

41 Dot Net (.NET) Framework Interview Questions and Answers

 itexamsusa.blogspot.com/2022/07/41-dot-net-net-framework-interview.html

41 Dot Net (.NET) Framework Interview Questions and Answers



**41 Dot Net (.NET)
Framework
Interview Questions
and Answers**

Want to switch to that high paying job? Or are you already been preparing hard to give interview the next weekend?

Do you know how many people get rejected in interviews by preparing only concepts but not focusing on actually which questions will be asked in the interview?

Don't be that person this time.

This is the most comprehensive Dot Net (.NET) framework interview questions book that you can ever find out. It contains:

here we have presented 41 most frequently asked and important Dot NET (.NET) framework interview questions and answers

Wide range of questions which cover not only basics in new Microsoft dot net framework but also most advanced and complex questions which will

help freshers, experienced professionals, senior developers, testers to crack their interviews.

41 Dot Net (.NET) Framework Interview Questions and Answers

1. What Is Microsoft.net?

.NET is a set of technologies designed to transform the internet into a full-scale distributed platform. It provides new ways of connecting systems, information and devices through a collection of web services. Also, It- provides a language independent, consistent programming model across all tiers of an application.

The goal of the .NET platform is to simplify web development by providing all of the tools and technologies that one needs to build distributed web applications.

2. What Is .net Framework?

The .NET Framework is set of technologies that form an integral part of the .NET Platform. It is Microsoft's managed code programming model for building applications that have visually stunning user experiences, seamless and secure communication, and the ability to model a range of business processes.

The .NET Framework has two main components: the common language runtime (CLR) and .NET Framework class library. The CLR is the foundation of the .NET framework and provides a common set of services for projects that act as building blocks to build up applications across all tiers. It simplifies development and provides a robust and simplified environment which provides common services to build application. The .NET framework class library is a collection of reusable types and exposes features of the runtime. It contains of a set of classes that is used to access common functionality.

3. What Is Clr?

The .NET Framework provides a runtime environment called the Common Language Runtime or CLR. The CLR can be compared to the Java Virtual Machine or JVM in Java. CLR handles the execution of code and provides useful services for the implementation of the program. In addition to executing code, CLR provides services such as memory management, thread management, security management, code verification, compilation, and other system services. It enforces rules that in turn provide a robust and secure execution environment for .NET applications.

4. What Is Cts?

Common Type System (CTS) describes the datatypes that can be used by managed code. CTS defines how these types are declared, used and managed in the runtime. It facilitates cross-language integration, type safety, and high performance code execution. The rules defined in CTS can be used to define your own classes and values.

5. What Is Cls?

Common Language Specification (CLS) defines the rules and standards to which languages must adhere to in order to be compatible with other .NET languages. This enables C# developers to inherit from classes defined in VB.NET or other .NET compatible languages.

6. What Is Managed Code?

The .NET Framework provides a run-time environment called the Common Language Runtime, which manages the execution of code and provides services that make the development process easier. Compilers and tools expose the runtime's functionality and enable you to write code that benefits from this managed execution environment. The code that runs within the common language runtime is called managed code.

7. What Is Msil?

When the code is compiled, the compiler translates your code into Microsoft intermediate language (MSIL). The common language runtime includes a JIT compiler for converting this MSIL then to native code. MSIL contains metadata that is the key to cross language interoperability. Since this metadata is standardized across all .NET languages, a program written in one language can understand the metadata and execute code, written in a different language. MSIL includes instructions for loading, storing, initializing, and calling methods on objects, as well as instructions for arithmetic and logical operations, control flow, direct memory access, exception handling, and other operations.

8. What Is Jit?

JIT is a compiler that converts MSIL to native code. The native code consists of hardware specific instructions that can be executed by the CPU.

Rather than converting the entire MSIL (in a portable executable [PE] file) to native code, the JIT converts the MSIL as it is needed during execution. This converted native code is stored so that it is accessible for subsequent calls.

9. What Is Portable Executable (pe)?

PE is the file format defining the structure that all executable files (EXE) and Dynamic Link Libraries (DLL) must use to allow them to be loaded and executed by Windows. PE is derived from the Microsoft Common Object File Format (COFF). The EXE and DLL files created using the .NET Framework obey the PE/COFF formats and also add additional header and data sections to the files that are only used by the CLR.

10. What Is An Application Domain?

Application domain is the boundary within which an application runs. A process can contain multiple application domains. Application domains provide an isolated environment to applications that is similar to the isolation provided by processes. An application running inside one application domain cannot directly access the code running inside another application domain. To access the code running in another application domain, an application needs to use a proxy.

11. How Does An Appdomain Get Created?

AppDomains are usually created by hosts. Examples of hosts are the Windows Shell, ASP.NET and IE. When you run a .NET application from the command-line, the host is the Shell. The Shell creates a new AppDomain for every application. AppDomains can also be explicitly created by .NET applications.

12. What Is An Assembly?

An assembly is a collection of one or more .exe or dll's. An assembly is the fundamental unit for application development and deployment in the .NET Framework. An assembly contains a collection of types and resources that are built to work together and form a logical unit of functionality. An assembly provides the CLR with the information it needs to be aware of type implementations.

13. What Are The Contents Of Assembly?

A static assembly can consist of four elements:

Assembly manifest-Contains the assembly metadata. An assembly manifest contains the information about the identity and version of the assembly. It also contains the information required to resolve references to types and resources.

Type metadata-Binary information that describes a program.

Microsoft intermediate language(MSIL)code.

A set of resources.

14. What Are The Different Types Of Assembly?

Assemblies can also be private or shared. A private assembly is installed in the installation directory of an application and is accessible to that application only. On the other hand, a shared assembly is shared by multiple applications. A shared assembly has a strong name and is installed in the GAC.

We also have satellite assemblies that are often used to deploy language-specific resources for an application.

15. What Is A Dynamic Assembly?

A dynamic assembly is created dynamically at run time when an application requires the types within these assemblies.

680 Dot Net (.NET) Framework Interview Questions and Answers book
[Get Book Here](#)

16. What Is A Strong Name?

You need to assign a strong name to an assembly to place it in the GAC and make it globally accessible. A strong name consists of a name that consists of an assembly's identity (text name, version number, and culture information), a public key and a digital signature generated over the assembly. The .NET Framework provides a tool called the Strong Name Tool (Sn.exe), which allows verification and key pair and signature generation

17. What Is Gac? What Are The Steps To Create An Assembly And Add It To The Gac?

The global assembly cache (GAC) is a machine-wide code cache that stores assemblies specifically designated to be shared by several applications on the computer. You should share assemblies by installing them into the global assembly cache only when you need to.

Steps

-Create a strong name using sn.exe tool eg: sn -k mykey.snk
-in AssemblyInfo.cs, add the strong name eg: [assembly:



```
AssemblyKeyFile("mykey.snk")]
```

-recompile project, and then install it to GAC in two ways:

drag & drop it to assembly folder (C:\WINDOWS\assembly OR C:\WINNT\assembly) (shfusion.dll tool)

```
gacutil -i abc.dll
```

18. What Is The Caspol.exe Tool Used For?

The caspol tool grants and modifies permissions to code groups at the user policy, machine policy, and enterprise policy levels.

19. What Is A Garbage Collector?

A garbage collector performs periodic checks on the managed heap to identify objects that are no longer required by the program and removes them from memory.

20. What Are Generations And How Are They Used By The Garbage Collector?

Generations are the division of objects on the managed heap used by the garbage collector. This mechanism allows the garbage collector to perform highly optimized garbage collection. The unreachable objects are placed in generation 0, the reachable objects are placed in generation 1, and the objects that survive the collection process are promoted to higher generations.

21. What Is Ilasm.exe Used For?

Ilasm.exe is a tool that generates PE files from MSIL code. You can run the resulting executable to determine whether the MSIL code performs as expected.

22. What Is Ildasm.exe Used For?

Ildasm.exe is a tool that takes a PE file containing the MSIL code as a parameter and creates a text file that contains managed code.

23. What Is The Resgen.exe Tool Used For?

ResGen.exe is a tool that is used to convert resource files in the form of .txt or .resx files to common language runtime binary .resources files that can be compiled into satellite assemblies.

24. What Is Boxing And Unboxing?

==Boxing

Value type to object type. Allocates memory on Heap.

==UnBoxing

Object type to value type. Allocates memory on Stack

25. How Can We Make A Thread Sleep For Infinite Period ?

You can also place a thread into the sleep state for an indeterminate amount of time by calling Thread.Sleep (System.Threading.Timeout.Infinite). To interrupt this sleep you can call the Thread.Interrupt method.

26. In Which Format You Can Pass The Value In The Sleep Function?

In milliseconds

27. What's Thread.sleep() In Threading ?

Thread's execution can be paused by calling the Thread.Sleep method. This method takes an integer value that determines how long the thread should sleep. Example Thread.CurrentThread.Sleep(2000). it will be paused for 2 seconds.

28. How Can You Reference Current Thread Of The Method ?

"Thread.CurrentThread" refers to the current thread running in the method. "CurrentThread" is a public static property.

29. What Does Addressof Operator Do In Background ?

The AddressOf operator creates a delegate object to the BackgroundProcess method. A delegate within VB.NET is a type-safe, object-oriented function pointer. After the thread has been instantiated, you begin the execution of the code by calling the Start() method of the thread.

30. Different Levels Of Priority Provided By .net.

- i) ThreadPriority.Highest
- ii) ThreadPriority.AboveNormal
- iii) ThreadPriority.Normal
- iv) ThreadPriority.BelowNormal
- v) ThreadPriority.Lowest

31. Is There Any Thread In Our .net Programs?

.NET program always has at least two threads running one is the main program and second is the garbage collector.

32. Namespace For The Thread Class?

All threading classes are defined in System.Threading namespace

33. Can We Have Multiple Threads In One App Domain ?

One or more threads run in an AppDomain. An AppDomain is a runtime representation of a logical process within a physical process. Each AppDomain is started with a single thread, but can create additional threads from any of its threads.

34. Did Vb6 Support Multi-threading ?

While VB6 supports multiple single-threaded apartments, it does not support a freethreading model, which allows multiple threads to run against the same set of data.

35. What Is A Thread ?

A thread is the basic unit to which the operating system allocates processor time.

36. What Is Multi-threading ?

Multi-threading forms subset of Multi-tasking. Instead of having to switch between programs this feature switches between different parts of the same program. Example you are writing in word and at the same time word is doing a spell check in background.

37. What Is Multi-tasking ?

It's a feature of modern operating systems with which we can run multiple programs at same time example Word, Excel etc.

38. What Is Equivalent For Regsvr32 Exe In .net ?

Regasm

39. How Do We Create Dcom Object In Vb6?

Using the CreateObject method you can create a DCOM object. You have to put the server name in the registry.

40. Can You Explain What Is Dcom ?

DCOM differs from COM in that it allows for creating objects distributed across a network, a protocol for invoking that object's methods, and secures access to the object. DCOM provides a wrapper around COM, hence it is a backwards compatible extension. DCOM uses Remote Procedural Calls (RPC) using Open Software Foundation's Distributed Computing Environment.

41. What Is Satellite Assembly? And Steps To Create Satellite Assembly?

When you write a multilingual or multi-cultural application in .NET, and want to distribute the core application separately from the localized modules, the localized assemblies that modify the core application are called satellite assemblies.

=====

for more IT Knowledge, visit <https://itexamtools.com/>

check my IT blog - <https://itexamsusa.blogspot.com/>

check my Medium IT articles - <https://itcertifications.medium.com/>

Join my facebook IT group - <https://www.facebook.com/groups/itexamtools>

check IT stuff on Pinterest <https://in.pinterest.com/itexamtools/>

find my IT stuff on twitter - https://twitter.com/texam_i