

VB.Net Interview Questions for Freshers Pdf

Question: 1

How many languages .NET is supporting now?

When .NET was introduced it came with several languages. VB.NET, C#, COBOL and Perl, etc.

The site DotNetLanguages.Net says 44 languages are supported.

Question: 2

What you mean by .NET framework?

The .NET framework is a software development framework from Microsoft.

It is a collection of tools, technologies, and languages that provides an environment to build and deploy robust enterprise applications easily.

.Net supports the following features:

Interoperability: .NET includes a large library and supports several programming languages hence enable us to use codes written in different languages.

Common Runtime Engine: It provides a runtime environment for the execution of code written in .NET languages. It manages the execution of .NET code including memory management, debugging and exception handling.

Language Independence: Through Common language infrastructure specifications (CLI), .NET allows exchanging of data type between programs developed in different languages.

Base Class Library: It contains classes, value types, and interfaces required in application development. The library is organized as namespaces and can be easily used in any application by just including the namespaces at the top.

Supports ADO.NET- It provides access to data sources such as Microsoft SQL server, XML etc. so, it can be used in applications to connect to these data sources and retrieve, manipulate, and update data.

Microsoft AJAX Library – It is used to create highly responsive web applications with minimum efforts.

Question: 3

What is IL?

Whenever we compile our code in .NET, it is converted to Intermediate Language (IL) which is also known as Microsoft Intermediate Language or common Intermediate Language.

The IL code is in binary format and is used to debug our code at low level.

Question: 4

What is difference between VB and VB.NET?

Following are the differences between VB and VB.NET:

VB6 is a powerful and simple language used to create desktop Window applications or reusable ActiveX libraries. VB.NET is an enhancement to VB6 and used for developing web applications, distributed applications, web services etc. VB.NET uses the .NET Runtime and Framework that adds several features like faster program execution and better memory management etc.

VB6 uses ASP to build web applications where as VB.NET uses ASP.NET to build web Applications.

VB6 uses the VB-Runtime for running programs whereas VB.NET uses the .NET's Common Language Runtime Which allows it to run on different platforms and operating systems

VB6 interprets the code whereas VB.NET compiles the code. Moreover the compiled code is cached to increase efficiency

VB6 uses COM (Component Object Model) as component architecture where as VB.NET uses assemblies as its component architecture. It means handling versioning and deploying applications is easier in VB.NET than in VB6

VB6 uses ADODB for accessing database where as VB.NET uses ADO.NET which is comparatively more efficient and also supports the disconnected data access.

We cannot create multithreaded applications in VB6. In VB.NET, we can create multi threaded applications.

Question: 5

What are CLR, CTS and CLS?

Common Language Runtime or (CLR)

It is a Runtime environment of .NET Framework.

It is a layer of the .NET Framework that converts the source code into an intermediate language called the Microsoft Intermediate Language.

CLR performs the following tasks:

Manages the code while execution

Provides various services required in execution of an application such as memory management, garbage collection, thread management, exception handling, security checks etc.

Manages versioning, deployment support, and debugging.

Manages different .NET Types.

Common Type System (CTS)

It is a formal specification that documents how types are declared, used and managed in order to be used by the CLR.

Every language running on the .NET platform has a base set of data types provided by CTS.

CTS ensure that the data types of objects written in various languages are able to interact with each other.

It also defines the rules to ensure that objects written in different languages can interact with each other.

Common Type System supports following types:

Value types – value types directly contain their data. Instances of value types are either allocated on the stack or allocated inline in a structure. Value types can be built-in, user-defined, or enumerations.

Reference types – Reference types store a reference to the value's memory address and are allocated on the heap. Examples of reference types include arrays, pointer typ