**Java vs .NET: Core Features And Differences Between The Two**

Microsoft .NET and Java are two leading technologies for building softwares, websites and web apps. With their growing popularity, most of the businesses face the challenge to choose from either of them as a primary development tool for creating intuitive applications. Both these technologies enable the creation of large-scale business applications and have evolved over the years to support and enhance desktop & server-side application development.

Between the two, you must first learn about the applications created using either of the technologies. Through this comparison blog, we aim to highlight the difference between Java and .NET. Our objective is to help businesses understand how either of the environments can fit in their operational requirements.

|  |  |  |
| --- | --- | --- |
| **Feature** | **Microsoft .NET** | **Java** |
| **Programming Languages** | C#, VB.NET, C++, .NET, PHP, Ruby, Python & more | Java, Clojure, Groovy, Scala, PHP, Ruby, Python, JavaScript & more |
| **Runtime** | CLR | JVM |
| **Supported IDE** | Microsoft Visual Studio, Rider, MonoDevelop | Eclipse, IntelliJ Idea, Oracle NetBeans, and Oracle JDeveloper |
| **Server Components** | .NET COM, OLE Automation | EJBs, JCA, JMX |
| **GUI Frameworks** | WPF, WinForms, UWP | JavaFX, Swing GUI Java, AWT, SWT & more |
| **Web Services Support** | Built-in | Add-on |
| **Unit Testing** | Microsoft Unit Testing Framework, NUnit | JUnit |
| **Web Application Framework** | ASP.NET Core, Spring .NET | Spring, Apache Wicket, JSF, Struts & more |
| **Web Server Scripting** | ASP.NET | JSF |
| **Data Access** | ADO.NET, oLeDB, Dapper & more | JDBC |
| **HTTP Engine** | IIS | Application Servers from Multiple Vendors, Glassfish, Tomcat & More |
| **Remoting** | SOAP, OpenAPI, DCOM & More | RMI, Rest API, GraphQL & More |

A .NET or Microsoft technology-based solution might be the correct choice for an Enterprise-grade application which requires strict security and high level of data integrity, whereas a [Java-based solution](https://www.rishabhsoft.com/java-application-development) would be suitable when the primary requirement is cross-platform operability in-line with its motto “write once, deploy

**.NET vs Java – The Differences**

Java is object-oriented programming (OOPs) language and .NET is a framework with C# as its programming language. Both Java and .NET are based on the object-oriented concept and are useful for developing enterprise solutions.

Both are intended to simplify the development of apps by providing a system of modular, standardized components and services. The following head-to-head Java vs .NET comparison can help businesses make a better choice.

**.NET Framework vs. Java**

* **Platform**  
  Multi-platform portability is ensured by compiling source code into an intermediate language executable by all Java Virtual Machines (JVM). The JVM translates the code into bytecode making it compatible with the machine code according to the operating system on which it is installed.  
  The .NET framework is secure than most of the open-source platforms. It comes with the Common Language Runtime (CLR) framework, which supports the use of components, developed in multiple languages. Further, Microsoft has even developed a CLR engine that converts the program code into the Microsoft Intermediate Language (MSIL) and finally “just in time,” translates it into native machine code.
* **Language Support**  
  The peculiarity of Java lies in the sharing of a single language across different platforms. However, the programs written with it work independently across different OS types. Java supports programming and scripting languages like Python, Ruby, Groovy, Scala, and Kotlin.  
  The DotNet framework supports languages especially for backend development and web services, like C# and C++. It currently supports about 20 languages. It allows you to program in any language you choose (including, Vb.NET, C # .NET, Perl and many others). However, it generates a specific code for the Windows platform only.  
  Therefore, while .NET supports a multi-programming environment; Java is focused on a single programming language that supports multiple environments. Learn more about the top [Java benefits for SMBs](https://www.rishabhsoft.com/blog/java-benefits-for-small-businesses).
* **IDE**  
  Java IDE comprises of a code editor, compiler, and debugger. Eclipse, IntelliJ Idea, Oracle NetBeans, and Oracle JDeveloper are the main IDEs designed to make writing and testing of the code easier. These IDEs come with inbuilt plugins and auto-fill options, which boosts Java’s flexibility and provides scope for innovation.  
  The .NET platform is integrated with Visual Studio, which allows editing, compiling, and run-time customization of the behavior of APIs using standard library macros. Further, developers do not need to evaluate the IDEs and other tools in advance.
* **Third-Party Integrators**  
  Java facilitates the integration of third-party tools and offers developers the freedom to choose their choice of OS during development.  
  .NET offers integrated services like SharePoint and Microsoft Exchange. It is rich in functions designed for creating applications on the Windows platform. Interoperability is unique to .NET, which allows its applications to run seamlessly on other platforms too.
* **Performance & Compatibility**  
  Java does not require conversion to machine language until the code gets executed. While .NET is compiled and then run on the system where they are deployed. That is why C# works better in a runtime environment.  
  Versions of Java older than Java7 does not support the simplified data structure, switch case.  
  .NET supports switch case for string variable in C# and native generic data. Query wise, Linq (Language Integrated Query) is not supported in Java unlike in .NET, but the latter allows questions on stored procedures.
* **Security**  
  For open source platforms like Java, security always remains a concern due to a lack of professional support. Since .NET is of a proprietary platform, Microsoft takes care of the security aspect. It provides round-the-clock support for its enterprise clients.

**.NET or Java – The Final Choice**

There cannot be a perfect conclusion to the .NET vs. Java comparison. It comprises of understanding the actual business requirements of the company. ASP.NET will leverage Rapid Application Development (RAD) solutions, while Java will dominate large-scale enterprise projects.

**Choose Java if:**

* You want an enterprise-grade application
* Seeking Portability & Platform independence
* Your application attracts high user volumes

**Choose .NET if:**

* Develop web services
* Highly secure application
* Feature-packed, intuitive application with a rich GUI