Q 5) How .net runtime generate executable files from your code?

Ans.

I) Virtual machine languages makes use of both a compiler and an interpreter. The compiler converts the source code into a kind of average machine language. In Java, this average machine language is called bytecode. In Visual Studio.NET languages, this average machine language is called MSIL (Microsoft Intermediate Language). (To keep the discussion on this page simpler, this compiled code will be referred to generically as bytecode from this point on.) The interpreter for virtual machine languages is a special program that provides the runtime libraries for the given operating system. That means that there is a different virtual machine interpreter for all of the supported operating systems.

II) way that virtual machine programming languages get some of the speed of compiled languages is that the source code is run through the compiler to create the bytecode. That conversion takes place before the program is ever run. The way that virtual machine languages gain their portability (platform independence) is by having a different interpreter for each supported operating system. This interpreter ties in the correct runtime libraries for each different operating system. The compiled bytecode is an average machine language that will work without changes with any of the virtual machine interpreters for that language. This process is illustrated next. We have a compiler that converts the source code into bytecode. This can be simulated by clicking on the Compile button. Once the bytecode has been created, that same bytecode can be used without any changes on any operating system that has a virtual machine interpreter for the programming language. Note that each of the virtual machine interpreters have different runtime library code, because each operating system has different runtime libraries. This is how the virtual machine language gets around platform dependency problems. Click Run Windows, Run Mac OSX or Run Linux to simulate running the program on any of those operating systems.