development, AI, ML, and Data Science (25 Questions)
1. Python for Web Development

Which of the following is NOT a popular Python web framework?

a) Django

b) Flask

c) React

d) Pyramid

What is the main feature of Django?

a) Microservices architecture

b) Batteries-included philosophy

c) Supports only SQL databases

d) Non-relational database support

Which Python package is commonly used for handling HTTP requests in Flask?

a) requests

b) urllib

c) jinja2

d) flask

Flask is best described as a:

a) Full-stack framework

b) Micro-framework

c) CLI tool

d) Web scraper

Which template engine is used by Flask by default?

a) Jinja2

b) Mako

c) Django Templates

d) Jupyter

2. Python for AI & Machine Learning

Which Python library is primarily used for data manipulation and analysis?

a) NumPy

b) Pandas

c) Matplotlib

d) Seaborn

In Machine Learning, which library is commonly used for building neural networks in Python?

a) Pandas

b) Keras

c) TensorFlow

d) PyTorch

Which of the following is an algorithm for supervised learning?

a) K-means clustering

b) Decision Trees

c) Principal Component Analysis (PCA)

d) DBSCAN

Which of the following is a popular Python library for Natural Language Processing (NLP)?

a) NLTK

b) NumPy

c) OpenCV

d) PyTorch

In machine learning, what does the acronym "SVM" stand for?

a) Statistical Variance Method

b) Support Vector Machine

c) Sample Vector Machine

d) Supervised Variable Model

3. Python for Data Science

Which of the following is NOT a Python visualization library?

a) Matplotlib

b) Seaborn

c) Plotly

d) NumPy

Pandas is often used for handling which type of data structure?

a) Arrays

b) DataFrames

c) Graphs

d) Neural Networks

Which of the following Python libraries is commonly used for scientific computing?

a) SciPy

b) NumPy

c) Pandas

d) All of the above

In which format does Pandas store data by default?

a) CSV

b) HDF5

c) DataFrame

d) JSON

The scikit-learn library is primarily used for:

a) Deep learning

b) Data visualization

c) Machine learning

d) Data manipulation

4. Python for Deep Learning and Neural Networks

Which Python framework is most popular for building deep learning models?

a) Django

b) Flask

c) TensorFlow

d) SciPy

Which of the following is NOT a type of neural network?

a) Convolutional Neural Networks (CNN)

b) Recurrent Neural Networks (RNN)

c) Random Forests

d) Multilayer Perceptrons (MLP)

What is the primary function of an activation function in a neural network?

a) To adjust learning rate

b) To calculate gradients

c) To introduce non-linearity

d) To split training and validation data

Which Python library is used for tensor manipulation in deep learning?

a) NumPy

b) PyTorch

c) TensorFlow

d) Keras

Which deep learning framework was developed by Facebook?

a) Keras

b) TensorFlow

c) PyTorch

d) Theano

5. Python for AI Models & Algorithms

Which Python library would you use to create a machine learning model using a decision tree?

a) sklearn

b) matplotlib

c) NumPy

d) seaborn

In supervised learning, what does the term "label" refer to?

a) Input data

b) The feature set

c) Output data

d) Data normalization

What type of problem does the k-nearest neighbors (KNN) algorithm solve?

a) Regression

b) Classification

c) Clustering

d) Optimization

What is overfitting in a machine learning model?

a) The model generalizes well to new data

b) The model fails to learn from training data

c) The model performs well on training data but poorly on new data

d) The model is too simple to capture data patterns

Which Python library is widely used for performing linear algebra operations like matrix multiplication?

a) Pandas

b) NumPy

c) SciPy

d) TensorFlow

MCQs for Python Frameworks in Deep Learning and Machine Learning (25 Questions)
1. Popular Python Frameworks in Machine Learning

Which framework is a high-level wrapper around TensorFlow for easier model development?

a) Keras

b) PyTorch

c) Fastai

d) Scikit-learn

PyTorch is developed by:

a) Microsoft

b) Google

c) Facebook

d) Amazon

What does the term "backpropagation" refer to in deep learning?

a) Forward pass through the network

b) Error correction through gradients

c) Model training via data augmentation

d) Data preprocessing

Which Python library is used for high-performance machine learning in large datasets?

a) Dask

b) TensorFlow

c) Scikit-learn

d) LightGBM

In machine learning, which of the following models is used for time-series forecasting?

a) K-means

b) LSTM

c) Logistic Regression

d) Random Forest

2. Deep Learning Frameworks

What does TensorFlow primarily focus on?

a) Static computation graphs

b) Dynamic computation graphs

c) Data preprocessing

d) Model visualization

Which is the primary purpose of the tf.keras module in TensorFlow?

a) To handle tensor operations

b) To define and train deep learning models

c) To build reinforcement learning models

d) To optimize hyperparameters

Which of the following is NOT a feature of PyTorch?

a) Dynamic computation graphs

b) Tensor manipulation

c) Static computation graphs

d) GPU support

What is the main advantage of using TensorFlow over PyTorch?

a) More user-friendly

b) Supports only CPU

c) Better for production deployment

d) More intuitive debugging

The transformers library by Hugging Face is primarily used for:

a) Time-series forecasting

b) Natural Language Processing

c) Data manipulation

d) Model deployment

3. ML Frameworks and Libraries

Which machine learning library is optimized for gradient boosting?

a) LightGBM

b) Scikit-learn

c) XGBoost

d) Keras

Which of the following is a deep learning library based on Theano?

a) PyTorch

b) Keras

c) TensorFlow

d) Fastai

In machine learning, what is the purpose of GridSearchCV in Scikit-learn?

a) Hyperparameter tuning

b) Data preprocessing

c) Cross-validation

d) Model evaluation

Which library provides the most comprehensive support for Reinforcement Learning (RL)?

a) TensorFlow

b) PyTorch