**What are mobile app programming languages?**

Mobile app programming languages are specialized coding languages used to develop applications for mobile devices like smartphones and tablets. These languages determine how an app functions, its performance, and its compatibility with different operating systems (iOS, Android, etc.).

**Types of mobile application programming language**

1. **Swift:**

Swift is a powerful and intuitive programming language for all Apple platforms, with a concise-yet-expressive syntax and modern features developed uniquely for the modern iOS development process. The main advantages of Swift are the ability to enhance the readability, speed, and performance of the system. In comparison to Python, it boosts the speed of the system up to eight times. Swift can also reduce errors in the development process and ensure a reliable and trendy mobile application.

1. **Kotlin:**

Kotlin is a statically typed, object-oriented programming language that is interoperable with the Java virtual machine (JVM), Java Class Libraries and Android . it is used in High-performance Android apps (e.g., Netflix, Pinterest).The Kotlin programming language was originally designed to improve the [Java](https://www.theserverside.com/definition/Java) programming language and is often used in conjunction with Java. Despite being the preferred development language of Android, Kotlin's interoperability with Java has led it to be used with many application types.

1. **Java:**

Java is an extremely transferable programming language used across platforms and different types of devices, from smartphones to smart TVs. It's used for creating mobile and web apps, enterprise software, Internet of Things (IoT) devices, gaming, big data, distributed, and cloud-based applications among other types. It is characterised by High quality learning resources, Inbuilt functions and libraries and High-quality development tools.

1. **Dart:**

Dart is a client-optimized language for developing fast apps on any platform. Its goal is to offer the most productive programming language for multi-platform paired with a [flexible execution runtime platform](https://dart.dev/overview" \l "platform) for app frameworks.Dart is a versatile programming language developed by Google, primarily used for front-end development. Its main use case is in Flutter development (Cross-Platform Apps), Google's UI toolkit for building cross-platform mobile, web, and desktop apps from a single codebase.Example: Apps like Google Ads, eBay Motors, and BMW use Flutter, powered by Dart.

1. **C#:**

C# is a cross-platform general purpose language that makes developers productive while writing highly performant code.C# programs can run on many different devices, from Internet of Things (IoT) devices to the cloud and everywhere in between.It is Used before Swift was introduced in 2014· Used before Swift was introduced in 2014 it has a More complex syntax compared to Swift its Still supported but gradually being phased out

1. **JavaScript:**

JavaScript, originally designed for web development, has evolved into a full-stack language that can also be used for mobile app development. With the rise of frameworks like React Native, Ionic, and NativeScript, JavaScript enables developers to create cross-platform mobile applications that run on iOS and Android with a single codebase.

JavaScript has a Cross-Platform Compatibility ,Rich Ecosystem and thousands of libraries and frameworks.

JavaScript can be used in mobile development through three main approaches:

* **Hybrid Apps (Web-Based)**
  + Hybrid apps use HTML, CSS, and JavaScript wrapped inside a native WebView (a browser-like component). These apps are platform-independent but rely on web technologies.
* **Cross-Platform Native Apps**
  + Unlike hybrid apps, cross-platform native frameworks use JavaScript to create truly native mobile apps by interacting with the native APIs.
* **Progressive Web Apps (PWAs)**
  + PWAs are web apps that act like mobile apps by using service workers for offline capabilities. Users can install them on their home screen without downloading them from an app store.

Comparism between the different Mobile Programming Languages

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Feature** | **JavaScript (React Native, Ionic, Cordova)** | **Swift (iOS)** | **Kotlin (Android)** | **Flutter (Dart)** | **C# (Xamarin, MAUI)** | **Java (Android)** |
| **Platform Support** | iOS & Android (cross-platform) | iOS only | Android only | iOS & Android (cross-platform) | iOS & Android (cross-platform) | Android only |
| **Performance** | Medium to High (near-native in React Native) | High | High | High (compiles to native) | Medium to High | High |
| **Code Reusability** | Yes (write once, run on both) | No | No | Yes | Yes | No |
| **Ease of Learning** | Easy (especially for web developers) | Moderate | Moderate | Moderate | Easy | Moderate |
| **Development Speed** | Fast (hot reload, reusable components) | Slower | Slower | Fast (hot reload) | Fast | Slower |
| **UI Customization** | Limited (relies on native UI components) | Full | Full | Full (custom UI engine) | Full | Full |
| **Native API Access** | Limited (requires bridges or plugins) | Full | Full | Full | Full | Full |
| **App Size** | Larger due to framework overhead | Smaller | Smaller | Larger due to Flutter engine | Larger | Smaller |
| **Community Support** | Large (React Native, Ionic, Cordova) | Large | Large | Growing fast | Medium | Large |
| **Best For** | Cross-platform apps, web-to-mobile transition | iOS apps, Apple ecosystem | Android apps, Google ecosystem | Beautiful, high-performance cross-platform apps | Enterprise apps, Microsoft integration | Android apps, enterprise applications |
| **Used By** | Facebook, Instagram, Airbnb, UberEats | Apple, Lyft, LinkedIn | Google, Pinterest, Netflix | Google Ads, Alibaba, eBay | Microsoft apps, Siemens | Twitter, Spotify, LinkedIn |

**References**

**<https://www.ibm.com>**

<https://www.builder.ai/>

<https://newwavesolution.com>

<https://developer.ibm.com>

<https://developer.apple.com/swift/>

<https://www.techtarget.com>

<https://aws.amazon.com/>

<https://developer.mozilla.org/en-US/docs/Learn_web_development/Core/Scripting>

<https://dart.dev/overview>