

Meta description: What programming language to choose for your AI development? Read the article to know why Python is the top pick for AI projects and how it brings value to ML applications across industries.

Why Python Is the Best Language for AI Development?

According to [the IDC study](#), being at the forefront of digital transformation initiatives, the AI software market is anticipated to reach \$240,27 billion by 2024.

When it comes to the practical side of AI implementation in software solutions, AI projects differ from traditional software projects. In order to develop an AI product, data scientists and machine learning developers are required to use appropriate technology stack, have solid skills, [big data](#), and deep research capabilities. According to O'Reilly's 2018 survey "[The State of Machine Learning Adoption in the Enterprise](#)", 81% of businesses employ data scientists, 39% - machine learning engineers, 20% - deep learning engineers. It is not an easy task to hire AI-related skilled specialists, and we would say, more challenging in comparison with hiring software engineers. Whereas traditional software engineering is focused on automating tasks by writing programs, ML engineering is aimed at instructing the computer to find a program that fits data.

The selection of the programming language for an AI project is essential and is defined by its ability to handle high-loaded advanced processes, be easy to support, have an extensive ecosystem of tools and libraries, etc. There are a lot of languages available like Java, C++, R, Javascript, and others. We turned to our [data scientists and ML engineers](#) to provide reasonable insights on what programming language is the most popular overall. Considering the overall popularity among machine learning researchers, Python leads the pack with 57% of data scientists and ML developers using it. Let's take a closer look at why Python is considered to be the top pick and what its main benefits are.

What Makes Python the Best Programming Language?

Being acknowledged by developers as the most intuitive and flexible programming language, Python empowers developers with extensive possibilities. Why do developers select Python for their AI projects? Below we outline the main advantages of Python.

Code Simplicity and Consistency

Being focused on the reduction of a number of lines of code, and consequently, on code simplification, Python is a great solution for machine learning projects that are apriori backed by complex deep learning algorithms and versatile workflows. With the "check as you code" methodology, Python allows reducing the burden of testing the code. Therefore, instead of handling technical issues of the language, developers are more concentrated on ML-enabled decision making.

Owing to Python's simple syntax and a low entry barrier, developers can easily learn the language and build complex models for machine learning projects. When it comes to collaboration between the developer's team, Python language is a good choice as well.

Large Ecosystem of Libraries and Frameworks

Considering the fact that machine learning algorithms are complex and are often tricky to implement, an extensive range of Python frameworks and libraries may be a rescue for developers who want to reduce development time. There is no need to build base-level items from the very beginning every time. With a rich technology stack, Python offers a set of libraries with pre-written codes to perform an action making the development process accelerated. Here are the following:

- [Tensorflow](#). It is an end-to-end open-source ML platform for AI-powered applications with a focus on training and inference of deep neural networks with huge datasets.
- [Scikit-learn](#). Is used for handling such ML algorithms as classification, regression, and clustering.
- SciPy. Focused on advanced scientific computing, this library contains such modules as integration, optimization, linear algebra, image procession, etc.
- Keras. Acting as an interface for the TensorFlow library, Keras is a user-friendly and extensible library aimed at simplifying coding for developing deep neural networks and allowing users to develop deep models on smartphones, on the web, and Java Virtual Machine.
- [Pybrain](#). It is mainly used for reinforcement and supervised learning. While Pybrain offers easy-to-use and flexible algorithms, there are also algorithms for state-of-the-art research.
- [NLTK](#). Natural Language Toolkit can be applied across industries to work with computational linguistics, human language data, and text analysis.

Flexibility

There are several benefits that Python's flexibility feature covers. Firstly, since there is no need to recompile the source code using Python language, developers are entitled to see rapid results once the changes are implemented. Secondly, flexibility allows combining Python with other languages which leads to advanced results. The possibility to choose either OOPs or scripting gives the developers advantages while struggling with complex algorithms.

With Python language, developers are empowered with an advantage to choose the programming style they prefer or even combine them to accomplish their goals:

- **Imperative style**. A sequence of computations is performed as a direct change in the program state. In this style, the computer gets commands on what steps to take to solve a problem.

- **The functional style.** Unlike imperative style, the functional style supports solution-focused thinking and is aimed at telling the computer what result has to be achieved. It is presented in mathematical equation form and does not consider the program state.
- **The object-oriented style.** Based on data fields that are treated as objects, OOP style structure programs by modeling real-world entities and relations between them. Due to the inability to implement encapsulation (data hiding), Python does not fully support this paradigm.
- **Procedural style.** With a step-by-step approach, tasks are placed in functions that are called as needed. Procedural style is considered to be the most popular among beginners and is used for sequencing, selection, and iteration.

Platform Independence

As a platform-independent language, Python can run on any platform: Windows, Linux, MacOS, and more others. In order to prepare code for running on various platforms and be used to create standalone executable programs for most common operating systems, developers need to modify only some lines of code without installing a Python interpreter. Pyinstaller packages are a great solution for an efficient process making it more comfortable and simple.

Community and Popularity

According to the [Stack Overflow](#) developer survey among 65,000 developers in 2020, Python takes 3d place (66,7%) among the developers who are developing with Python language, and the 1st place (30%) among the developers who have expressed interest in developing with it.



We can see that the popularity of Python is growing. There are a lot of forums and tutorials where developers exchange experiences and use cases related to machine learning solutions. Moreover, since Python is an open-source programming language, there are plenty of tutorial materials and documentation along with developers' support community online.

How Python Brings Value to AI Development Across Industries

Now it is obvious from the above-mentioned information that Python with its simplicity, flexibility, and with a range of frameworks and libraries, is suitable for the development of AI-powered applications. Let's take a look at how businesses leverage Python for AI and ML products.

- Travel. [Skyscanner](#), a world-known metasearch engine and travel agency, applied a Python-based ML algorithm to make travel offers customized by suggesting new airplane routes based on customers' preferences and needs, comparing thousands of destinations.
- Healthcare. [AiCure](#) is an advanced data analytics application allowing to monitor patient behavior and provide them with meaningful and high-quality data to maintain treatment regimens.
- Transportation. In order to solve Uber's most challenging problems, data scientists developed [Michelangelo PyML](#) (integration with the initial Michelangelo ML platform). This solution makes it easy to serve arbitrary Python-based models at scale for online and offline use cases.

Summary

With a widely developed library ecosystem, Python is considered to be the top choice for machine learning products since complex operations can be executed without the need for extensive coding. Thanks to platform independence, tests can be run on any platform without the need for significant code modifications.

Since Python's popularity has been growing, it is important to hire experienced AI developers and data scientists who have solid competencies to develop complex ML-based projects. If you still have questions or looking for Python developers to help you jump-start your AI project [get in touch with us](#).