

Chapter 1 : .NET Framework - Wikipedia

Network Programming in racedaydvl.com Framework Describes the layered, extensible, and managed implementation of Internet services that you can quickly and easily integrate into your apps. racedaydvl.com Framework Apps.

Each chapter concludes with a Summary. Classes-Wrapping Data and Behavior Together. Introducing the Framework Class Library. The Elements of a Namespace. Programming with the Framework Class Library. Forms, Menus, and Controls. Key Classes Related to Windows Forms. The Form Class Hierarchy. Visual Characteristics of Forms. Working with Menu Items. An Introduction to Controls. Font, Text, and Printing Operations. Font, Text, and Printing. FontPad, a Simple Text Editor. Adding Printing Capabilities to FontPad. Stream and File Operations. Directory and File Operations. Reading and Writing to Files and Streams. Adding Open and Save to FontPad. Key Classes Related to Network Programming. Using the WebClient Class. A Socket Transmitter Application. Key Classes Related to Drawing. A Forms-Based Drawing Application. Reading and Writing XML. Introducing the XmlNodeReader Class. The Hotel Reservations Desk. Key Classes Related to Threading. Understanding and Applying Threads. Basic Operations with Threads. Variables and Their Scope. Key Classes Related to Messaging. Client Request and Server Response. Data Storage and Access. Key Classes Related to Data. An Overview of ADO. Updating Data Directly to a Database. Managing Cached Data and Schemas. Key Classes Related to Directory Services. Basics of Directory Services. Accessing Objects in the Directory. Creating a Serviced Component. Managing Collections of Objects. Profiling, Debugging, and Exception Handling. Handling Errors with Structured Exception Handlers. Globalization and Localization Techniques. Working with Regional Data. Deploying, Configuring, and Licensing. The Basic Unit of Deployment. Deploying Private and Shared Components. Deploying Assemblies into the GAC. Using Application Configuration Files. You will need the Free Adobe Acrobat Reader to view the file s for the book. If you do not already have Acrobat installed on your machine, click the "Get Acrobat Reader" button to download and install. Click on the links below to display the PDF file in a new window. Right-click on the link and select Save As if you want to download it to your hard drive.

Chapter 2 : A new look for .NET Reference Source | .NET Blog

Quick Reference for Effective Programming in racedaydvl.com Framework! Quick, accurate answers to all racedaydvl.com technology questions are within easy reach--just grab your copy of ".NET Framework Programmer's Reference.

There present variety of motives behind it due to which the readers quit reading the eBooks at their first most attempt to use them. Yet, there exist some techniques that can help the readers to have a good and powerful reading experience. A person ought to fix the correct brightness of display before reading the eBook. Due to this they suffer with eye sores and head aches. The best solution to overcome this acute problem is to decrease the brightness of the screens of eBook by making particular changes in the settings. An excellent eBook reader should be set up. You can also use free software that may offer the readers that have many functions to the reader than simply a simple platform to read the desired eBooks. Aside from offering a place to save all your precious eBooks, the eBook reader software even give you a great number of attributes as a way to boost your eBook reading experience compared to the standard paper books. You can even improve your eBook reading experience with help of options supplied by the software program for example the font size, full display mode, the particular number of pages that need to be shown at once and also change the color of the backdrop. You should not use the eBook continuously for a lot of hours without breaks. You must take appropriate breaks after specific intervals while reading. Nonetheless, this does not mean that you ought to step away from the computer screen every now and then. Constant reading your eBook on the computer screen for a long time without taking any rest can cause you headache, cause your neck pain and suffer with eye sores and also cause night blindness. So, it is critical to provide your eyes rest for some time by taking rests after specific time intervals. This will help you to prevent the problems that otherwise you may face while reading an eBook continuously. While reading the eBooks, you should favor to read big text. Normally, you will see that the text of the eBook will be in moderate size. It is suggested to read the eBook with enormous text. So, increase the size of the text of the eBook while reading it on the monitor. It is suggested that never use eBook reader in full screen mode. It is suggested not to go for reading the eBook in full-screen mode. Although it might look simple to read with full-screen without turning the page of the eBook fairly frequently, it put ton of stress on your eyes while reading in this mode. Always favor to read the eBook in the exact same length that will be similar to the printed book. This really is so, because your eyes are used to the span of the printed book and it would be comfortable for you to read in exactly the same way. Test out different shapes or sizes until you find one with which you will be comfortable to read eBook. By using different techniques of page turn you can also boost your eBook experience. You can try many methods to turn the pages of eBook to improve your reading experience. Check out whether you can turn the page with some arrow keys or click a special section of the display, aside from using the mouse to handle everything. Lesser the movement you need to make while reading the eBook better is going to be your reading experience. This will definitely help make reading easier. By using all these effective techniques, you can definitely enhance your eBook reading experience to a great extent. This advice will help you not only to prevent particular risks that you may face while reading eBook consistently but also facilitate you to enjoy the reading experience with great comfort. The download link provided above is randomly linked to our ebook promotions or third-party advertisements and not to download the ebook that we reviewed. We recommend to buy the ebook to support the author. Thank you for reading.

Chapter 3 : C Sharp (programming language) - Wikipedia

Programming Microsoft LINQ rcedaydvl.com Framework 4 (Developer Reference) rcedaydvl.com, rcedaydvl.com, rcedaydvl.com, rcedaydvl.com, rcedaydvl.com Download Note: If you're looking for a free download links of Programming Microsoft LINQ rcedaydvl.com Framework 4 (Developer Reference) pdf, epub, docx and torrent then this site is not for you.

You can refer to details on .NET Standard Microsoft docs. .NET Framework release after. If you use Visual Studio and use NuGet 3. This also removes the need for binding redirects when using .NET Standard libraries on .NET Runtime GC Performance Improvements. Usually the LOH allocator waits for the whole duration of the BGC sweep process before it can satisfy requests to allocate memory. This can hinder performance. .NET has always allowed developers to store user credentials with hashed passwords in configuration files. SHA-1 is still the default to preserve compatibility. Refer to the following sample to leverage this new feature. SHA-2 support for Message.HashAlgorithm In previous versions, if application code specified a System.Messaging does not do any hashing with these values. Following snippet illustrates how you can enable hashing on a queue and create a message with these new values. Configuration builders Configuration builders allow developers to inject and build configuration for applications at runtime, allowing configuration data to be pulled from sources beyond the traditional. In previous versions of the .NET Framework, configuration has been static. Applications only draw configuration data from a limited chain of. With Configuration Builders, applications can apply a custom-defined set of builders to any section of config. These builders are free to modify the configuration data contained in the given config section, or build it entirely from scratch possibly drawing new data from new sources that are not static files. To use the Configuration Builders feature, developers simply need to declare builders in config, then apply them to configuration sections with the ConfigBuilders tag. To implement custom Configuration Builder, developers can inherit from the System. Here are some code samples that will enable you to declare, use and apply configuration builders. .NET processes requests in its predefined pipeline which includes 23 events. .NET executes each event handler as an execution step. .NET selectively flows only the HttpContext which may not be sufficient for ambient context scenarios. With this feature we enable modules to restore ambient data. The ExecutionStepInvoker is intended for libraries that care about the execution flow of the application tracing, profiling, diagnostics, transactions, etc. We have added a new API to enable this: Now, we have provided support for a new API that allows for a standardized way to create an HttpCookie object from a string and accurately capture properties of the cookie like expiration date, path, the secure indicator. Furthermore, it assigns cookie values appropriately. Since the syntax for C# 7. ValueTuple, this should make migrating from System.Tuple to using the new tuple syntax easier. This attribute will be used by the compiler to mark members that have readonly-ref return types or parameters. If the compiler is running against an older. At compile time the API provides a way to do that statically through reflection. Whenever the compiler needs to check for runtime support, it would look for the corresponding well-known enum member, for instance, System.If the member exists, then the feature check is successful or the feature is supported. The value for that enum member is ignored. At runtime the check for feature support is done by calling a static method. This is enabled by the addition of the framework type RuntimeFeature. Tools can query it by calling the static method bool IsSupported string to check whether the feature is supported or not, by passing in the string name for a given feature. Libraries that generate code at runtime, like C# Scripting, would benefit from being able to detect whether the runtime supports Portable PDBs or not. Emitting Portable PDBs has performance benefits; it is faster and has much smaller memory footprint. In absence of this new API the library would need to resort to hard-coding build numbers of the mscorlib or conservatively assume that. Following sample illustrates how this can be passed. It is recompiled to target the. Applications that target the. Windows Forms Accessibility improvements Windows Forms accessibility changes are in the following areas: Windows 10 has changed the values for some high contrast system colors and Windows Forms is based on the Windows 10 Win32 framework. For the best experience, run on the latest version of Windows and opt in to the latest OS changes by adding an app. DataGridViewCell

now reports the correct read-only status to Narrator. Narrator can now read Disabled ToolStripMenuItem text when previously it would skip over disabled menu items. UI Accessibility Patterns Developers of accessibility technology tools will now be able to leverage common UI Accessibility patterns and properties for several WinForms controls. However, if a UI element changes somewhere in the screen and it is not being focused at that point in time, the user may not be notified, and so they may be missing important information. LiveRegions are meant to solve this problem. A developer can use them to inform the screen reader, or any other UIAutomation client, that an important change has been made to a UI element. The screen reader can then make decisions of its own as to how and when to inform the user of this change. AutomationProperties , as well as Set and Get methods. A new enumeration for the possible values of LiveSetting has been added to System. How to make a LiveRegion? Announcing an important UI change When the data changes on your LiveRegion, and you feel the need to inform a screen reader about that change, you need to explicitly raise an event as illustrated by the following sample. Screen reader You can observe the following accessibility improvements in the screen reader area after you opt-in to the Accessibility improvements in. Expander control The focus visual for the expander control is now visible.

Chapter 4 : Programming with the .NET Framework

DOWNLOAD NOW» The definitive reference every ricedaydvl.com developer needs to increase productivity inside ricedaydvl.com Framework. This book answers the primary questions that ricedaydvl.com developers need answered as they start to develop code.

Collapse the table of content Expand the table of content This documentation is archived and is not being maintained. This documentation is archived and is not being maintained. NET is a general purpose development platform. It can be used for any kind of app type or workload where general purpose solutions are used. It has several key features that are attractive to many developers, including automatic memory management and modern programming languages, that make it easier to efficiently build high-quality apps. NET are available, based on open. NET Standards that specify the fundamentals of the platform. NET Framework, which is one of the existing. This topic will also talk about other. NET technologies and where you can find their related documentation. It can be installed locally with your app with only the packages you need. It provides a lightweight development model and the flexibility to work with your favorite development tools on your favorite development platform. NET Core page to find installation instructions for each supported platform. You can currently use. NET Core to develop console or Web applications: To read about developing modern cloud-based Web application, see the ASP. To read about working with data, see the Entity Framework documentation. NET Framework, see the overview. We update the latest versions of the. NET Framework documentation on a regular basis with content fixes and enhancements, but we do not maintain older versions. Earlier versions of the. NET Framework documentation are available from the table of contents pane on the left. You can use the. To read about creating Windows 8. For information about creating portable. For additional information about developing apps, visit the Windows desktop apps and web development sections of the MSDN Library. You can use Visual Studio for your development tasks and select from a wide range of programming languages. NET Framework also releases out-of-band packages with new functionality and improved cross-platform support. For information about these, see The. You can extend the capabilities of your apps with the following.

Chapter 5 : .Net Framework Programmer's Reference PDF Kindle - RajAlexandros

NET is an integral part of many apps running on Windows and provides common functionality for those apps to run. For developers, racedaydvl.com Framework provides a comprehensive and consistent programming model for building apps that have visually stunning user experiences and seamless and secure.

However, the language specification does not state the code generation requirements of the compiler: Typing[edit] C supports strongly typed implicit variable declarations with the keyword `var`, and implicitly typed arrays with the keyword `new[]` followed by a collection initializer. C supports a strict Boolean data type, `bool`. Statements that take conditions, such as `while` and `if`, require an expression of a type that implements the `true` operator, such as the Boolean type. The only implicit conversions by default are those that are considered safe, such as widening of integers. This is enforced at compile-time, during JIT, and, in some cases, at runtime. No implicit conversions occur between Booleans and integers, nor between enumeration members and integers except for literal 0, which can be implicitly converted to any enumerated type. Enumeration members are placed in their own scope. The C language does not allow for global variables or functions. All methods and members must be declared within classes. Static members of public classes can substitute for global variables and functions. Metaprogramming[edit] Metaprogramming via C attributes is part of the language. Methods can be void or can return something like string, integer, double, decimal, float and bool. If a method is void it means that the method does not return any data type. The type dynamic allows for run-time method binding, allowing for JavaScript-like method calls and run-time object composition. C has support for strongly-typed function pointers via the keyword `delegate`. Synchronized], and has support for mutually-exclusive locks via the keyword `lock`. Property[edit] C provides properties as syntactic sugar for a common pattern in which a pair of methods, accessor getter and mutator setter encapsulate operations on a single attribute of a class. Memory access[edit] In C, memory address pointers can only be used within blocks specifically marked as unsafe, and programs with unsafe code need appropriate permissions to run. Most object access is done through safe object references, which always either point to a "live" object or have the well-defined null value; it is impossible to obtain a reference to a "dead" object one that has been garbage collected, or to a random block of memory. An unsafe pointer can point to an instance of a value-type, array, string, or a block of memory allocated on a stack. Code that is not marked as unsafe can still store and manipulate pointers through the `System.IntPtr` type, but it cannot dereference them. Managed memory cannot be explicitly freed; instead, it is automatically garbage collected. Garbage collection addresses the problem of memory leaks by freeing the programmer of responsibility for releasing memory that is no longer needed. Exception[edit] Checked exceptions are not present in C in contrast to Java. This has been a conscious decision based on the issues of scalability and versionability. However, unlike Java, C supports operator overloading. A developer can query any. Intellisense support, strong filtering capabilities, type safety with compile error checking ability, and brings consistency for querying data over a variety of sources. Common type system[edit] C has a unified type system. For example, every type inherits a `ToString` method. Categories of data types[edit] CTS separates data types into two categories: Value types are derived from `System.ValueType`, always have a default value, and can always be created and copied. Some other limitations on value types are that they cannot derive from each other but can implement interfaces and cannot have an explicit default parameterless constructor. Examples of value types are all primitive types, such as `int` a signed bit integer, `float` a bit IEEE floating-point number, `char` a bit Unicode code unit, and `System.DateTime` identifies a specific point in time with nanosecond precision. Other examples are `enum` enumerations and `struct` user defined structures. In contrast, reference types have the notion of referential identity - each instance of a reference type is inherently distinct from every other instance, even if the data within both instances is the same. This is reflected in default equality and inequality comparisons for reference types, which test for referential rather than structural equality, unless the corresponding operators are overloaded such as the case for `System`. In general, it is not always possible to create an instance of a reference type, nor to copy an existing instance, or perform a value comparison on two existing instances, though specific reference types can provide such services by exposing a

public constructor or implementing a corresponding interface such as `ICloneable` or `IComparable`. Examples of reference types are `object` the ultimate base class for all other C# classes, `System.String` a string of Unicode characters, and `System.Array` a base class for all C# arrays. Both type categories are extensible with user-defined types.

Boxing and unboxing [edit] Boxing is the operation of converting a value-type object into a value of a corresponding reference type. Unboxing is the operation of converting a value of a reference type previously boxed into a value of a value type. A boxed object of type `T` can only be unboxed to a `T` or a nullable `T`.

Libraries [edit] The C# specification details a minimum set of types and class libraries that the compiler expects to have available. The following is a very simple C# program, a version of the classic "Hello world" example: Each line has a purpose: In this case, when the compiler sees use of the `Console` type later in the source code, it tries to find a type named `Console`, first in the current assembly, followed by all referenced assemblies. In this case the compiler fails to find such a type, since the name of the type is actually `System.Console`. The compiler then attempts to find a type named `System.Console` by using the `System` prefix from the `using` statement, and this time it succeeds. The `using` statement allows the programmer to state all candidate prefixes to use during compilation instead of always using full type names. Everything between the following pair of braces describes `Program`. NET runtime calls the `Main` method. `Main` may also be called from elsewhere, like any other method, e.g. The `static` keyword makes the method accessible without an instance of `Program`. Otherwise, the program would require an instance, but any instance would require a program. To avoid that irresolvable circular dependency, C# compilers processing console applications like that above report an error, if there is no static `Main` method. The `void` keyword declares that `Main` has no return value. `Console` is a static class in the `System` namespace. It provides an interface to the standard input, output, and error streams for console applications. The program calls the `Console` method `WriteLine`, which displays on the console a line with the argument, the string "Hello, world! `WriteLine`" is almost the same argument!

Drawing library An example of what it can do: This process usually takes 6-9 months. The C# language definition and the CLI are standardized under ISO and Ecma standards that provide reasonable and non-discriminatory licensing protection from patent claims. Microsoft has agreed not to sue open source developers for violating patents in non-profit projects for the part of the framework that is covered by the OSP. The compiler, which is entirely written in managed code C#, has been opened up and functionality surfaced as APIs. It is thus enabling developers to create refactoring and diagnostics tools. NET class libraries up to. The Mono project provides an open-source C# compiler, a complete open-source implementation of the Common Language Infrastructure including the required framework libraries as they appear in the ECMA specification, and a nearly complete implementation of the Microsoft proprietary. As of Mono 2. Xamarin provides tools to develop cross-platform applications for common mobile and desktop operating systems, using C# as a codebase and compiling to native code. Mono is a common choice for game engines due to its cross-platform nature. The Unity game engine uses Mono C# as its primary scripting language.

Chapter 6 : Announcing the .NET Framework | .NET Blog

This section describes the programming essentials you need to create .NET applications, from creating assemblies from your code to securing your application. Many of the fundamentals covered in this section are used to create any application using the .NET Framework. This section provides conceptual.

Select the language for the download page. This option does not download the localized resources of the .NET Framework; it only affects the text displayed on the download page. If prompted, select the download that matches your system architecture, and then choose Next. When the download prompt appears do ONE of the following: If you want to install the .NET Framework on your computer, choose Run, and then follow the prompts on your screen. If you want to download the .NET Framework for redistribution, choose Save, and then follow the prompts on your screen. If you want to download resources for additional languages, follow the instructions in the next section to install one or more language packs. Note If you encounter any problems during the installation, see Troubleshooting. When you install these versions on a system that has the .NET Framework 4 installed, the assemblies are replaced. NET Framework 4 files. If you want to go back to the .NET Framework 4, you must reinstall it and any updates to it. You must have administrative credentials to install the .NET Framework. If you previously installed the .NET Framework, you must have administrative credentials to install the .NET Framework. For more information about this issue, see Microsoft Security Advisory and Knowledge Base article [To install language packs](#) Language packs are executable files that contain the localized resources such as translated error messages and UI text for supported languages. NET Framework error messages and other text are displayed in English. Note that the web installer automatically installs the language pack that matches your operating system, but you can download additional language packs to your computer. NET Framework components that are required to run an app, so you must run the web or offline installer before you install a language pack. If you have already installed a language pack, uninstall it, install the .NET Framework, and then reinstall the language pack. Open the language pack download page for the .NET Framework.

Chapter 7 : .NET Framework Development Guide | Microsoft Docs

NET Framework Programmer's Reference by Dan Rahmel, , available at Book Depository with free delivery worldwide.

He will show you a significantly improved experience around .NET Framework reference source. For a tutorial, checkout our Channel 9 video: This new browsing experience provides the following features: Go to Definition and Finding References. You can now browse through. This tool was developed for internal use but we found it to be so incredibly useful that we had to share it with you. The video above highlights some of the cool features of this site and how you can use it. For more details, checkout the help page. You can download the entire reference source as a zip file, which now also includes project and solution files, so that you can easily open and browse it in VS. Jump to Reference Source right from Visual Studio. If you have any more interesting ideas around how you can use leverage this site please touch base with us by using the feedback button on the site or leave a comment here in the post. NET framework sources for. This has been one of the highest voted items on User Voice: We are pleased to close this item as resolved. To read about how to setup VS to take advantage of this feature please go here. So what does it mean for your day-to-day activities? One day after a framework update you suddenly realize that the order of the controls in your app has suddenly changed. You fire up VS and follow the instructions described above and start stepping through you code like you normally would. You get to the point where you were doing your sorting: We find that this is an extremely useful diagnostic tool for our developers when they are investigating issues internally. This will give you the ability to better understand and diagnose issues between our and your code. NET framework update namely. However these builds would be rendered effectively useless the moment any update to the framework was released, since the binaries on the updated box no longer matched the PDBs that were indexed on the reference source server. Unfortunately the design of the system that we had in place was geared towards doing single and infrequent pushes of sources and symbols out and did not account for the sheer volume of builds and patches that come are produced out of the .NET framework build system. The summary of this is going forward the reference source debugging experience should just work. If it does not use the troubleshooting instructions at the link provided above and send us an email with the data requested, we will do our best to turn it around quickly. There is one caveat to this experiences; for security updates or updates that are otherwise deemed to have changes that we do not want leaked think security exploits you will still have a debugging experience, but rather than the file that corresponds to that PDB, you will get the last broadly shipped copy of that file. This could manifest itself in a slightly skewed debugging experience if you are stepping through a file where the fix was made. How does reference source relate to the Microsoft Symbol Server? However all PDBs that are present here do not have any source information in them, which makes them not very useful for stepping through sources. When you are trying to debug. Doing so could result in the symbols being loaded from the Microsoft Symbol Server and the source stepping experience would not work in that case. Ensure that the checkbox in front of Microsoft Symbol Server is unchecked. Our most immediate goal is to retire the current page at <http://> Please take a look at the old site and let us know of any concerns you have around deprecating it. Updating the indexed sources. The version of the framework that we currently have indexed is. Adding source for assemblies. As you can probably notice, the set of assemblies that we have is not complete. Summary Today we announced a new browsing experience for the. We would love to hear your feedback. Please let us know what you think about the new browsing experience by leaving a comment on this blog or by emailing us.

Chapter 8 : Programming Microsoft .NET (core Reference) - Jeff Prosise - Google Books

Note: Citations are based on reference standards. However, formatting rules can vary widely between applications and fields of interest or study. The specific requirements or preferences of your reviewing publisher, classroom teacher, institution or organization should be applied.

By implementing the core aspects of .NET Framework within the scope of CLI, these functions will not be tied to one language but will be available across the many languages supported by the framework. Common Language Runtime[edit] Main article: It serves as the execution engine of .NET Framework and offers many services such as memory management , type safety , exception handling , garbage collection , security and thread management. All programs written for .NET Framework consist of one or more assemblies, each of which must contain a manifest bearing the metadata for the assembly. The complete name of an assembly not to be confused with the file name on disk contains its simple text name, version number, culture, and public key token. Assemblies are considered equivalent if they share the same complete name. A private key can also be used by the creator of the assembly for strong naming. The public key token identifies which private key an assembly is signed with. Only the creator of the key pair typically the person signing the assembly can sign assemblies that have the same strong name as a prior version assembly, since the creator possesses the private key. Strong naming is required to add assemblies to Global Assembly Cache. Starting with Visual Studio .NET Native compilation technology allows for the compilation of .NET Framework includes a set of standard class libraries. The class library is organized in a hierarchy of namespaces. Most of the built-in application programming interfaces APIs are part of either System. These class libraries implement many common functions, such as file reading and writing, graphic rendering, database interaction, and XML document manipulation. The class libraries are available for all CLI compliant languages. The class library is divided into two parts with no clear boundary: BCL classes are available in .NET Framework as well as its alternative implementations including .NET Core and Mono. With the introduction of alternative implementations e.g. .NET Core, .NET platforms are encouraged to implement a version of the standard library allowing them to re-use extant third-party libraries to run without new versions of them. NET Standard Library allows an independent evolution of the library and app model layers within the .NET ecosystem. It is used to retrieve third-party libraries into a .NET project with a global library feed at NuGet. App models[edit] Atop the class libraries, multiple app models are used to create apps. NET Core apps by default. Other app models are offered by alternative implementations of the .NET Core are available on. The retroactive architectural definition of app models showed up in early .NET and was also applied to prior technologies like Windows Forms or WPF. NET decompilers such as .NET Reflector reveal only the managed code. Interoperability[edit] Because computer systems commonly require interaction between newer and older applications,. NET Framework provides means to access functions implemented in newer and older programs that execute outside. EnterpriseServices namespaces of the framework. Language independence[edit]. Because of this feature,. NET Framework supports the exchange of types and object instances between libraries and applications written using any conforming .NET Framework also enforce type safety. This prevents ill-defined casts, wrong method invocations, and memory size issues when accessing an object. This also makes most CLI languages statically typed with or without type inference. This makes it possible for third parties to create compatible implementations of the framework and its languages on other platforms. NET Framework has its own security mechanism with two general features: CAS is based on evidence that is associated with a specific assembly. Typically the evidence is the source of the assembly whether it is installed on the local machine or has been downloaded from the Internet. CAS uses evidence to determine the permissions granted to the code. Other code can demand that calling code be granted a specified permission. The demand causes CLR to perform a call stack walk: Managed CIL bytecode is easier to reverse-engineer than native code, unless obfuscated. NET decompiler programs enable developers with no reverse-engineering skills to view the source code behind unobfuscated. In contrast, apps compiled to native machine code are much harder to reverse-engineer, and source code is almost never produced successfully, mainly because of compiler

optimizations and lack of reflection. Labs , Turbo , and Red Gate Software. Method-level encryption tools for .NET code are available from vendors such as SafeNet. Memory management[edit] CLR frees the developer from the burden of managing memory allocating and freeing up when done ; it handles memory management itself by detecting when memory can be safely freed. As long as a reference to an object exists, which may be either direct, or via a graph of objects, the object is considered to be in use. When no reference to an object exists, and it cannot be reached or used, it becomes garbage, eligible for collection. It is a non-deterministic, compacting, mark-and-sweep garbage collector. GC runs only when a set amount of memory has been used or there is enough pressure for memory on the system. Since it is not guaranteed when the conditions to reclaim memory are reached, GC runs are non-deterministic. .NET application has a set of roots, which are pointers to objects on the managed heap managed objects. These include references to static objects and objects defined as local variables or method parameters currently in scope, and objects referred to by CPU registers. It uses CLI metadata and reflection to discover the objects encapsulated by an object, and then recursively walk them. It then enumerates all the objects on the heap which were initially allocated contiguously using reflection. All objects not marked as reachable are garbage. However, this leaves chunks of free space between objects which were initially contiguous. The objects are then compacted together to make free space on the managed heap contiguous again. The latest version of .NET framework uses concurrent garbage collection along with user code, making pauses unnoticeable, because it is done in the background. .NET Framework is also generational. Newly created objects are tagged Generation 0. Objects that survive one garbage collection are tagged Generation 1. Generation 1 objects that survive another collection are Generation 2. The framework uses up to Generation 2 objects. This raises the efficiency of garbage collection, as older objects tend to have longer lifetimes than newer objects. .NET Framework compiles the CIL code into executable code using its just-in-time compiler , and caches the executable program into the. To speed up the first launch, developers may use the Native Image Generator utility to manually ahead-of-time compile and cache any. Simd namespace in In case the CPU lacks support for those extensions, the instructions are simulated in software. .NET Framework is the predominant implementation of. Other implementations for parts of the framework exist. Although the runtime engine is described by an ECMA-ISO specification, other implementations of it may be encumbered by patent issues; ISO standards may include the disclaimer, "Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO shall not be held responsible for identifying any or all such patent rights. Also, parts of FCL have Windows-specific functions and behavior, so implementation on non-Windows platforms can be problematic. Some alternative implementations of parts of the framework are listed here. .NET Micro Framework is a .NET platform for extremely resource-constrained devices. .NET , [68] albeit with an amount of hacking, and with limited functionalities and debugging in an emulator or on hardware , both using Microsoft Visual Studio. It also features a subset of. .NET Core is an alternative Microsoft implementation of the managed code framework; it has similarities with. It is cross-platform and free and open-source. It is dual-licensed as free and proprietary software. It includes support for ASP. NET, and Windows Forms libraries for a wide range of architectures and operating systems.

Chapter 9 : Install the .NET Framework developer pack or redistributable | Microsoft Docs

.NET Framework (pronounced dot net) is a software framework developed by Microsoft that runs primarily on Microsoft. racedaydvl.com includes a large class library named Framework Class Library (FCL) and provides language interoperability (each language can use code written in other languages) across several programming languages.