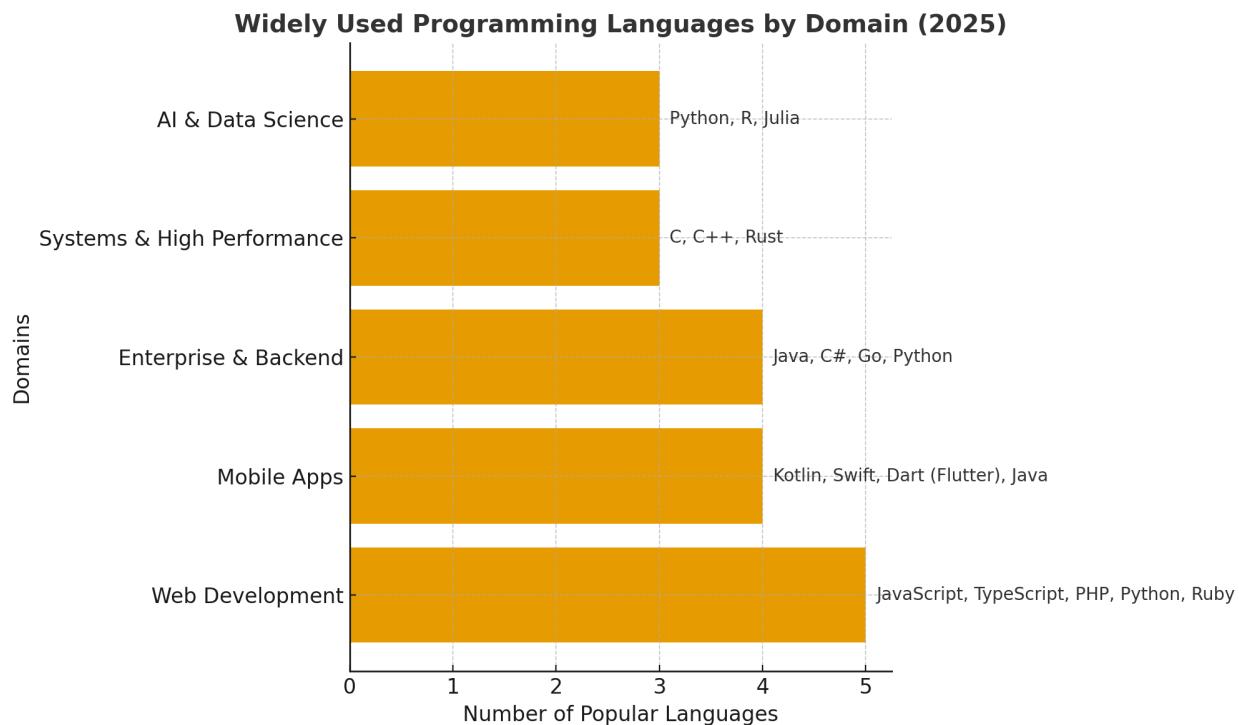




Widely Used Programming Languages (2025)



1. Python

- **Why Popular?**
 - Easy to learn, simple syntax (great for beginners).
 - Huge library support for AI/ML, data science, and web development.
- **Use Cases:**

- Artificial Intelligence (TensorFlow, PyTorch)
- Data Science & Analytics (Pandas, NumPy, Jupyter)
- Web Development (Django, Flask, FastAPI)
- Automation & Scripting

Example:

```
print("Hello, Python!")
```



2. Java

- **Why Popular?**

- “Write once, run anywhere” → JVM makes it portable.
- Enterprise-grade reliability.

- **Use Cases:**

- Banking, e-commerce, large-scale enterprise systems.
- Android app development.
- Backend APIs (Spring Boot).

Example:

```
public class Hello {  
    public static void main(String[] args) {  
        System.out.println("Hello, Java!");  
    }  
}
```



3. JavaScript (JS)

- **Why Popular?**

- Runs in every browser → backbone of web development.
- Can be used for both frontend & backend (with Node.js).

- **Use Cases:**

- Web development (React, Angular, Vue).
- Backend (Node.js, Express.js).
- Mobile apps (React Native).

Example:

```
console.log("Hello, JavaScript!");
```



4. C# (C-Sharp)

- **Why Popular?**

- Developed by Microsoft, tightly integrated with .NET ecosystem.
- Strong for Windows apps, enterprise software, and games.

- **Use Cases:**

- Game development (Unity engine).
- Desktop apps (Windows Forms, WPF).
- Web APIs with ASP.NET Core.

Example:

```
Console.WriteLine("Hello, C#!");
```

-
-

5. C / C++

- **Why Popular?**

- Close to hardware, high performance.
- Still dominant in system-level programming.

- **Use Cases:**

- Operating systems, embedded systems.
- Game engines.
- High-performance apps (databases, compilers).

Example (C++):

```
#include <iostream>
using namespace std;
int main() {
    cout << "Hello, C++!" << endl;
    return 0;
}
```

-
-

6. Go (Golang)

- **Why Popular?**

- Created by Google → fast, simple, and concurrent.

- Excellent for cloud-native applications.

- **Use Cases:**

- Cloud services (Kubernetes, Docker).
- Microservices.
- Networking tools.

Example:

```
package main
import "fmt"
func main() {
    fmt.Println("Hello, Go!")
}
```

●

7. Rust

- **Why Popular?**

- Memory safety + high performance.
- Loved by developers for reliability.

- **Use Cases:**

- Systems programming.
- WebAssembly.
- High-security apps.

Example:

```
fn main() {
    println!("Hello, Rust!");
}
```

}

-
-

8. PHP

- **Why Popular?**

- Powers a huge chunk of the web (WordPress, Drupal).
 - Still widely used despite competition.

- **Use Cases:**

- Web development (Laravel, WordPress).

Example:

```
<?php  
echo "Hello, PHP!";  
?>
```

-
-

9. TypeScript (TS)

- **Why Popular?**

- Superset of JavaScript with static typing.
 - Helps prevent bugs in large projects.

- **Use Cases:**

- Large-scale frontend apps.
 - Backend APIs with Node.js.

Example:

```
let message: string = "Hello, TypeScript!";
console.log(message);
```

-
-

10. Kotlin

- **Why Popular?**

- Official language for Android development (replacing Java).
- Modern, concise, and interoperable with Java.

- **Use Cases:**

- Android apps.
- Backend development (Ktor, Spring).

Example:

```
fun main() {
    println("Hello, Kotlin!")
}
```

-
-



Summary of Language Usage

- **Web Development:** JavaScript, TypeScript, PHP, Python, Ruby.
- **Mobile Apps:** Kotlin, Swift, Dart (Flutter), Java.
- **Enterprise & Backend:** Java, C#, Go, Python.

- **Systems & High Performance:** C, C++, Rust.
- **AI & Data Science:** Python, R, Julia.