

Date: 19th May-2025

THE HISTORY AND CREATION OF PROGRAMMING LANGUAGES

Ozodbek Usmonov Olti o'g'li

2nd-year student of the Faculty of Applied Mathematics and Intellectual
Technologies, Mirzo Ulugbek National University of Uzbekistan,

ozodbekusmonov435@gmail.com

Anotatsiya: Ushbu maqolada dasturlash tillarining yaratilish tarixi va rivojlanishi haqida umumiy ma'lumotlar berilgan. C#, Java, Python, C++ kabi mashhur dasturlash tillarining paydo bo'lishi, ularning asosiy xususiyatlari va qo'llanish sohalari tahlil qilinadi. Zamonaviy dasturlash tillarining dasturchilar uchun qanday imkoniyatlar yaratishi ko'rsatib o'tilgan.

Annotation: This article provides an overview of the history and development of programming languages. It analyzes the emergence, key features, and applications of popular languages such as C#, Java, Python, and C++. The article highlights the opportunities modern programming languages offer to developers.

Аннотация: В данной статье представлен обзор истории и развития языков программирования. Рассматриваются появление, основные особенности и области применения популярных языков, таких как C#, Java, Python и C++. Отмечаются возможности, которые современные языки программирования предоставляют разработчикам.

Kalit so'zlar: Dasturlash tillari, Yaratilish tarixi, C#, Java, Python, C++, Dasturlash paradigmalari, Ilovalar, Kompyuter texnologiyalari

Keywords: Programming languages, History of creation, C#, Java, Python, C++, Programming paradigms, Applications, Computer technologies

Ключевые слова: Языки программирования, История создания, C#, Java, Python, C++, Парадигмы программирования, Приложения, Компьютерные технологии.

Nowadays, computer technologies are rapidly advancing, and programming languages are at the core of this development. A programming language is a special language used to give precise instructions to a computer, transforming human thoughts into a form understandable by machines. With programming languages, various software, applications, games, websites, and even artificial intelligence systems are created.

Each programming language stands out with its unique features, areas of application, and history. Some are simple and easy to learn, while others are designed for high-level control and efficiency. Modern programming languages often offer a combination of object-oriented, functional, and procedural paradigms.

C# Programming Language

C# (pronounced "C-sharp") is a modern, multi-paradigm programming language developed by Microsoft in 2000. It was created by a team led by renowned software



Date: 19th May-2025

engineer Anders Hejlsberg. C# was initially introduced together with the .NET Framework and was optimized specifically to work within this environment.

C# is a fully object-oriented language that supports key OOP principles such as encapsulation, inheritance, and polymorphism. To ensure memory safety, C# includes features like automatic memory management (garbage collection) and a strong type system (type safety). Its syntax closely resembles that of C, C++, and Java, making it easier to learn for developers familiar with those languages. While it was originally designed for Windows only, C# now supports cross-platform development through .NET Core and .NET 5/6/7, including Linux, macOS, mobile, and web platforms.

C# is used in many fields. Some notable areas include:

Desktop Applications: Creating GUI-based programs using Windows Forms and WPF.

Web Development: Building dynamic websites and web applications with ASP.NET.

Mobile Development: Developing iOS and Android apps using Xamarin technology.

Game Development: Unity game engine uses C# as its primary scripting language.

Cloud Computing: Many C#-based services are used in the Microsoft Azure cloud platform.

Advantages of C# include: strong IDE support (Visual Studio), rich libraries and APIs; excellent documentation and community support; built-in support for parallel and asynchronous programming (async/await).

C# is a modern, reliable, and feature-rich programming language that allows developers to efficiently build large and complex systems. Playing a crucial role within the Microsoft ecosystem, this language continues to evolve and remains one of the most popular programming languages in use today.

C++ Programming Language

C++ is a powerful, efficient, and multi-paradigm programming language developed in 1983 by Bjarne Stroustrup at Bell Laboratories. It was created as an extension of the C programming language, adding object-oriented programming (OOP) features. C++ fully supports OOP principles such as encapsulation, inheritance, and polymorphism. It allows system-level access to memory and hardware resources, offering high performance. The language supports procedural, object-oriented, and generic (template-based) programming paradigms.

Speed and Efficiency: C++ is widely used in performance-critical systems and game development. **Manual Memory Management:** Developers have full control over memory allocation and deallocation, allowing for precise system optimization.

Here are some key areas where C++ is commonly used:

System Software: Operating systems, device drivers, and high-performance applications.

Date: 19th May-2025

Game Development: Many popular game engines (such as Unreal Engine and CryEngine) are built using C++.

Embedded Systems: Software designed for specific hardware devices.

Financial Systems: Banking and financial applications that require high-speed calculations and precision.

Graphics and Multimedia: Visual effects and animation software requiring high efficiency.

Key Advantages of C++: high performance and speed, strong control and flexibility, support for multiple programming paradigms, large and active community, extensive libraries and frameworks (such as STL – Standard Template Library).

C++ is a powerful, flexible, and high-performance language that serves as a core tool for developing system software, games, and complex real-time applications. It remains a top choice for scenarios requiring high control and efficiency and is still widely used in the modern software development world.

Python Programming Language

Python is a high-level, general-purpose programming language known for its simplicity and readability. It was created in 1991 by Guido van Rossum in the Netherlands. Originally designed to teach programming concepts, Python is now widely used in fields such as scientific computing, artificial intelligence, web development, and more. Python is an interpreted, dynamically typed, multi-paradigm language. Its clear and concise syntax makes it ideal for both beginners and experienced developers.

Some major areas where Python is used include:

Web Development: Building robust web applications using frameworks like Django and Flask.

Data Analysis and AI: Libraries like NumPy, Pandas, TensorFlow, and Scikit-learn are used extensively.

Automation and Scripting: Automating tasks and managing systems with simple scripts.

Scientific Computing: Used in academic and research environments with tools like Matplotlib, SciPy, and Jupyter.

Game and GUI Development: Creating games and interfaces using Pygame and Tkinter.

Key Advantages of Python: easy-to-read syntax, vast library ecosystem, cross-platform compatibility, strong community and documentation, rapid prototyping capability. Python is a widely used and versatile programming language that balances simplicity with powerful functionality. It helps developers build fast, effective, and reliable solutions across diverse domains.

Programming languages have become an integral part of today's digital world. Each language is distinguished by its purpose, area of application, and technical capabilities. Popular languages such as C++, C#, Java, and Python are widely used in various fields — from system programming to artificial intelligence.



Date: 19th May-2025

C++ offers high performance and low-level control, while C# provides powerful tools for modern application development within the Microsoft ecosystem. Java is well known for its cross-platform capabilities, making it ideal for enterprise and mobile applications. Python stands out for its simplicity and rich library ecosystem, making it a leading choice in scientific computing and artificial intelligence.

Therefore, understanding the capabilities of different programming languages and choosing the right one is crucial for developers. Each language has its own strengths and serves as an effective tool for solving different kinds of problems.

REFERENCES:

1. Kernighan, B. W., & Ritchie, D. M. (1988). The C Programming Language. Prentice Hall.
2. Stroustrup, B. (2013). The C++ Programming Language (4th ed.). Addison-Wesley.
3. Albahari, J., & Albahari, B. (2021). C# 10 in a Nutshell: The Definitive Reference. O'Reilly Media.
4. Schildt, H. (2018). Java: The Complete Reference (11th ed.). McGraw-Hill Education.
5. Lutz, M. (2013). Learning Python (5th ed.). O'Reilly Media.
6. Zandstra, M. (2019). PHP Objects, Patterns, and Practice. Apress.
7. Downey, A. (2015). Think Python: How to Think Like a Computer Scientist (2nd ed.). O'Reilly Media.

