[[CodeX](https://medium.com/codex?source=post_page-----ad721d4b0cde--------------------------------)](https://medium.com/codex?source=post_page-----ad721d4b0cde--------------------------------)

Published in

[CodeX](https://medium.com/codex?source=post_page-----ad721d4b0cde--------------------------------)

You have **1** free member-only story left this month.

[Sign up for Medium and get an extra one](https://medium.com/m/signin?operation=register&redirect=https%3A%2F%2Fmedium.com%2Fcodex%2Fasp-net-core-6-logging-ad721d4b0cde&source=-----ad721d4b0cde---------------------metered_view_2-----------)

[[](https://medium.com/@justin.muench?source=post_page-----ad721d4b0cde--------------------------------)](https://medium.com/@justin.muench?source=post_page-----ad721d4b0cde--------------------------------)

[Justin Muench](https://medium.com/@justin.muench?source=post_page-----ad721d4b0cde--------------------------------)

Dec 14, 2022

·

14 min read

·

Member-only

·

Listen

**NET 6 is the fastest full stack web framework, which lowers compute costs if you're running in the cloud. Ultimate productivity: . NET 6 and Visual Studio 2022 provide hot reload, new git tooling, intelligent code editing, robust diagnostics and testing tools, and better team collaboration.Nov 8, 2022**

**.NET Core 6 — Logging and Exception Handlin****g**

Photo by [JESHOOTS.COM](https://unsplash.com/@jeshoots?utm_source=medium&utm_medium=referral) on [Unsplash](https://unsplash.com/?utm_source=medium&utm_medium=referral)

Logging is a critical part of any application development process, and ASP.NET Core 6 makes it easier than ever to implement logging functionality in your web apps.

In this blog post, we'll take an in-depth look at the built-in logging features of ASP.NET Core 6 and various third-party tools you can use for enhanced logging capabilities.

Whether you're just getting started with developing in ASP.NET Core or have been coding for years, the tips included here will help optimize your production workflow from start to finish!

**Logging vs. Monitoring**

***ASP.NET 6.0 provides a range of features and tools for logging and monitoring applications. At its core, logging involves capturing detailed information about the events and processes in an application, such as user actions, system events, and performance metrics. This information is typically stored in a log file or database. It can be analyzed to identify and diagnose problems, optimize performance, and gain insights into the behavior and usage of the software.***

On the other hand, monitoring involves observing and measuring an application's performance and behavior in real time and providing alerts or notifications when certain conditions are met. For example, a monitoring system might be configured to send an alert if the response time of an application exceeds a certain threshold or if an error occurs. Monitoring can help identify and resolve issues before they affect users and provide valuable information for performance tuning and optimization.

Some of the popular libraries and tools for logging in ASP.NET 6.0 include Microsoft.Extensions.Logging, NLog, and Serilog.

**Why do we need logging?**

There are several reasons why logging is essential in ASP.NET 6.0 and other software development environments. Here are a few reasons:

1. Logging helps to identify and diagnose problems in the software. By capturing detailed information about the events and processes within the software, developers can use logs to identify the root causes of errors and other issues. This can reduce the time and effort required to fix bugs and improve the overall reliability of the software.
2. Logging can provide valuable insights into the performance and usage of the software. By analyzing log data, developers can better understand how the software is used, the most popular features, and where performance bottlenecks may occur. This information can be used to optimize the software and improve the user experience.
3. Logging can help to meet compliance and regulatory requirements. Some industries have legal requirements for logging specific types of information, such as user activity or financial transactions. By implementing a robust logging system, software developers can ensure that they meet these requirements and avoid potential legal penalties.
4. Logging can provide a valuable data source for machine learning and other data-driven processes. By capturing detailed information about the events and processes within the software, developers can use this data to train machine learning algorithms and improve the performance of the software over time.

**Prerequisites**

* [.NET 6.0 SDK](https://dotnet.microsoft.com/download/dotnet/6.0)
* [Visual Studio 2019](https://visualstudio.microsoft.com/downloads/) (or a compatible IDE)

**Project Structure**

I have written a sample .NET 6.0 Customer Service.

The solution includes:

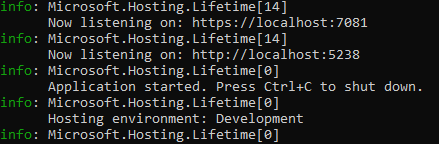
* LoggingAndMonitoringAPIExample: This is the main ASP.NET 6.0 web application, which demonstrates how to perform logging and monitoring in a web application.
* LoggingAndMonitoringAPIExample.Logic: This project contains the logic for the logging and monitoring functionality.
* LoggingAndMonitoringAPIExample.Tests: This project includes unit tests for the LoggingAndMonitoringAPIExample and LoggingAndMonitoringAPIExample.Logic projects, which test the logging and monitoring functionality.

Based on this solution, logging and monitoring will be examined and explained in more detail.

**Basic — Logging**

I want to start with the basic logging, which is included.

As soon as you start a .NET Core web application, you will notice that some info output already appears in the associated debug console. These log entries from the application tell us that the application is running and on which port and under which hosting environment it can be reached.



Generally, an ILogger<T> instance is injected into the class where the logging should occur.

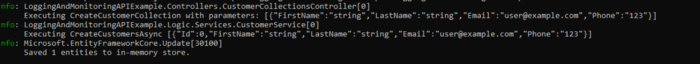
[ApiController]  
 [Route("api/[controller]")]  
 public class CustomerCollectionsController : ControllerBase   
 {  
 private readonly ICustomerService \_customerService;  
 private readonly IMapper \_mapper;  
 private readonly ILogger<CustomerCollectionsController> \_logger;  
  
  
 public CustomerCollectionsController(ICustomerService customerService, IMapper mapper, ILogger<CustomerCollectionsController> logger)  
 {  
 \_customerService = customerService;  
 \_mapper = mapper;  
 \_logger = logger;  
 }  
  
 [HttpPost]  
 [Produces("application/json", "application/xml", Type = typeof(IEnumerable<CustomerDto>))]  
 public async Task<ActionResult<IEnumerable<CustomerDto>>> CreateCustomerCollection([FromBody] IEnumerable<CustomerForCreationDto> customerCollection)  
 {  
 \_logger.LogInformation("Executing {Action} with parameters: {Parameters}", nameof(CreateCustomerCollection), JsonSerializer.Serialize(customerCollection));  
  
 var customerEntities = \_mapper.Map<IEnumerable<Customer>>(customerCollection);  
 var result = await \_customerService.CreateCustomersAsync(customerEntities);  
  
 var customerCollectionToReturn = \_mapper.Map<IEnumerable<CustomerDto>>(result);  
 var idsAsString = string.Join(",", customerCollectionToReturn.Select(x => x.Id));  
  
 return CreatedAtRoute("GetCustomers", new { customerIds = idsAsString }, customerCollectionToReturn);  
 }  
 }

It is considered best practice to create a private read-only field initialized in the constructor via DI. In addition, no more than 2–3 classes should be injected, according to Martin C. Robert.

The logger has different log methods for different logging levels; more about that later. Within the CreateCustomerCollection method, the [LogInformation](https://learn.microsoft.com/en-us/dotnet/api/microsoft.extensions.logging.loggerextensions.loginformation?view=dotnet-plat-ext-7.0#microsoft-extensions-logging-loggerextensions-loginformation(microsoft-extensions-logging-ilogger-system-string-system-object())) method is called. A message and the desired objects logged are passed to this method as parameters.

However, it should be noted that it is not intended to use string interpolation, even if it looks like it. The reason follows later, but the desired objects should always be passed as parameters to the method.

You should now find the log entry in the console.



Later in the post, we'll look at best practices and different locations/types for logging.

**Do's and Dont's**

* Ensure to log important events, such as when a user logs in or out, when an error occurs, or when a critical operation is performed. These events can help you troubleshoot issues and monitor the health of your application.
* Include relevant information in your logs, such as timestamps, user IDs, and details about the events logged. This can make it easier to understand and analyze your logs.
* Do use a logging framework or library to handle your application logs. This can make configuring and managing your logs easier and provide features such as log rotation and centralized storage.
* Don't log sensitive information, such as passwords or personally identifiable information (PII). These types of logs can be a security risk if unauthorized individuals access them.
* Don't use logs as a replacement for proper error handling. Your application should still have try/catch blocks and other error-handling mechanisms in place, and your logs should complement these mechanisms, not replace them.
* Store your logs in a different location than your application code. Instead, store your logs in another place, such as a centralized log server. This can make it easier to access and analyze your logs and can help prevent logs from being lost or deleted if your application code is modified or deleted.

**Log Levels and Filters**

For the logs to be helpful, you must ensure they are neat. For this purpose, filters and logging levels can be used.

**Log Levels**

**Trace**— Lowest Level and is used for low-level diagnostics to solve complex issues [1]. This should never be activated in production because it can contain sensitive data, and there will be a lot of messages [1].

**Debug** — More standard level to gather low-level diagnostic and debug data [1]. The volume of entries is also high here, so you should be careful if you apply it in a production environment [1].

**Information**—Is used for more general messages [1].

**Warning**— Is used for abnormal or unexpected events but generally without a failure in the application[1].

**Error** — While error caused a failure of the app but may be user or situation-specific [1].

**Critical** — Errors not specific to the user or situation, like out-of-disk space or crashed database [1]. The whole application is down [1].

**Log-Levels in Use**

Now let's take a look at the different log levels in use. We want to take a closer look at the Get Endpoint in the Customer Controller.

[HttpGet("{customerId}", Name = "GetCustomer")]  
 [Produces("application/json", "application/xml", Type = typeof(CustomerDto))]  
 public async Task<ActionResult<CustomerDto>> GetCustomer(int customerId)  
 {  
 \_logger.LogInformation("Executing {Action} with Id {customerId}", nameof(GetCustomer), customerId);  
   
 var customer = await \_customerService.GetCustomerAsync(customerId);  
  
 if (customer != null)  
 {  
 var result = \_mapper.Map<CustomerDto>(customer);  
  
 return Ok(result);  
 }  
  
 \_logger.LogWarning("No customer found with Id {customerId}", customerId);  
  
 return NotFound();  
 }

First, we log information on the request, including the passed CustomerId. If we do not find a customer, we log a warning message.

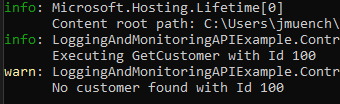
public async Task<Customer?> GetCustomerAsync(int id)  
 {  
 \_logger.LogDebug("Executing {Action} {Parameters}", nameof(GetCustomerAsync),  
 id);  
  
 var result = await \_customerContext.Customers.FirstOrDefaultAsync(x => x.Id == id);  
 return result;  
 }

In the service class, I log here in the debug level.

For example, if we have a synchronous method, we could log the load times for querying.

public Customer? GetCustomer(int id)  
{  
 var timer = new Stopwatch();  
 timer.Start();  
 var result = \_customerContext.Customers.FirstOrDefault(x => x.Id == id);  
 timer.Stop();  
  
 \_logger.LogDebug("Query took {Time} ms for {Action} and Id {Id}", timer.ElapsedMilliseconds,  
 nameof(GetCustomer), id);  
  
 return result;  
}

If we run the whole thing, we will notice that we get the warnings and information logs displayed but not the debug messages. This is because they are hidden by default.



Besides the log levels, there is also the*log category.* In the case of passing [ILogger<T>](https://learn.microsoft.com/en-us/dotnet/api/microsoft.extensions.logging.ilogger?view=dotnet-plat-ext-7.0&viewFallbackFrom=net-6.0), the type corresponds to the class's namespace. Alternatively, we can select the category by injecting [ILoggerFactory](https://learn.microsoft.com/en-us/dotnet/api/microsoft.extensions.logging.iloggerfactory?view=dotnet-plat-ext-7.0&viewFallbackFrom=net-6.0) instead of ILogger<T>.

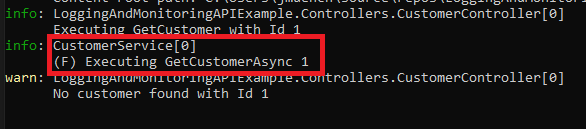
private readonly ILogger \_factoryLogger;  
private readonly ILogger<CustomerService> \_logger;  
  
public CustomerService(CustomerDbContext customerContext, ILoggerFactory loggerFactory, ILogger<CustomerService> logger)  
{  
 \_customerContext = customerContext;  
   
 \_factoryLogger = loggerFactory.CreateLogger("CustomerServiceCategory");  
   
 \_logger = logger;  
}

For demonstration purposes, I have now injected both the ILogger and the IFactoryLogger.

I updated the GetCustomerAsync Method and added the \_factoryLogger as well.

public async Task<Customer?> GetCustomerAsync(int id)  
{  
 \_logger.LogDebug("Executing {Action} {Parameters}", nameof(GetCustomerAsync),  
 id);  
   
 \_factoryLogger.LogInformation("(F) Executing {Action} {Parameters}", nameof(GetCustomerAsync),  
 id);  
  
 var result = await \_customerContext.Customers.FirstOrDefaultAsync(x => x.Id == id);  
 return result;  
}

As you can see below, the log category is shown as well.



**Log Filters**

Log filters help us to control what is written into our logs [1].

They are set up in three parts [1]:

Provider + Category + Minimum Level [1]

The Provider is where are logs are written. The category is shown in the example above, and the minimum level for the combination [1].

Log filters are the essential mechanism for controlling noise in the logs [1]. In high-traffic environments, it is necessary to be careful because they can produce a lot of messages [1].

It is best practice to specify the filters in the configuration (appsettings, environment variables, command line, etc.), but it is also possible to set it in the code [1]. The benefit of using the configuration is that you can change it during the run time [1].

Let's first show the missing debug log messages. For this, we open appsettings.json in the root directory.

{  
 "Logging": {  
 "LogLevel": {  
 "Default": "Information",  
 "Microsoft.AspNetCore": "Warning"  
 }  
 },  
 "AllowedHosts": "\*"  
}

It sets the minimum log level to information. This is why we do not see any debug entries [1].

We should also see the debug entries if we change the default level to debug and rerun our code.

https://miro.medium.com/max/631/1*KsxDfMz9KmdZAA1M13wGmA.png

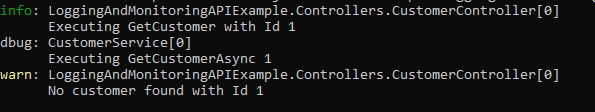
*(If you don't see them, pick the right appsettings for the environment. If the running environment is set up to development in the launch settings, then you have to change the appsettings.Development.json)*

If we want to set the LogLevel differently for a category, we can list the specific category in the appsettings.

{  
 "Logging": {  
 "LogLevel": {  
 "Default": "Information",  
 "CustomerService": "Debug",  
 "Microsoft.AspNetCore": "Warning"  
 }  
 }  
}

public CustomerService(CustomerDbContext customerContext, ILoggerFactory loggerFactory)  
{  
 \_customerContext = customerContext;  
 \_logger = loggerFactory.CreateLogger("CustomerService");  
}

In this case, we only see the debug entries from our CustomerService logger, not all other debug entries.



Alternatively, the log level can be set in the environment variables in the launchsettings.

"profiles": {  
"LoggingAndMonitoringAPIExample": {  
 "commandName": "Project",  
 "dotnetRunMessages": true,  
 "launchBrowser": true,  
 "launchUrl": "swagger",  
 "applicationUrl": "https://localhost:7081;http://localhost:5238",  
 "environmentVariables": {  
 "ASPNETCORE\_ENVIRONMENT": "Development",  
 //Specify log level... below  
 "Logging\_\_LogLevel\_\_CustomerService": "Debug"   
 }  
}

**Providers**

Next, let's look at the providers. These are the destination where the log entries are written [1].

Let's assume we want two different providers and logging different levels.

In this case, we can wrap the Log Level in the app settings with the console like below:

{  
 "Logging": {  
 "Console": {  
 "LogLevel": {  
 "Default": "Information",  
 "CustomerService": "Debug",  
 "Microsoft.AspNetCore": "Warning"  
 }  
 }  
 }  
}

A second provider can be Debug, using the system.Diagnostics.Debug, which writes to TraceListeners [1].

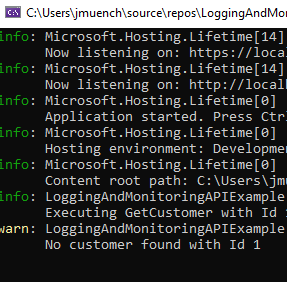
{   
 "Logging": {  
 "Console": {  
 "LogLevel": {  
 "Default": "Information",  
 "CustomerService": "Information",  
 "Microsoft.AspNetCore": "Warning"  
 }  
 },  
 "Debug": {  
 "LogLevel": {  
 "Default": "Information",  
 "CustomerService": "Debug",  
 "Microsoft.AspNetCore": "Warning"  
 }  
 }  
 }  
}

We added a second section in our appsettings wrapped in "Debug."

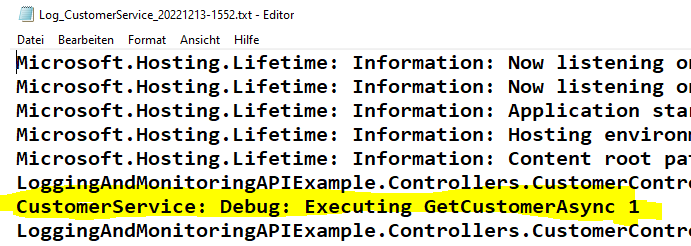
To see the debug messages a little bit clearer, we added in our Programm.cs [1]:

var path = Environment.GetFolderPath(Environment.SpecialFolder.LocalApplicationData);  
var tracePath = Path.Join(path, $"Log\_CustomerService\_{DateTime.Now.ToString("yyyyMMdd-HHmm")}.txt");  
Trace.Listeners.Add(new TextWriterTraceListener(tracePath));  
Trace.AutoFlush = true;

Now we can call the GetCustomer Method and look at the file and console.



The file should be stored in your local AppData directory.



As you can see, we do log debug entries into the file, but they are not shown in the console.

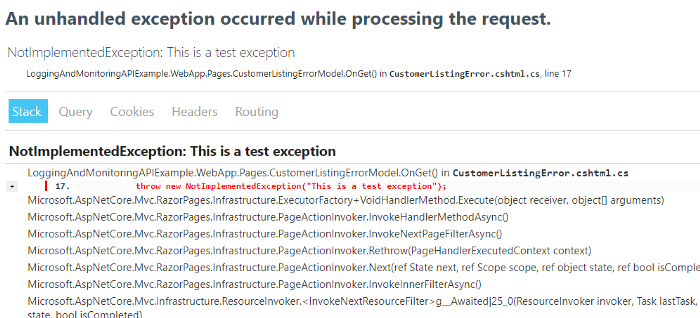
**Exception Handling and Logging**

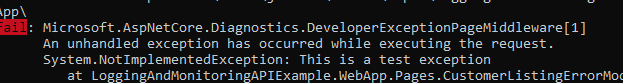
To demonstrate error logging, I added a WebApp to the project that will eventually call the CustomerCollections.

In addition, I also added an endpoint in the API for throwing an exception and another on a new page like the following:

private readonly HttpClient \_apiClient;  
 public CustomerListingErrorModel(HttpClient apiClient)  
 {  
 \_apiClient = apiClient;  
 \_apiClient.BaseAddress = new Uri("https://localhost:7081/");  
 }  
  
 public void OnGet()  
 {  
 throw new NotImplementedException("This is a test exception");  
 }

That page throws an error when called. However, this is not an excellent way — to display this error, as seen in the picture below.





This is a very trivial example, but distributed system errors are common, not necessarily because of application errors but because of the many components that can go wrong. To find these errors, decent error handling must be built in.

**Principles**

Exception handling is a critical principle to consider when developing an application [1]. The primary goal should be to always provide an elegant user experience by shielding users from errors and preventing them from seeing details that could compromise the security of their data or the system [1].

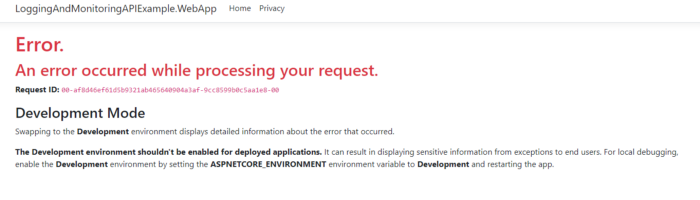
Applying global exception handling within the application architecture helps to achieve this, as well as leveraging try/catch blocks when specific errors need to be addressed individually [1].

Additionally, receiving and properly managing generated error IDs for easy referencing at a later time is imperative [1].

Meanwhile, knowing how to rely on logs during local development can help ensure that any errors are caught and logged before they can cause further harm down the line [1].

Try/catch blocks should only be used when adding value with them [1]. For example, if you need to add additional data to an exception when you make a Linq SQL request or so [1]. Alternatively, you can use try/catch if you want to continue processing in an exceptional case [1].

I have now enabled redirection to the error page in the web app for the Dev environment, so we get a better error page than the one above.



https://miro.medium.com/max/604/1*KCm0xLlF_FzhavVq4ZDZ4w.png

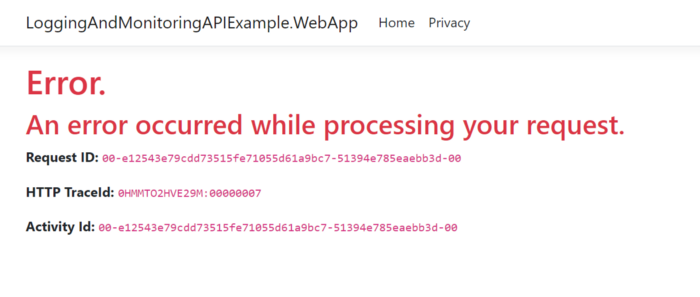
But we want a little more information and do not need the development mode message.

I added the Activity and TraceId to the ErrorModel.

public class ErrorModel : PageModel  
{  
 public string? RequestId { get; set; }  
  
 public bool ShowRequestId => !string.IsNullOrEmpty(RequestId);  
  
 public string TraceId { get; set; }  
  
 public Activity? CurrentActiviy { get; set; }  
  
 private readonly ILogger<ErrorModel> \_logger;  
  
 public ErrorModel(ILogger<ErrorModel> logger)  
 {  
 \_logger = logger;  
 }  
  
 public void OnGet()  
 {  
 RequestId = Activity.Current?.Id ?? HttpContext.TraceIdentifier;  
 TraceId = HttpContext.TraceIdentifier ?? string.Empty;  
 CurrentActiviy = Activity.Current;  
 }

Furthermore, I made some changes to the Error Page because most of the information shown on the error page is irrelevant to the user.

@page  
@model ErrorModel  
@{  
 ViewData["Title"] = "Error";  
}  
  
<h1 class="text-danger">Error.</h1>  
<h2 class="text-danger">An error occurred while processing your request.</h2>  
  
@if (Model.ShowRequestId)  
{  
 <p>  
 <strong>Request ID:</strong> <code>@Model.RequestId</code>  
 </p>  
 <p>  
 <strong>HTTP TraceId:</strong> <code>@Model.TraceId</code>  
 </p>  
 <p>  
 <strong>Activity Id:</strong> <code>@Model.CurrentActiviy?.Id</code>  
 </p>  
}

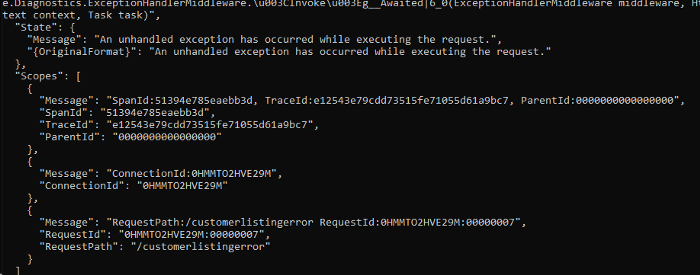


But we, as developers, need information like Ids to find the error on the logs.

The exception shown in the console is hard to read, so we added some formatting to the [appsettings](https://learn.microsoft.com/en-us/dotnet/core/extensions/console-log-formatter#set-formatter-with-configuration), making it much easier to see what happened.

{  
 "Logging": {  
 "LogLevel": {  
 "Default": "Information",  
 "Microsoft.AspNetCore": "Warning"  
 },  
 "Console": {  
 "FormatterName": "json",  
 "FormatterOptions": {  
 "SingleLine": true,  
 "IncludeScopes": true,  
 "TimestampFormat": "HH:mm:ss ",  
 "UseUtcTimestamp": true,  
 "JsonWriterOptions": {  
 "Indented": true  
 }  
 }  
 }  
 },  
 "AllowedHosts": "\*"  
}

And as you can see in the console, the error will be formatted.



So we looked at how to make the bugs look more likable in the UI. However, there is no error page when we call an API.

But again, we only want to provide a little information but ensure that the client gets the received errors in a format it can handle. For this, there is a [convention](https://www.rfc-editor.org/rfc/rfc7807). In .NET 7.0, there is already a [middleware](https://learn.microsoft.com/en-us/aspnet/core/fundamentals/error-handling?view=aspnetcore-7.0#pds7) for this.

Unfortunately, in .NET 6.0, not. But there is a NuGet Package called [Hellang.Middleware.ProblemDetails](https://www.nuget.org/packages/Hellang.Middleware.ProblemDetails/), which will help us [1].

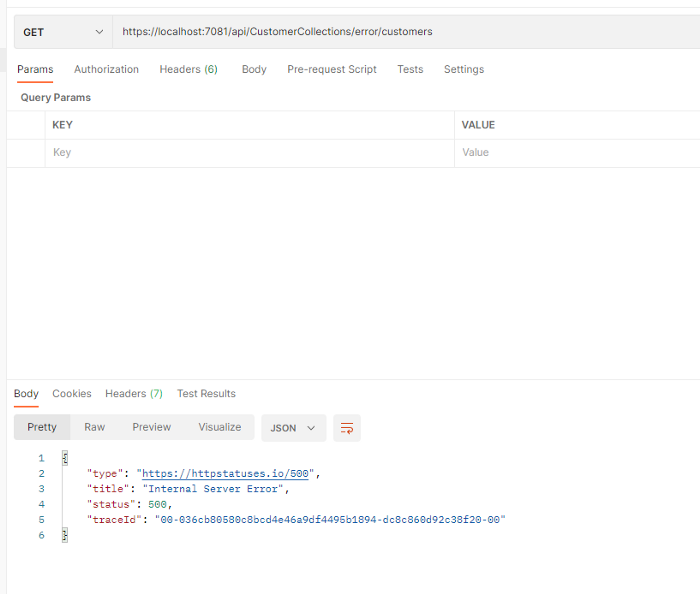
Adding this to the API Project:

builder.Services.AddProblemDetails();

and

app.UseProblemDetails();

Let's change to the production environment and rerun the API endpoint.



As you can see, the error message is now formatted in a way the client excepts it because of RFC7807. But there is one more thing we should change. The response can contain exception details by default [1].

To change this, we can modify the AddProblemDetails Method in the program.cs [1].

builder.Services.AddProblemDetails(opts =>  
{  
 opts.IncludeExceptionDetails = (ctx, ex) => false;  
 opts.OnBeforeWriteDetails = (ctx, dtls) =>  
 {  
 if (dtls.Status == 500)  
 dtls.Detail = "An error occured. Use Trace Id when contacting us.";  
 };  
});

Now we get uniform error responses, which is why we can customize the WebApp to handle the errors properly.

I injected a logger into my CustomerListingModel class.

public class CustomerListingModel : PageModel  
{  
 private readonly HttpClient \_apiClient;  
 private readonly ILogger<CustomerListingModel> \_logger;  
 public IEnumerable<CustomerDto> Customers { get; set; }  
 public CustomerListingModel(HttpClient apiClient, ILoggerFactory loggerFactory)  
 {  
 \_apiClient = apiClient;  
 \_apiClient.BaseAddress = new Uri("https://localhost:7081/");  
 \_logger = loggerFactory.CreateLogger<CustomerListingModel>();  
 }  
  
 public async Task OnGetAsync()  
 {  
 var response = await \_apiClient.GetAsync("api/CustomerCollections/error/customers");  
  
 if (!response.IsSuccessStatusCode)  
 {  
 var details = await response.Content.ReadFromJsonAsync<ProblemDetails>() ?? new ProblemDetails();  
  
 \_logger.LogWarning("API failure: {adress} Response {response}, Trace {traceId}",  
 $"{\_apiClient.BaseAddress}api/CustomerCollections", response.StatusCode, details.Extensions["traceId"]?.ToString());  
 throw new Exception("Api call failed");  
 }  
  
 Customers = await response.Content.ReadFromJsonAsync<List<CustomerDto>>() ?? new List<CustomerDto>();  
 }  
}

Through this logging and the unified response via ProblemDetails, we can deserialize the response and get the TraceId. We can log this away and show it to the user at the same time.

This is more text than I expected, so I will probably write a second part here.

**Conclusion**

In conclusion, logging is essential to any application built with ASP.NET Core 6 or other modern technologies like NodeJS or JavaEE 8+.

Correctly implemented logs can provide invaluable insights into user behavior and usage patterns and serve as early warning signs for potential issues that could arise later on down the line — making them valuable tools in any developer's toolbox!

With this guide in hand, you should now have a better understanding of how to get started with logging in to ASP.NET Core 6 so that you can begin taking advantage of this powerful feature today!

Want to connect?  
You can contact me via my social media channels ([Twitter](https://twitter.com/muench_justin), [LinkedIn](https://www.linkedin.com/in/justin-m%C3%BCnch-0b1087133/)).

**References**

1. [Logging and Monitoring in ASP.NET Core 6 by Erik Dahl](https://app.pluralsight.com/library/courses/logging-monitoring-aspdotnet-core-6/table-of-contents)
2. <https://learn.microsoft.com/en-us/aspnet/core/fundamentals/logging/?view=aspnetcore-6.0>

**Repo**

**[GitHub - muench-develops/LoggingAndMonitoringAPIExample: Test Project API Design](https://github.com/muench-develops/LoggingAndMonitoringAPIExample" \t "_blank)**

[To get started with the project, you will need to have the following prerequisites installed: Once you have the…](https://github.com/muench-develops/LoggingAndMonitoringAPIExample" \t "_blank)

[github.com](https://github.com/muench-develops/LoggingAndMonitoringAPIExample" \t "_blank)

[Aspnetcore](https://medium.com/tag/aspnetcore?source=post_page-----ad721d4b0cde---------------aspnetcore-----------------)

[Programming](https://medium.com/tag/programming?source=post_page-----ad721d4b0cde---------------programming-----------------)

[Engineering](https://medium.com/tag/engineering?source=post_page-----ad721d4b0cde---------------engineering-----------------)

[Logging](https://medium.com/tag/logging?source=post_page-----ad721d4b0cde---------------logging-----------------)

[Exception Handling](https://medium.com/tag/exception-handling?source=post_page-----ad721d4b0cde---------------exception_handling-----------------)

80

Enjoy the read? Reward the writer.Beta

Your tip will go to Justin Muench through a third-party platform of their choice, letting them know you appreciate their story.

Give a tip

Sign up for CrunchX

By CodeX

A weekly newsletter on what's going on around the tech and programming space [Take a look.](https://medium.com/codex/newsletters/crunchx?source=newsletter_v3_promo--------------------------newsletter_v3_promo-----------)



Get this newsletter

By signing up, you will create a Medium account if you don’t already have one. Review our [Privacy Policy](https://policy.medium.com/medium-privacy-policy-f03bf92035c9?source=newsletter_v3_promo--------------------------newsletter_v3_promo-----------) for more information about our privacy practices.

[More from CodeX](https://medium.com/codex?source=post_page-----ad721d4b0cde--------------------------------)

Follow

Everything connected with Tech & Code. Follow to join our 1M+ monthly readers

[[Brian Olson](https://medium.com/@devblabs?source=post_page-----ad721d4b0cde----0----------------------------)](https://medium.com/@devblabs?source=post_page-----ad721d4b0cde----0----------------------------)

[Brian Olson](https://medium.com/@devblabs?source=post_page-----ad721d4b0cde----0----------------------------)

[·Dec 14, 2022](https://medium.com/codex/your-first-question-when-writing-a-doc-bd3388bb8b49?source=post_page-----ad721d4b0cde----0----------------------------)

Member-only

**[Your First Question When Writing a Doc](https://medium.com/codex/your-first-question-when-writing-a-doc-bd3388bb8b49?source=post_page-----ad721d4b0cde----0----------------------------)**

[Disclaimer: All opinions are my own As a software engineer, I do a surprising amount of writing Design docs Blogs like this one (ha!) Feedback for performance reviews Documentation for products I work on Explanations of problems and solutions Chat messages for individuals and my team Emails (as little as…](https://medium.com/codex/your-first-question-when-writing-a-doc-bd3388bb8b49?source=post_page-----ad721d4b0cde----0----------------------------)

[Engineering](https://medium.com/tag/engineering?source=post_page-----ad721d4b0cde---------------engineering-----------------)

[3 min read](https://medium.com/codex/your-first-question-when-writing-a-doc-bd3388bb8b49?source=post_page-----ad721d4b0cde----0----------------------------)

[[](https://medium.com/codex/your-first-question-when-writing-a-doc-bd3388bb8b49?source=post_page-----ad721d4b0cde----0----------------------------)](https://medium.com/codex/your-first-question-when-writing-a-doc-bd3388bb8b49?source=post_page-----ad721d4b0cde----0----------------------------)

Share your ideas with millions of readers.

[Write on Medium](https://medium.com/new-story?source=post_page_footer_cta_write-------------------------------------)

[[Jeffrey Clos](https://medium.com/@jclos2679?source=post_page-----ad721d4b0cde----1----------------------------)](https://medium.com/@jclos2679?source=post_page-----ad721d4b0cde----1----------------------------)

[Jeffrey Clos](https://medium.com/@jclos2679?source=post_page-----ad721d4b0cde----1----------------------------)

[·Dec 14, 2022](https://medium.com/codex/the-geneinno-s2-underwater-scooter-making-light-of-ocean-lake-exploration-dd4c75cadd9?source=post_page-----ad721d4b0cde----1----------------------------)

Member-only

**[The Geneinno S2 Underwater Scooter, Making Light of Ocean/Lake Exploration](https://medium.com/codex/the-geneinno-s2-underwater-scooter-making-light-of-ocean-lake-exploration-dd4c75cadd9?source=post_page-----ad721d4b0cde----1----------------------------)**

[Small, Powerful, and Most of all, Fun! — Jeffrey Clos is a participant in the Amazon Associates LLC associates program, an affiliate advertising program designed to provide a means for sites to earn advertising fees by advertising and linking to Amazon.com. There are some links within this article to the described product. If you’re looking to explore the…](https://medium.com/codex/the-geneinno-s2-underwater-scooter-making-light-of-ocean-lake-exploration-dd4c75cadd9?source=post_page-----ad721d4b0cde----1----------------------------)

[Electric Vehicles](https://medium.com/tag/electric-vehicles?source=post_page-----ad721d4b0cde---------------electric_vehicles-----------------)

[3 min read](https://medium.com/codex/the-geneinno-s2-underwater-scooter-making-light-of-ocean-lake-exploration-dd4c75cadd9?source=post_page-----ad721d4b0cde----1----------------------------)

[[](https://medium.com/codex/the-geneinno-s2-underwater-scooter-making-light-of-ocean-lake-exploration-dd4c75cadd9?source=post_page-----ad721d4b0cde----1----------------------------)](https://medium.com/codex/the-geneinno-s2-underwater-scooter-making-light-of-ocean-lake-exploration-dd4c75cadd9?source=post_page-----ad721d4b0cde----1----------------------------)

[[Techstack](https://medium.com/@techstack-ltd?source=post_page-----ad721d4b0cde----2----------------------------)](https://medium.com/@techstack-ltd?source=post_page-----ad721d4b0cde----2----------------------------)

[Techstack](https://medium.com/@techstack-ltd?source=post_page-----ad721d4b0cde----2----------------------------)

[·Dec 14, 2022](https://medium.com/codex/how-to-craft-a-test-strategy-document-sample-test-strategy-inside-cffbdd291491?source=post_page-----ad721d4b0cde----2----------------------------)

**[How To Craft a Test Strategy Document [Sample Test Strategy Inside]](https://medium.com/codex/how-to-craft-a-test-strategy-document-sample-test-strategy-inside-cffbdd291491?source=post_page-----ad721d4b0cde----2----------------------------)**

[Without testing documentation, measuring QA quality remains challenging even for seasoned professionals. The need for more documentation is even more apparent when onboarding new QA engineers and scaling your product. With documentation, the value and purpose of quality assurance remain clear. When quality is measured correctly, you can predict test…](https://medium.com/codex/how-to-craft-a-test-strategy-document-sample-test-strategy-inside-cffbdd291491?source=post_page-----ad721d4b0cde----2----------------------------)

[Quality Assurance](https://medium.com/tag/quality-assurance?source=post_page-----ad721d4b0cde---------------quality_assurance-----------------)

[9 min read](https://medium.com/codex/how-to-craft-a-test-strategy-document-sample-test-strategy-inside-cffbdd291491?source=post_page-----ad721d4b0cde----2----------------------------)

[[](https://medium.com/codex/how-to-craft-a-test-strategy-document-sample-test-strategy-inside-cffbdd291491?source=post_page-----ad721d4b0cde----2----------------------------)](https://medium.com/codex/how-to-craft-a-test-strategy-document-sample-test-strategy-inside-cffbdd291491?source=post_page-----ad721d4b0cde----2----------------------------)

[[Lucky Brian](https://medium.com/@luckybriann?source=post_page-----ad721d4b0cde----3----------------------------)](https://medium.com/@luckybriann?source=post_page-----ad721d4b0cde----3----------------------------)

[Lucky Brian](https://medium.com/@luckybriann?source=post_page-----ad721d4b0cde----3----------------------------)

[·Dec 14, 2022](https://medium.com/codex/metaverse-development-company-in-usa-9b816204c1d8?source=post_page-----ad721d4b0cde----3----------------------------)

**[Insights On The Metaverse Technology And Its Future Potential](https://medium.com/codex/metaverse-development-company-in-usa-9b816204c1d8?source=post_page-----ad721d4b0cde----3----------------------------)**

[Insights On The Metaverse Technology And Its Future Potential — The Metaverse is a post-reality universe that combines the real world with digital virtual worlds. It is a place where many people can interact with each other all the time. The Metaverse is built on the merging of augmented reality (AR) and virtual reality (VR) technologies, which make it possible…](https://medium.com/codex/metaverse-development-company-in-usa-9b816204c1d8?source=post_page-----ad721d4b0cde----3----------------------------)

[Metaverse Development](https://medium.com/tag/metaverse-development?source=post_page-----ad721d4b0cde---------------metaverse_development-----------------)

[4 min read](https://medium.com/codex/metaverse-development-company-in-usa-9b816204c1d8?source=post_page-----ad721d4b0cde----3----------------------------)

[[](https://medium.com/codex/metaverse-development-company-in-usa-9b816204c1d8?source=post_page-----ad721d4b0cde----3----------------------------)](https://medium.com/codex/metaverse-development-company-in-usa-9b816204c1d8?source=post_page-----ad721d4b0cde----3----------------------------)

[[Dmitry Yarygin](https://medium.com/@nomadic-dmitry?source=post_page-----ad721d4b0cde----4----------------------------)](https://medium.com/@nomadic-dmitry?source=post_page-----ad721d4b0cde----4----------------------------)

[Dmitry Yarygin](https://medium.com/@nomadic-dmitry?source=post_page-----ad721d4b0cde----4----------------------------)

[·Dec 13, 2022](https://medium.com/codex/test-automation-in-your-browser-no-local-setup-needed-11bcbd64765c?source=post_page-----ad721d4b0cde----4----------------------------)

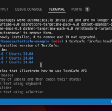
Member-only

**[Test Automation in your Browser. No local setup needed?](https://medium.com/codex/test-automation-in-your-browser-no-local-setup-needed-11bcbd64765c?source=post_page-----ad721d4b0cde----4----------------------------)**

[Test Automation is one of the essential tasks of the QA Engineer now. But most of our current work involves preparing the environment and setting up tests complexly. I was researching my options for running tests right in my browser without too much configuration effort. I immediately got the idea…](https://medium.com/codex/test-automation-in-your-browser-no-local-setup-needed-11bcbd64765c?source=post_page-----ad721d4b0cde----4----------------------------)

[Programming](https://medium.com/tag/programming?source=post_page-----ad721d4b0cde---------------programming-----------------)

[4 min read](https://medium.com/codex/test-automation-in-your-browser-no-local-setup-needed-11bcbd64765c?source=post_page-----ad721d4b0cde----4----------------------------)

[[](https://medium.com/codex/test-automation-in-your-browser-no-local-setup-needed-11bcbd64765c?source=post_page-----ad721d4b0cde----4----------------------------)](https://medium.com/codex/test-automation-in-your-browser-no-local-setup-needed-11bcbd64765c?source=post_page-----ad721d4b0cde----4----------------------------)

[Read more from CodeX](https://medium.com/codex?source=post_page-----ad721d4b0cde--------------------------------)

Recommended from Medium

[[Surender Vikram Singh](https://medium.com/@surendervikram?source=post_internal_links---------0----------------------------)](https://medium.com/@surendervikram?source=post_internal_links---------0----------------------------)

[Surender Vikram Singh](https://medium.com/@surendervikram?source=post_internal_links---------0----------------------------)

**[Top 7 Trends That Will Define Future of Web Development in 2019](https://medium.com/@surendervikram/top-7-trends-that-will-define-future-of-web-development-in-2019-757ec237fb34?source=post_internal_links---------0----------------------------)**

[[](https://medium.com/@surendervikram/top-7-trends-that-will-define-future-of-web-development-in-2019-757ec237fb34?source=post_internal_links---------0----------------------------)](https://medium.com/@surendervikram/top-7-trends-that-will-define-future-of-web-development-in-2019-757ec237fb34?source=post_internal_links---------0----------------------------)

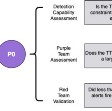
[[Joshua Prager](https://medium.com/@bouj33boy?source=post_internal_links---------1----------------------------)](https://medium.com/@bouj33boy?source=post_internal_links---------1----------------------------)

[Joshua Prager](https://medium.com/@bouj33boy?source=post_internal_links---------1----------------------------)

in

[Posts By SpecterOps Team Members](https://medium.com/specter-ops-posts?source=post_internal_links---------1----------------------------)

**[Prioritization of the Detection Engineering Backlog](https://medium.com/specter-ops-posts/prioritization-of-the-detection-engineering-backlog-dcb18a896981?source=post_internal_links---------1----------------------------)**

[[](https://medium.com/specter-ops-posts/prioritization-of-the-detection-engineering-backlog-dcb18a896981?source=post_internal_links---------1----------------------------)](https://medium.com/specter-ops-posts/prioritization-of-the-detection-engineering-backlog-dcb18a896981?source=post_internal_links---------1----------------------------)

[[HOOD Group](https://medium.com/@hoodgroup?source=post_internal_links---------2----------------------------)](https://medium.com/@hoodgroup?source=post_internal_links---------2----------------------------)

[HOOD Group](https://medium.com/@hoodgroup?source=post_internal_links---------2----------------------------)

**[Agile Project Planning (APP) — Part 2: Misunderstandings around RE Process, product backlog and…](https://medium.com/@hoodgroup/agile-project-planning-app-part-2-misunderstandings-around-re-process-product-backlog-and-b86cade7c4f0?source=post_internal_links---------2----------------------------)**

[[](https://medium.com/@hoodgroup/agile-project-planning-app-part-2-misunderstandings-around-re-process-product-backlog-and-b86cade7c4f0?source=post_internal_links---------2----------------------------)](https://medium.com/@hoodgroup/agile-project-planning-app-part-2-misunderstandings-around-re-process-product-backlog-and-b86cade7c4f0?source=post_internal_links---------2----------------------------)

[[Shreyash Somvanshi](https://medium.com/@ShreyashSomvanshi?source=post_internal_links---------3----------------------------)](https://medium.com/@ShreyashSomvanshi?source=post_internal_links---------3----------------------------)

[Shreyash Somvanshi](https://medium.com/@ShreyashSomvanshi?source=post_internal_links---------3----------------------------)

**[Installing or Uninstalling R on Ubuntu](https://medium.com/@ShreyashSomvanshi/installing-or-uninstalling-r-on-ubuntu-90bb6a997ac1?source=post_internal_links---------3----------------------------)**

[[](https://medium.com/@ShreyashSomvanshi/installing-or-uninstalling-r-on-ubuntu-90bb6a997ac1?source=post_internal_links---------3----------------------------)](https://medium.com/@ShreyashSomvanshi/installing-or-uninstalling-r-on-ubuntu-90bb6a997ac1?source=post_internal_links---------3----------------------------)

[[Pavel Rekun](https://medium.com/@menos?source=post_internal_links---------4----------------------------)](https://medium.com/@menos?source=post_internal_links---------4----------------------------)

[Pavel Rekun](https://medium.com/@menos?source=post_internal_links---------4----------------------------)

**[Castro Progress Report: May 2017](https://medium.com/@menos/castro-progress-report-may-2017-3b70077c4e57?source=post_internal_links---------4----------------------------)**

[[](https://medium.com/@menos/castro-progress-report-may-2017-3b70077c4e57?source=post_internal_links---------4----------------------------)](https://medium.com/@menos/castro-progress-report-may-2017-3b70077c4e57?source=post_internal_links---------4----------------------------)

[[Kintone](https://medium.com/@kintone?source=post_internal_links---------5----------------------------)](https://medium.com/@kintone?source=post_internal_links---------5----------------------------)

[Kintone](https://medium.com/@kintone?source=post_internal_links---------5----------------------------)

in

[Digital Transformation Talk](https://medium.com/digital-transformation-talk?source=post_internal_links---------5----------------------------)

**[No-code is the future of software development, but is it secure?](https://medium.com/digital-transformation-talk/no-code-is-the-future-of-software-development-but-is-it-secure-abe587cdc314?source=post_internal_links---------5----------------------------)**

[[](https://medium.com/digital-transformation-talk/no-code-is-the-future-of-software-development-but-is-it-secure-abe587cdc314?source=post_internal_links---------5----------------------------)](https://medium.com/digital-transformation-talk/no-code-is-the-future-of-software-development-but-is-it-secure-abe587cdc314?source=post_internal_links---------5----------------------------)

[[Hylo Biz](https://medium.com/@hylobiz?source=post_internal_links---------6----------------------------)](https://medium.com/@hylobiz?source=post_internal_links---------6----------------------------)

[Hylo Biz](https://medium.com/@hylobiz?source=post_internal_links---------6----------------------------)

**[How to Easily Generate e-Way Bills?](https://medium.com/@hylobiz/how-to-easily-generate-e-way-bills-45299e07c28e?source=post_internal_links---------6----------------------------)**

[[](https://medium.com/@hylobiz/how-to-easily-generate-e-way-bills-45299e07c28e?source=post_internal_links---------6----------------------------)](https://medium.com/@hylobiz/how-to-easily-generate-e-way-bills-45299e07c28e?source=post_internal_links---------6----------------------------)

[[Nick Sharkey](https://medium.com/@nicholassharkey?source=post_internal_links---------7----------------------------)](https://medium.com/@nicholassharkey?source=post_internal_links---------7----------------------------)

[Nick Sharkey](https://medium.com/@nicholassharkey?source=post_internal_links---------7----------------------------)

**[Acquiring and Consolidating NFL Data (but really just R](https://medium.com/@nicholassharkey/acquiring-and-consolidating-nfl-data-but-really-just-r-stuff-8636770c4835?source=post_internal_links---------7----------------------------)**

### [Download .NET 6.0 (Linux, macOS, and Windows) - Microsoft](https://dotnet.microsoft.com/en-us/download/dotnet/6.0)

[https://dotnet.microsoft.com › en-us › download › dotnet](https://dotnet.microsoft.com/en-us/download/dotnet/6.0)

NET 6.0 downloads for Linux, macOS, and Windows. .NET is a free, cross-platform, ... **NET Core Runtime enables you to run existing web/server applications**.

# Download .NET 6.0

Not sure what to download? [See recommended downloads for the latest version of .NET](https://dotnet.microsoft.com/en-us/download).

6.0.13 **Security patch Tooltip: This release contains fixes for security issues. If using an older patch release, you should upgrade to get these fixes.**

[Release notes](https://github.com/dotnet/core/blob/main/release-notes/6.0/6.0.13/6.0.13.md?WT.mc_id=dotnet-35129-website)

**Latest release date**

January 10, 2023

## Build apps - SDK

Tooltip: Do you want to build apps? The software development kit (SDK) includes everything you need to build and run .NET applications, using command-line tools and any editor (like Visual Studio).

### SDK 6.0.405

| Downloads for .NET 6.0 SDK (v6.0.405) | | |
| --- | --- | --- |
| **OS** | **Installers** | **Binaries** |
| Linux | [Package manager instructions](https://learn.microsoft.com/dotnet/core/install/linux?WT.mc_id=dotnet-35129-website) | [Arm32](https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/sdk-6.0.405-linux-arm32-binaries) | [Arm32 Alpine](https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/sdk-6.0.405-linux-arm32-alpine-binaries) | [Arm64](https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/sdk-6.0.405-linux-arm64-binaries) | [Arm64 Alpine](https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/sdk-6.0.405-linux-arm64-alpine-binaries) | [x64](https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/sdk-6.0.405-linux-x64-binaries) | [x64 Alpine](https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/sdk-6.0.405-linux-x64-alpine-binaries) |
| macOS | [Arm64](https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/sdk-6.0.405-macos-arm64-installer) | [x64](https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/sdk-6.0.405-macos-x64-installer) | [Arm64](https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/sdk-6.0.405-macos-arm64-binaries) | [x64](https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/sdk-6.0.405-macos-x64-binaries) |
| Windows | [Arm64](https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/sdk-6.0.405-windows-arm64-installer) | [x64](https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/sdk-6.0.405-windows-x64-installer) | [x86](https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/sdk-6.0.405-windows-x86-installer) | [winget instructions](https://learn.microsoft.com/dotnet/core/install/windows?WT.mc_id=dotnet-35129-website#install-with-windows-package-manager-winget) | [Arm64](https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/sdk-6.0.405-windows-arm64-binaries) | [x64](https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/sdk-6.0.405-windows-x64-binaries) | [x86](https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/sdk-6.0.405-windows-x86-binaries) |
| All | [dotnet-install scripts](https://dotnet.microsoft.com/en-us/download/dotnet/scripts) |  |

**Included runtimes**

.NET Runtime 6.0.13

ASP.NET Core Runtime 6.0.13

.NET Desktop Runtime 6.0.13

**Language support**

C# 10.0

F# 6.0

Visual Basic 16.9

### SDK 6.0.308

| Downloads for .NET 6.0 SDK (v6.0.308) | | |
| --- | --- | --- |
| **OS** | **Installers** | **Binaries** |
| Linux | [Package manager instructions](https://learn.microsoft.com/dotnet/core/install/linux?WT.mc_id=dotnet-35129-website) | [Arm32](https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/sdk-6.0.308-linux-arm32-binaries) | [Arm32 Alpine](https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/sdk-6.0.308-linux-arm32-alpine-binaries) | [Arm64](https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/sdk-6.0.308-linux-arm64-binaries) | [Arm64 Alpine](https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/sdk-6.0.308-linux-arm64-alpine-binaries) | [x64](https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/sdk-6.0.308-linux-x64-binaries) | [x64 Alpine](https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/sdk-6.0.308-linux-x64-alpine-binaries) |
| macOS | [Arm64](https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/sdk-6.0.308-macos-arm64-installer) | [x64](https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/sdk-6.0.308-macos-x64-installer) | [Arm64](https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/sdk-6.0.308-macos-arm64-binaries) | [x64](https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/sdk-6.0.308-macos-x64-binaries) |
| Windows | [Arm64](https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/sdk-6.0.308-windows-arm64-installer) | [x64](https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/sdk-6.0.308-windows-x64-installer) | [x86](https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/sdk-6.0.308-windows-x86-installer) | [winget instructions](https://learn.microsoft.com/dotnet/core/install/windows?WT.mc_id=dotnet-35129-website#install-with-windows-package-manager-winget) | [Arm64](https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/sdk-6.0.308-windows-arm64-binaries) | [x64](https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/sdk-6.0.308-windows-x64-binaries) | [x86](https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/sdk-6.0.308-windows-x86-binaries) |
| All | [dotnet-install scripts](https://dotnet.microsoft.com/en-us/download/dotnet/scripts) |  |

**Visual Studio support**

Visual Studio 2022 (v17.2)

Visual Studio 2022 for Mac (v17.4)

**Included in**

Visual Studio 17.2.11

**Included runtimes**

.NET Runtime 6.0.13

ASP.NET Core Runtime 6.0.13

.NET Desktop Runtime 6.0.13

**Language support**

C# 10.0

F# 6.0

Visual Basic 16.9

### SDK 6.0.113

| Downloads for .NET 6.0 SDK (v6.0.113) | | |
| --- | --- | --- |
| **OS** | **Installers** | **Binaries** |
| Linux | [Package manager instructions](https://learn.microsoft.com/dotnet/core/install/linux?WT.mc_id=dotnet-35129-website) | [Arm32](https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/sdk-6.0.113-linux-arm32-binaries) | [Arm32 Alpine](https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/sdk-6.0.113-linux-arm32-alpine-binaries) | [Arm64](https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/sdk-6.0.113-linux-arm64-binaries) | [Arm64 Alpine](https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/sdk-6.0.113-linux-arm64-alpine-binaries) | [x64](https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/sdk-6.0.113-linux-x64-binaries) | [x64 Alpine](https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/sdk-6.0.113-linux-x64-alpine-binaries) |
| macOS | [Arm64](https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/sdk-6.0.113-macos-arm64-installer) | [x64](https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/sdk-6.0.113-macos-x64-installer) | [Arm64](https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/sdk-6.0.113-macos-arm64-binaries) | [x64](https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/sdk-6.0.113-macos-x64-binaries) |
| Windows | [Arm64](https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/sdk-6.0.113-windows-arm64-installer) | [x64](https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/sdk-6.0.113-windows-x64-installer) | [x86](https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/sdk-6.0.113-windows-x86-installer) | [winget instructions](https://learn.microsoft.com/dotnet/core/install/windows?WT.mc_id=dotnet-35129-website#install-with-windows-package-manager-winget) | [Arm64](https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/sdk-6.0.113-windows-arm64-binaries) | [x64](https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/sdk-6.0.113-windows-x64-binaries) | [x86](https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/sdk-6.0.113-windows-x86-binaries) |
| All | [dotnet-install scripts](https://dotnet.microsoft.com/en-us/download/dotnet/scripts) |  |

**Visual Studio support**

Visual Studio 2022 (v17.0)

Visual Studio 2022 for Mac (v17.4)

**Included in**

Visual Studio 17.0.18

**Included runtimes**

.NET Runtime 6.0.13

ASP.NET Core Runtime 6.0.13

.NET Desktop Runtime 6.0.13

**Language support**

C# 10.0

F# 6.0

Visual Basic 16.9

## Run apps - Runtime

Tooltip: Do you want to run apps? The runtime includes everything you need to run .NET applications. The runtime is also included in the SDK.

### ASP.NET Core Runtime 6.0.13

The ASP.NET Core Runtime enables you to run existing web/server applications. **On Windows, we recommend installing the Hosting Bundle, which includes the .NET Runtime and IIS support.**

**IIS runtime support (ASP.NET Core Module v2)**

16.0.22335.13

| Downloads for ASP.NET Core 6.0 Runtime (v6.0.13) | | |
| --- | --- | --- |
| **OS** | **Installers** | **Binaries** |
| Linux | [Package manager instructions](https://learn.microsoft.com/dotnet/core/install/linux?WT.mc_id=dotnet-35129-website) | [Arm32](https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/runtime-aspnetcore-6.0.13-linux-arm32-binaries) | [Arm32 Alpine](https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/runtime-aspnetcore-6.0.13-linux-arm32-alpine-binaries) | [Arm64](https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/runtime-aspnetcore-6.0.13-linux-arm64-binaries) | [Arm64 Alpine](https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/runtime-aspnetcore-6.0.13-linux-arm64-alpine-binaries) | [x64](https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/runtime-aspnetcore-6.0.13-linux-x64-binaries) | [x64 Alpine](https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/runtime-aspnetcore-6.0.13-linux-x64-alpine-binaries) |
| macOS |  | [Arm64](https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/runtime-aspnetcore-6.0.13-macos-arm64-binaries) | [x64](https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/runtime-aspnetcore-6.0.13-macos-x64-binaries) |
| Windows | [Hosting Bundle](https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/runtime-aspnetcore-6.0.13-windows-hosting-bundle-installer) | [x64](https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/runtime-aspnetcore-6.0.13-windows-x64-installer) | [x86](https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/runtime-aspnetcore-6.0.13-windows-x86-installer) | [winget instructions](https://learn.microsoft.com/dotnet/core/install/windows?WT.mc_id=dotnet-35129-website#install-with-windows-package-manager-winget) | [Arm64](https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/runtime-aspnetcore-6.0.13-windows-arm64-binaries) | [x64](https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/runtime-aspnetcore-6.0.13-windows-x64-binaries) | [x86](https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/runtime-aspnetcore-6.0.13-windows-x86-binaries) |

### .NET Desktop Runtime 6.0.13

The .NET Desktop Runtime enables you to run existing Windows desktop applications. **This release includes the .NET Runtime; you don't need to install it separately.**

| Downloads for .NET 6.0 Desktop Runtime (v6.0.13) | | |
| --- | --- | --- |
| **OS** | **Installers** | **Binaries** |
| Windows | [Arm64](https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/runtime-desktop-6.0.13-windows-arm64-installer) | [x64](https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/runtime-desktop-6.0.13-windows-x64-installer) | [x86](https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/runtime-desktop-6.0.13-windows-x86-installer) | [winget instructions](https://learn.microsoft.com/dotnet/core/install/windows?WT.mc_id=dotnet-35129-website#install-with-windows-package-manager-winget) |  |

### .NET Runtime 6.0.13

The .NET Runtime contains just the components needed to run a console app. Typically, you'd also install either the ASP.NET Core Runtime or .NET Desktop Runtime.

| Downloads for .NET 6.0 Runtime (v6.0.13) | | |
| --- | --- | --- |
| **OS** | **Installers** | **Binaries** |
| Linux | [Package manager instructions](https://learn.microsoft.com/dotnet/core/install/linux?WT.mc_id=dotnet-35129-website) | [Arm32](https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/runtime-6.0.13-linux-arm32-binaries) | [Arm32 Alpine](https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/runtime-6.0.13-linux-arm32-alpine-binaries) | [Arm64](https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/runtime-6.0.13-linux-arm64-binaries) | [Arm64 Alpine](https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/runtime-6.0.13-linux-arm64-alpine-binaries) | [x64](https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/runtime-6.0.13-linux-x64-binaries) | [x64 Alpine](https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/runtime-6.0.13-linux-x64-alpine-binaries) |
| macOS | [Arm64](https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/runtime-6.0.13-macos-arm64-installer) | [x64](https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/runtime-6.0.13-macos-x64-installer) | [Arm64](https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/runtime-6.0.13-macos-arm64-binaries) | [x64](https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/runtime-6.0.13-macos-x64-binaries) |
| Windows | [Arm64](https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/runtime-6.0.13-windows-arm64-installer) | [x64](https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/runtime-6.0.13-windows-x64-installer) | [x86](https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/runtime-6.0.13-windows-x86-installer) | [winget instructions](https://learn.microsoft.com/dotnet/core/install/windows?WT.mc_id=dotnet-35129-website#install-with-windows-package-manager-winget) | [Arm64](https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/runtime-6.0.13-windows-arm64-binaries) | [x64](https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/runtime-6.0.13-windows-x64-binaries) | [x86](https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/runtime-6.0.13-windows-x86-binaries) |
| All | [dotnet-install scripts](https://dotnet.microsoft.com/en-us/download/dotnet/scripts) |  |

6.0.12 **Security patch Tooltip: This release contains fixes for security issues. If using an older patch release, you should upgrade to get these fixes.**

6.0.11

6.0.10 **Security patch Tooltip: This release contains fixes for security issues. If using an older patch release, you should upgrade to get these fixes.**

6.0.9 **Security patch Tooltip: This release contains fixes for security issues. If using an older patch release, you should upgrade to get these fixes.**

[**https://download.visualstudio.microsoft.com/download/pr/b7f1ac8a-c25d-408f-b54c-cea1a30007fb/697ad34ba7c9bf71310194ba1c4c1270/dotnet-sdk-6.0.308-win-x86.exe**](https://download.visualstudio.microsoft.com/download/pr/b7f1ac8a-c25d-408f-b54c-cea1a30007fb/697ad34ba7c9bf71310194ba1c4c1270/dotnet-sdk-6.0.308-win-x86.exe)

**Using Visual Studio?** This release is only compatible with Visual Studio 2022 (v17.2). Using a different version? See [.NET SDKs for Visual Studio](https://dotnet.microsoft.com/en-us/download/visual-studio-sdks).

If your download doesn't start after 30 seconds, [click here to download manually](https://download.visualstudio.microsoft.com/download/pr/b7f1ac8a-c25d-408f-b54c-cea1a30007fb/697ad34ba7c9bf71310194ba1c4c1270/dotnet-sdk-6.0.308-win-x86.exe).

Direct link

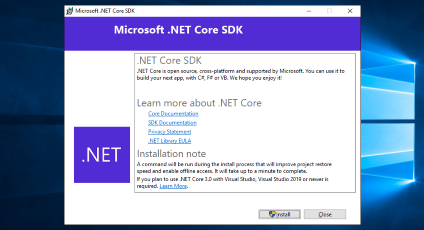
<https://download.visualstudio.microsoft.com/download/pr/b7f1ac8a-c25d-408f-b54c-cea1a30007fb/697ad34ba7c9bf71310194ba1c4c1270/dotnet-sdk-6.0.308-win-x86.exe>

Copy

Checksum (SHA512)

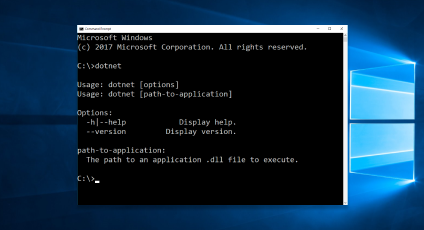


Copy



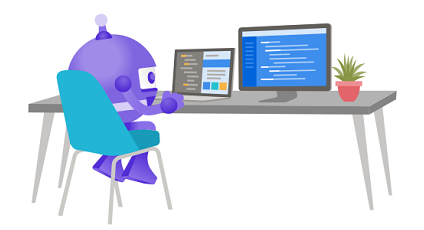
## Step 1: Run Installer

When your download completes, run the installer and complete the steps to install .NET on your machine.



## Step 2: Verify Installation

When the installer completes, open a new command prompt and run the dotnet command. This will verify .NET is correctly installed and ready to use.



## Step 3: Get Started

Now that you've got .NET installed, let's build your first app with our step-by-step tutorial.

[**Hello World in 5 minutes tutorial**](https://dotnet.microsoft.com/en-us/learn/dotnet/hello-world-tutorial/intro?sdk-installed=true)

### [.NET Community](https://dotnet.microsoft.com/en-us/platform/community)

[Connect with other .NET developers.](https://dotnet.microsoft.com/en-us/platform/community)

## [.NET for Beginners videos](https://dotnet.microsoft.com/en-us/learn/videos)

[Explore the .NET developer platform with these free, short videos.](https://dotnet.microsoft.com/en-us/learn/videos)

### [Introducing ML.NET](https://dotnet.microsoft.com/en-us/apps/machinelearning-ai/ml-dotnet)

[A new machine learning framework for .NET](https://dotnet.microsoft.com/en-us/apps/machinelearning-ai/ml-dotnet)

To determine system type for Windows

System Type x64-based PC means that **you have a 64-bit system**. System Type x86-based PC means that you have a 32-bit system.Jul 8, 2021

Top of Form



Bottom of Form

[https://lh3.googleusercontent.com/ogw/AAEL6sjWA7X4rfgitmnRYwhrm8drVKTysRtXBRKLxuo8iQ=s32-c-mo](https://accounts.google.com/SignOutOptions?hl=en&continue=https://www.google.com/search%3Fq%3Dwhat%2Barchitecture%2Bis%2Bdoes%2Bapple%2Bm1%26rlz%3D1C1GCEU_enUS899US899%26sxsrf%3DAJOqlzXuFQ34in2oWmwW1_M-KduyAcZqFQ%253A1674073657767%26ei%3DOVbIY7ysLu6zqtsP656QuAY%26oq%3Dwhat%2Barchitecture%2Bis%2Bdoes%2BAppl%26gs_lcp%3DCgxnd3Mtd2l6LXNlcnAQARgDMgUIIRCgATIFCCEQoAEyBQghEKsCMggIIRAWEB4QHTIICCEQFhAeEB0yCAghEBYQHhAdMggIIRAWEB4QHTIICCEQFhAeEB0yCAghEBYQHhAdMggIIRAWEB4QHToKCAAQRxDWBBCwAzoHCAAQsAMQQzoECCMQJzoFCAAQgAQ6BggAEBYQHjoFCAAQhgM6CAgAEBYQHhAPOgoIABCABBCHAhAUSgQIQRgASgQIRhgAUJoKWKNhYLOTAWgCcAF4AIABjgGIAYsQkgEEOS4xMpgBAKABAcgBCcABAQ%26sclient%3Dgws-wiz-serp)

# Search Modes

All

[Images](https://www.google.com/search?q=what+architecture+is+does+apple+m1&rlz=1C1GCEU_enUS899US899&sxsrf=AJOqlzX4gakv8Rhhg90TaexYqE0dKR_UHA:1674073679208&source=lnms&tbm=isch&sa=X&ved=2ahUKEwiWiui3-tH8AhVHmWoFHQ9NBGEQ_AUoAXoECAEQAw)

[Shopping](https://www.google.com/search?q=what+architecture+is+does+apple+m1&rlz=1C1GCEU_enUS899US899&sxsrf=AJOqlzX4gakv8Rhhg90TaexYqE0dKR_UHA:1674073679208&source=lnms&tbm=shop&sa=X&ved=2ahUKEwiWiui3-tH8AhVHmWoFHQ9NBGEQ_AUoAnoECAEQBA)

[News](https://www.google.com/search?q=what+architecture+is+does+apple+m1&rlz=1C1GCEU_enUS899US899&sxsrf=AJOqlzX4gakv8Rhhg90TaexYqE0dKR_UHA:1674073679208&source=lnms&tbm=nws&sa=X&ved=2ahUKEwiWiui3-tH8AhVHmWoFHQ9NBGEQ_AUoA3oECAEQBQ)

[Videos](https://www.google.com/search?q=what+architecture+is+does+apple+m1&rlz=1C1GCEU_enUS899US899&sxsrf=AJOqlzX4gakv8Rhhg90TaexYqE0dKR_UHA:1674073679208&source=lnms&tbm=vid&sa=X&ved=2ahUKEwiWiui3-tH8AhVHmWoFHQ9NBGEQ_AUoBHoECAEQBg)

More

Tools

About 17,500,000 results (0.67 seconds)

# Search Results

## Featured snippet from the web

M1 Pro and M1 Max introduce a **system-on-a-chip (SoC)** architecture to pro systems for the first time. The chips feature fast unified memory, industry-leading performance per watt, and incredible power efficiency, along with increased memory bandwidth and capacity.Oct 18, 2021

### [Introducing M1 Pro and M1 Max: the most powerful chips ...](https://www.apple.com/newsroom/2021/10/introducing-m1-pro-and-m1-max-the-most-powerful-chips-apple-has-ever-built/" \l ":~:text=M1%20Pro%20and%20M1%20Max%20introduce%20a%20system%2Don%2Da,increased%20memory%20bandwidth%20and%20capacity.)

[https://www.apple.com › newsroom › 2021/10 › introduci...](https://www.apple.com/newsroom/2021/10/introducing-m1-pro-and-m1-max-the-most-powerful-chips-apple-has-ever-built/" \l ":~:text=M1%20Pro%20and%20M1%20Max%20introduce%20a%20system%2Don%2Da,increased%20memory%20bandwidth%20and%20capacity.)

[About featured snippets](https://www.google.com/url?url=https://support.google.com/websearch?p%3Dfeatured_snippets%26hl%3Den-US&rct=j&q=&esrc=s&usg=AOvVaw185ES_DZIoacbenypeQ03C&hl=en&sa=X&ved=2ahUKEwiWiui3-tH8AhVHmWoFHQ9NBGEQrpwBegQIFhAC)•

[Feedback](https://www.google.com/search?q=what+architecture+is+does+apple+m1&rlz=1C1GCEU_enUS899US899&sxsrf=AJOqlzXuFQ34in2oWmwW1_M-KduyAcZqFQ%3A1674073657767&ei=OVbIY7ysLu6zqtsP656QuAY&oq=what+architecture+is+does+Appl&gs_lcp=Cgxnd3Mtd2l6LXNlcnAQARgDMgUIIRCgATIFCCEQoAEyBQghEKsCMggIIRAWEB4QHTIICCEQFhAeEB0yCAghEBYQHhAdMggIIRAWEB4QHTIICCEQFhAeEB0yCAghEBYQHhAdMggIIRAWEB4QHToKCAAQRxDWBBCwAzoHCAAQsAMQQzoECCMQJzoFCAAQgAQ6BggAEBYQHjoFCAAQhgM6CAgAEBYQHhAPOgoIABCABBCHAhAUSgQIQRgASgQIRhgAUJoKWKNhYLOTAWgCcAF4AIABjgGIAYsQkgEEOS4xMpgBAKABAcgBCcABAQ&sclient=gws-wiz-serp)

People also ask

Is Mac M1 ARM64 or x64?

So, developers need to be able to run x86\_64 containers on newer Mac computers that are built with an Apple Silicon (M1) processor, which is an **ARM64 architecture**.Sep 21, 2022

### [Running x86\_64-based containers on Mac computers with an Apple ...](https://developer.ibm.com/tutorials/running-x86-64-containers-mac-silicon-m1/)

[https://developer.ibm.com › tutorials › running-x86-64-c...](https://developer.ibm.com/tutorials/running-x86-64-containers-mac-silicon-m1/)

Search for: [Is Mac M1 ARM64 or x64?](https://www.google.com/search?rlz=1C1GCEU_enUS899US899&sxsrf=AJOqlzX4gakv8Rhhg90TaexYqE0dKR_UHA:1674073679208&q=Is+Mac+M1+ARM64+or+x64%3F&sa=X&ved=2ahUKEwiWiui3-tH8AhVHmWoFHQ9NBGEQzmd6BAgiEAU)

Is M1 ARM or ARM64?

Is Mac M1 chip ARM64?

Is Mac M1 x64 or x86?

Is the M1 a x64?

Is MacBook Air M1 ARM or x64?

[Feedback](https://www.google.com/search?q=what+architecture+is+does+apple+m1&rlz=1C1GCEU_enUS899US899&sxsrf=AJOqlzXuFQ34in2oWmwW1_M-KduyAcZqFQ%3A1674073657767&ei=OVbIY7ysLu6zqtsP656QuAY&oq=what+architecture+is+does+Appl&gs_lcp=Cgxnd3Mtd2l6LXNlcnAQARgDMgUIIRCgATIFCCEQoAEyBQghEKsCMggIIRAWEB4QHTIICCEQFhAeEB0yCAghEBYQHhAdMggIIRAWEB4QHTIICCEQFhAeEB0yCAghEBYQHhAdMggIIRAWEB4QHToKCAAQRxDWBBCwAzoHCAAQsAMQQzoECCMQJzoFCAAQgAQ6BggAEBYQHjoFCAAQhgM6CAgAEBYQHhAPOgoIABCABBCHAhAUSgQIQRgASgQIRhgAUJoKWKNhYLOTAWgCcAF4AIABjgGIAYsQkgEEOS4xMpgBAKABAcgBCcABAQ&sclient=gws-wiz-serp)

### [Apple M1 - Wikipedia](https://en.wikipedia.org/wiki/Apple_M1)

[https://en.wikipedia.org › wiki › Apple\_M1](https://en.wikipedia.org/wiki/Apple_M1)

Apple M1 is a series **of ARM-based systems-on-a-chip (SoCs**) designed by Apple Inc. as a central processing unit (CPU) and graphics processing unit (GPU) for ...

Max. CPU clock rate: 3.2 GHz

L1 cache: 192+128 KB per core (performance ...

L2 cache: Performance Cores; M1: 12 MB; M1 ...

Last level cache: M1: 8 MB; M1 Pro: 24 MB; M1 ...

‎[Apple M2](https://en.wikipedia.org/wiki/Apple_M2) · ‎[Apple T2](https://en.wikipedia.org/wiki/Apple_T2) · ‎[Mac transition to Intel processors](https://en.wikipedia.org/wiki/Mac_transition_to_Intel_processors) · ‎[Asahi Linux](https://en.wikipedia.org/wiki/Asahi_Linux)

### [Apple M1 Processor Overview and Compatibility - Toptal](https://www.toptal.com/apple/apple-m1-processor-compatibility-overview)

[https://www.toptal.com › apple › apple-m1-processor-c...](https://www.toptal.com/apple/apple-m1-processor-compatibility-overview)

The M1 is **an ARM processor**, not an x86 processor. · It integrates more components than an Intel CPU. · The Apple M1 also integrates RAM in the same package. · It ...

‎[Designed By Apple, Marketed...](https://www.toptal.com/apple/apple-m1-processor-compatibility-overview#:~:text=Designed%20by%20Apple%2C%20Marketed%20by%20Apple) · ‎[Apple M1 Design And Features](https://www.toptal.com/apple/apple-m1-processor-compatibility-overview#:~:text=Apple%20M1%20Design%20and%20Features) · ‎[Implications For Real-World...](https://www.toptal.com/apple/apple-m1-processor-compatibility-overview#:~:text=Implications%20for%20Real%2Dworld%20Users)

### [Apple M1 Chip: Everything You Need to Know - MacRumors](https://www.macrumors.com/guide/m1/)

[https://www.macrumors.com › guide › m1](https://www.macrumors.com/guide/m1/)

Oct 13, 2022 — Built into the M1 chip, **the unified memory architecture** lets the CPU, GPU, and other processor components don't need to copy data between one ...

‎[Apple's M1 Chip Explained](https://www.macrumors.com/guide/m1/#:~:text=Apple%27s%20M1%20Chip%20Explained) · ‎[Gpu](https://www.macrumors.com/guide/m1/#:~:text=GPU,-The%20%E2%80%8CApple%20Silicon%E2%80%8C%20chip%20has) · ‎[Running Apps On M1 Macs](https://www.macrumors.com/guide/m1/#:~:text=Running%20Apps%20on%20M1%20Macs)

### [Apple M1 Chip. Everything You Wanted to Know About It](https://logidots.com/insights/apple-m1-chip-everything-you-wanted-to-know-about-it/)

[https://logidots.com › insights › apple-m1-chip-everythi...](https://logidots.com/insights/apple-m1-chip-everything-you-wanted-to-know-about-it/)

M1 uses a 5-nanometre lithography process · This Chipset has a whopping 16 billion transistors. · M1 is the first computer Chipset to use **the Architecture of ARM** ...

‎[What Is The Apple M1 Chip?](https://logidots.com/insights/apple-m1-chip-everything-you-wanted-to-know-about-it/#:~:text=What%20is%20the%20Apple%20M1%20Chip%3F) · ‎[Apple M1 Architecture](https://logidots.com/insights/apple-m1-chip-everything-you-wanted-to-know-about-it/#:~:text=Apple%20M1%20Architecture) · ‎[Apple M1: Performance](https://logidots.com/insights/apple-m1-chip-everything-you-wanted-to-know-about-it/#:~:text=Apple%20M1%3A%20Performance)

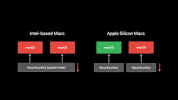
### [Apple unleashes M1](https://www.apple.com/newsroom/2020/11/apple-unleashes-m1/)

[https://www.apple.com › newsroom › 2020/11 › apple-...](https://www.apple.com/newsroom/2020/11/apple-unleashes-m1/)

Nov 10, 2020 — The M1 chip brings **the Apple Neural Engine** to the Mac, greatly accelerating machine learning (ML) tasks. Featuring Apple's most advanced 16-core ...

### [Explore the new system architecture of Apple silicon Macs](https://developer.apple.com/videos/play/wwdc2020/10686/)

[https://developer.apple.com › videos › play › wwdc2020](https://developer.apple.com/videos/play/wwdc2020/10686/)

[](https://developer.apple.com/videos/play/wwdc2020/10686/)

Discover how Macs with **Apple** silicon will deliver modern advantages using **Apple's** System-on-Chip (SoC) **architecture**. Leveraging a unified memory ...

Apple Developer · Sep 18, 2020

### [Apple unveils M1 Ultra, the world's most powerful chip for a ...](https://www.apple.com/newsroom/2022/03/apple-unveils-m1-ultra-the-worlds-most-powerful-chip-for-a-personal-computer/)

[https://www.apple.com › newsroom › 2022/03 › apple-...](https://www.apple.com/newsroom/2022/03/apple-unveils-m1-ultra-the-worlds-most-powerful-chip-for-a-personal-computer/)

Mar 8, 2022 — **Groundbreaking UltraFusion Architecture** ... The foundation for M1 Ultra is the extremely powerful and power-efficient M1 Max. To build M1 Ultra, ...

### [Computer Architect's View on Apple M1 Ultra Chip](https://yasassri.medium.com/computer-architects-view-on-apple-m1-ultra-chip-821805860c42)

[https://yasassri.medium.com › computer-architects-view...](https://yasassri.medium.com/computer-architects-view-on-apple-m1-ultra-chip-821805860c42)

Mar 20, 2022 — Usually, the general-purpose computers use **Complex Instruction Set Computing (CISC) architecture for CPUs**. Still, the Apple M1 platform is based ...

‎[Apple Silicon M1 Series](https://yasassri.medium.com/computer-architects-view-on-apple-m1-ultra-chip-821805860c42#:~:text=Apple%20Silicon%20M1%20Series) · ‎[Learn More About Computer...](https://yasassri.medium.com/computer-architects-view-on-apple-m1-ultra-chip-821805860c42#:~:text=Learn%20more%20about%20Computer%20Architecture) · ‎[You Might Be Interested](https://yasassri.medium.com/computer-architects-view-on-apple-m1-ultra-chip-821805860c42#:~:text=You%20might%20be%20interested%2C)

### [Computer Architect's View on Apple M1 Ultra Chip - LinkedIn](https://www.linkedin.com/pulse/computer-architects-view-apple-m1-ultra-chip-yasas-sri-wickramasinghe)

[https://www.linkedin.com › pulse › computer-architects-vi...](https://www.linkedin.com/pulse/computer-architects-view-apple-m1-ultra-chip-yasas-sri-wickramasinghe)

Mar 20, 2022 — Usually, the general-purpose computers use **Complex Instruction Set Computing (CISC) architecture for CPUs**. Still, the Apple M1 platform is based ...

People also ask

Is my Mac ARM64 or x64?

How do I know if my Mac is ARM64 or x64?

Is Mac M1 ARM or x86?

How do I know if I have ARM64 or x64?

[Feedback](https://www.google.com/search?q=what+architecture+is+does+apple+m1&rlz=1C1GCEU_enUS899US899&sxsrf=AJOqlzXuFQ34in2oWmwW1_M-KduyAcZqFQ%3A1674073657767&ei=OVbIY7ysLu6zqtsP656QuAY&oq=what+architecture+is+does+Appl&gs_lcp=Cgxnd3Mtd2l6LXNlcnAQARgDMgUIIRCgATIFCCEQoAEyBQghEKsCMggIIRAWEB4QHTIICCEQFhAeEB0yCAghEBYQHhAdMggIIRAWEB4QHTIICCEQFhAeEB0yCAghEBYQHhAdMggIIRAWEB4QHToKCAAQRxDWBBCwAzoHCAAQsAMQQzoECCMQJzoFCAAQgAQ6BggAEBYQHjoFCAAQhgM6CAgAEBYQHhAPOgoIABCABBCHAhAUSgQIQRgASgQIRhgAUJoKWKNhYLOTAWgCcAF4AIABjgGIAYsQkgEEOS4xMpgBAKABAcgBCcABAQ&sclient=gws-wiz-serp)

Related searches

* [apple m1 architecture](https://www.google.com/search?rlz=1C1GCEU_enUS899US899&sxsrf=AJOqlzX4gakv8Rhhg90TaexYqE0dKR_UHA:1674073679208&q=Apple+M1+architecture&sa=X&ved=2ahUKEwiWiui3-tH8AhVHmWoFHQ9NBGEQ1QJ6BAhEEAE)
* [m1](https://www.google.com/search?rlz=1C1GCEU_enUS899US899&sxsrf=AJOqlzX4gakv8Rhhg90TaexYqE0dKR_UHA:1674073679208&q=M1+chip+vs+Intel+i7&sa=X&ved=2ahUKEwiWiui3-tH8AhVHmWoFHQ9NBGEQ1QJ6BAhBEAE)**[chip vs intel i7](https://www.google.com/search?rlz=1C1GCEU_enUS899US899&sxsrf=AJOqlzX4gakv8Rhhg90TaexYqE0dKR_UHA:1674073679208&q=M1+chip+vs+Intel+i7&sa=X&ved=2ahUKEwiWiui3-tH8AhVHmWoFHQ9NBGEQ1QJ6BAhBEAE)**
* [apple](https://www.google.com/search?rlz=1C1GCEU_enUS899US899&sxsrf=AJOqlzX4gakv8Rhhg90TaexYqE0dKR_UHA:1674073679208&q=Apple+M2&sa=X&ved=2ahUKEwiWiui3-tH8AhVHmWoFHQ9NBGEQ1QJ6BAg-EAE)**[m2](https://www.google.com/search?rlz=1C1GCEU_enUS899US899&sxsrf=AJOqlzX4gakv8Rhhg90TaexYqE0dKR_UHA:1674073679208&q=Apple+M2&sa=X&ved=2ahUKEwiWiui3-tH8AhVHmWoFHQ9NBGEQ1QJ6BAg-EAE)**
* [apple m1](https://www.google.com/search?rlz=1C1GCEU_enUS899US899&sxsrf=AJOqlzX4gakv8Rhhg90TaexYqE0dKR_UHA:1674073679208&q=Apple+M1+chip&sa=X&ved=2ahUKEwiWiui3-tH8AhVHmWoFHQ9NBGEQ1QJ6BAg9EAE)**[chip](https://www.google.com/search?rlz=1C1GCEU_enUS899US899&sxsrf=AJOqlzX4gakv8Rhhg90TaexYqE0dKR_UHA:1674073679208&q=Apple+M1+chip&sa=X&ved=2ahUKEwiWiui3-tH8AhVHmWoFHQ9NBGEQ1QJ6BAg9EAE)**
* [apple m1](https://www.google.com/search?rlz=1C1GCEU_enUS899US899&sxsrf=AJOqlzX4gakv8Rhhg90TaexYqE0dKR_UHA:1674073679208&q=Apple+M1+software+compatibility+list&sa=X&ved=2ahUKEwiWiui3-tH8AhVHmWoFHQ9NBGEQ1QJ6BAg6EAE)**[software compatibility list](https://www.google.com/search?rlz=1C1GCEU_enUS899US899&sxsrf=AJOqlzX4gakv8Rhhg90TaexYqE0dKR_UHA:1674073679208&q=Apple+M1+software+compatibility+list&sa=X&ved=2ahUKEwiWiui3-tH8AhVHmWoFHQ9NBGEQ1QJ6BAg6EAE)**
* **[macbook pro](https://www.google.com/search?rlz=1C1GCEU_enUS899US899&sxsrf=AJOqlzX4gakv8Rhhg90TaexYqE0dKR_UHA:1674073679208&q=MacBook+Pro+M1&sa=X&ved=2ahUKEwiWiui3-tH8AhVHmWoFHQ9NBGEQ1QJ6BAg5EAE)**[m1](https://www.google.com/search?rlz=1C1GCEU_enUS899US899&sxsrf=AJOqlzX4gakv8Rhhg90TaexYqE0dKR_UHA:1674073679208&q=MacBook+Pro+M1&sa=X&ved=2ahUKEwiWiui3-tH8AhVHmWoFHQ9NBGEQ1QJ6BAg5EAE)
* [apple m1](https://www.google.com/search?rlz=1C1GCEU_enUS899US899&sxsrf=AJOqlzX4gakv8Rhhg90TaexYqE0dKR_UHA:1674073679208&q=Apple+M1+Pro&sa=X&ved=2ahUKEwiWiui3-tH8AhVHmWoFHQ9NBGEQ1QJ6BAg3EAE)**[pro](https://www.google.com/search?rlz=1C1GCEU_enUS899US899&sxsrf=AJOqlzX4gakv8Rhhg90TaexYqE0dKR_UHA:1674073679208&q=Apple+M1+Pro&sa=X&ved=2ahUKEwiWiui3-tH8AhVHmWoFHQ9NBGEQ1QJ6BAg3EAE)**
* [m1](https://www.google.com/search?rlz=1C1GCEU_enUS899US899&sxsrf=AJOqlzX4gakv8Rhhg90TaexYqE0dKR_UHA:1674073679208&q=M1+processor+vs+Intel&sa=X&ved=2ahUKEwiWiui3-tH8AhVHmWoFHQ9NBGEQ1QJ6BAg0EAE)**[processor vs intel](https://www.google.com/search?rlz=1C1GCEU_enUS899US899&sxsrf=AJOqlzX4gakv8Rhhg90TaexYqE0dKR_UHA:1674073679208&q=M1+processor+vs+Intel&sa=X&ved=2ahUKEwiWiui3-tH8AhVHmWoFHQ9NBGEQ1QJ6BAg0EAE)**

# Install .NET on Linux

* Article
* 01/10/2023
* 2 minutes to read
* 7 contributors

Feedback

Install on Linux

This article details how to install .NET on various Linux distributions either manually, via a package manager, or via a [container](https://learn.microsoft.com/en-us/dotnet/core/docker/introduction#net-images).

## Manual installation

You can install .NET manually in the following ways:

* [Manual install](https://learn.microsoft.com/en-us/dotnet/core/install/linux-scripted-manual#manual-install)
* [Scripted install](https://learn.microsoft.com/en-us/dotnet/core/install/linux-scripted-manual#scripted-install)

You may need to install [.NET dependencies](https://github.com/dotnet/core/blob/main/release-notes/7.0/linux-packages.md) if you install .NET manually.

## Packages

.NET is available in [official package archives](https://github.com/dotnet/core/blob/main/linux.md) for various Linux distributions and [packages.microsoft.com](https://packages.microsoft.com/).

* [Alpine](https://learn.microsoft.com/en-us/dotnet/core/install/linux-alpine)
* [CentOS](https://learn.microsoft.com/en-us/dotnet/core/install/linux-centos)
* [Debian](https://learn.microsoft.com/en-us/dotnet/core/install/linux-debian)
* [Fedora](https://learn.microsoft.com/en-us/dotnet/core/install/linux-fedora)
* [openSUSE](https://learn.microsoft.com/en-us/dotnet/core/install/linux-opensuse)
* [SLES](https://learn.microsoft.com/en-us/dotnet/core/install/linux-sles)
* [Snap](https://learn.microsoft.com/en-us/dotnet/core/install/linux-snap)
* [Ubuntu](https://learn.microsoft.com/en-us/dotnet/core/install/linux-ubuntu)

.NET is [supported by Microsoft](https://github.com/dotnet/core/blob/main/microsoft-support.md) when downloaded from a Microsoft source. Best effort support is offered from Microsoft when downloaded from elsewhere. You can open issues at [dotnet/core](https://github.com/dotnet/core) if you run into problems.

## Next steps

* [How to check if .NET is already installed](https://learn.microsoft.com/en-us/dotnet/core/install/how-to-detect-installed-versions?pivots=os-linux).
* [Tutorial: Create a new app with Visual Studio Code](https://learn.microsoft.com/en-us/dotnet/core/tutorials/with-visual-studio-code).
* [Tutorial: Containerize a .NET app](https://learn.microsoft.com/en-us/dotnet/core/docker/build-container).

## Recommended content

### [Install .NET on Linux without using a package manager - .NET](https://learn.microsoft.com/en-us/dotnet/core/install/linux-scripted-manual?source=recommendations)

Demonstrates how to install the .NET SDK and the .NET Runtime on Linux without a package manager. Use the install script or manually extract the binaries.

### [Install .NET on Linux with Snap - .NET](https://learn.microsoft.com/en-us/dotnet/core/install/linux-snap?source=recommendations)

Demonstrates how to install either the .NET SDK or the .NET Runtime on Linux with Snap.

### [Install .NET on Debian - .NET](https://learn.microsoft.com/en-us/dotnet/core/install/linux-debian?source=recommendations)

Demonstrates the various ways to install .NET SDK and .NET Runtime on Debian.

### [Install .NET on Fedora - .NET](https://learn.microsoft.com/en-us/dotnet/core/install/linux-fedora?source=recommendations)

Demonstrates the various ways to install .NET SDK and .NET Runtime on Fedora.

Show more

# Install .NET on Linux

* Article
* 01/10/2023
* 2 minutes to read
* 7 contributors

Feedback

Install on Linux

This article details how to install .NET on various Linux distributions either manually, via a package manager, or via a [container](https://learn.microsoft.com/en-us/dotnet/core/docker/introduction#net-images).

## Manual installation

You can install .NET manually in the following ways:

* [Manual install](https://learn.microsoft.com/en-us/dotnet/core/install/linux-scripted-manual#manual-install)
* [Scripted install](https://learn.microsoft.com/en-us/dotnet/core/install/linux-scripted-manual#scripted-install)

You may need to install [.NET dependencies](https://github.com/dotnet/core/blob/main/release-notes/7.0/linux-packages.md) if you install .NET manually.

## Packages

.NET is available in [official package archives](https://github.com/dotnet/core/blob/main/linux.md) for various Linux distributions and [packages.microsoft.com](https://packages.microsoft.com/).

* [Alpine](https://learn.microsoft.com/en-us/dotnet/core/install/linux-alpine)
* [CentOS](https://learn.microsoft.com/en-us/dotnet/core/install/linux-centos)
* [Debian](https://learn.microsoft.com/en-us/dotnet/core/install/linux-debian)
* [Fedora](https://learn.microsoft.com/en-us/dotnet/core/install/linux-fedora)
* [openSUSE](https://learn.microsoft.com/en-us/dotnet/core/install/linux-opensuse)
* [SLES](https://learn.microsoft.com/en-us/dotnet/core/install/linux-sles)
* [Snap](https://learn.microsoft.com/en-us/dotnet/core/install/linux-snap)
* [Ubuntu](https://learn.microsoft.com/en-us/dotnet/core/install/linux-ubuntu)

.NET is [supported by Microsoft](https://github.com/dotnet/core/blob/main/microsoft-support.md) when downloaded from a Microsoft source. Best effort support is offered from Microsoft when downloaded from elsewhere. You can open issues at [dotnet/core](https://github.com/dotnet/core) if you run into problems.

## Next steps

* [How to check if .NET is already installed](https://learn.microsoft.com/en-us/dotnet/core/install/how-to-detect-installed-versions?pivots=os-linux).
* [Tutorial: Create a new app with Visual Studio Code](https://learn.microsoft.com/en-us/dotnet/core/tutorials/with-visual-studio-code).
* [Tutorial: Containerize a .NET app](https://learn.microsoft.com/en-us/dotnet/core/docker/build-container).

## Recommended content

### [Install .NET on Linux without using a package manager - .NET](https://learn.microsoft.com/en-us/dotnet/core/install/linux-scripted-manual?source=recommendations)

Demonstrates how to install the .NET SDK and the .NET Runtime on Linux without a package manager. Use the install script or manually extract the binaries.

### [Install .NET on Linux with Snap - .NET](https://learn.microsoft.com/en-us/dotnet/core/install/linux-snap?source=recommendations)

Demonstrates how to install either the .NET SDK or the .NET Runtime on Linux with Snap.

### [Install .NET on Debian - .NET](https://learn.microsoft.com/en-us/dotnet/core/install/linux-debian?source=recommendations)

Demonstrates the various ways to install .NET SDK and .NET Runtime on Debian.

### [Install .NET on Fedora - .NET](https://learn.microsoft.com/en-us/dotnet/core/install/linux-fedora?source=recommendations)

Demonstrates the various ways to install .NET SDK and .NET Runtime on Fedora.

Show more

# Install the .NET SDK or the .NET Runtime on Ubuntu

* Article
* 01/10/2023
* 13 minutes to read
* 12 contributors

Feedback

.NET is supported on Ubuntu. This article describes how to install .NET on Ubuntu. When an [Ubuntu version](https://wiki.ubuntu.com/Releases) falls out of support, .NET is no longer supported with that version.

Install the SDK (which includes the runtime) if you want to develop .NET apps. Or, if you only need to run apps, install the Runtime. If you're installing the Runtime, we suggest you install the **ASP.NET Core Runtime** as it includes both .NET and ASP.NET Core runtimes.

Use the dotnet --list-sdks and dotnet --list-runtimes commands to see which versions are installed. For more information, see [How to check that .NET is already installed](https://learn.microsoft.com/en-us/dotnet/core/install/how-to-detect-installed-versions).

**Important**

Package manager installs are only supported on the **x64** architecture. Other architectures, such as **Arm**, must install .NET by some other means such as with Snap, an installer script, or through a manual binary installation.

For more information on installing .NET **without a package manager**, see one of the following articles:

* [Alternatively install .NET with Snap.](https://learn.microsoft.com/en-us/dotnet/core/install/linux-snap)
* [Alternatively install .NET with install-dotnet script.](https://learn.microsoft.com/en-us/dotnet/core/install/linux-scripted-manual#scripted-install)
* [Manually install .NET](https://learn.microsoft.com/en-us/dotnet/core/install/linux-scripted-manual#manual-install)

## Supported distributions

The following table is a list of currently supported .NET releases and the versions of Ubuntu they're supported on.

| **Ubuntu** | **.NET** |
| --- | --- |
| [22.10](https://learn.microsoft.com/en-us/dotnet/core/install/linux-ubuntu#2210) | 7, 6 |
| [22.04 (LTS)](https://learn.microsoft.com/en-us/dotnet/core/install/linux-ubuntu#2204) | 7, 6 |
| [20.04 (LTS)](https://learn.microsoft.com/en-us/dotnet/core/install/linux-ubuntu#2004) | 7, 6 |
| [18.04 (LTS)](https://learn.microsoft.com/en-us/dotnet/core/install/linux-ubuntu#1804) | 7, 6 |
| [16.04 (LTS)](https://learn.microsoft.com/en-us/dotnet/core/install/linux-ubuntu#1604) | 6 |

The following versions of .NET are ❌ no longer supported:

* .NET 5
* .NET Core 3.1
* .NET Core 3.0
* .NET Core 2.2
* .NET Core 2.1
* .NET Core 2.0

## Install preview versions

Preview and release candidate versions of .NET aren't available in package managers. You can install previews and release candidates of .NET in one of the following ways:

* [Snap package](https://learn.microsoft.com/en-us/dotnet/core/install/linux-snap)
* [Scripted install with install-dotnet.sh](https://learn.microsoft.com/en-us/dotnet/core/install/linux-scripted-manual#scripted-install)
* [Manual binary extraction](https://learn.microsoft.com/en-us/dotnet/core/install/linux-scripted-manual#manual-install)

## Remove preview versions

When using a package manager to manage your installation of .NET, you may run into a conflict if you've previously installed a preview release. The package manager may interpret the non-preview release as an earlier version of .NET. To install the non-preview release, first uninstall the preview versions. For more information about uninstalling .NET, see [How to remove the .NET Runtime and SDK](https://learn.microsoft.com/en-us/dotnet/core/install/remove-runtime-sdk-versions?pivots=os-linux#uninstall-net).

## 22.10

**Important**

.NET 7 isn't yet ready in the Ubuntu feed, and is only available via the Microsoft feeds. However, .NET 6 is available in the 22.10 Ubuntu feed. These instructions demonstrate how to install .NET 7 via the Microsoft package manager feed.

Installing with APT can be done with a few commands. Before you install .NET, run the following commands to add the Microsoft package signing key to your list of trusted keys and add the package repository.

Open a terminal and run the following commands:

BashCopy

wget https://packages.microsoft.com/config/ubuntu/22.10/packages-microsoft-prod.deb -O packages-microsoft-prod.deb

sudo dpkg -i packages-microsoft-prod.deb

rm packages-microsoft-prod.deb

### Install the SDK

The .NET SDK allows you to develop apps with .NET. If you install the .NET SDK, you don't need to install the corresponding runtime. To install the .NET SDK, run the following commands:

BashCopy

sudo apt-get update && \

sudo apt-get install -y dotnet-sdk-7.0

**Important**

If you receive an error message similar to **Unable to locate package dotnet-sdk-7.0**, see the [**troubleshooting**](https://learn.microsoft.com/en-us/dotnet/core/install/linux-ubuntu#troubleshooting) section.

### Install the runtime

The ASP.NET Core Runtime allows you to run apps that were made with .NET that didn't provide the runtime. The following commands install the ASP.NET Core Runtime, which is the most compatible runtime for .NET. In your terminal, run the following commands:

BashCopy

sudo apt-get update && \

sudo apt-get install -y aspnetcore-runtime-7.0

**Important**

If you receive an error message similar to **Unable to locate package aspnetcore-runtime-7.0**, see the [**troubleshooting**](https://learn.microsoft.com/en-us/dotnet/core/install/linux-ubuntu#troubleshooting) section.

As an alternative to the ASP.NET Core Runtime, you can install the .NET Runtime, which doesn't include ASP.NET Core support: replace aspnetcore-runtime-7.0 in the previous command with dotnet-runtime-7.0:

BashCopy

sudo apt-get install -y dotnet-runtime-7.0

**Note**

[**Ubuntu 22.10 includes OpenSSL 3**](https://discourse.ubuntu.com/t/openssl-3-0-transition-plans/24453) as the baseline version. Versions of .NET prior to .NET 6 don't support OpenSSL 3. Microsoft doesn't test or support using OpenSSL 1.x on Ubuntu 22.10. For more information, see [**.NET 6 Security Improvements**](https://devblogs.microsoft.com/dotnet/announcing-net-6/#security).

## 22.04

**Warning**

If you've previously installed .NET from packages.microsoft.com, you may run into issues swapping to the built in Ubuntu package manager feeds for .NET. For more information, see the [**Advisory on installing .NET on Ubuntu**](https://github.com/dotnet/core/issues/7699) and [**Troubleshoot .NET package mixups**](https://learn.microsoft.com/en-us/dotnet/core/install/linux-package-mixup#whats-going-on).

.NET 6 is included in the Ubuntu 22.04 package manager feeds.

**Important**

.NET 7 **isn't** included in the Ubuntu feeds and you must use the [**22.04 Microsoft package feed**](https://learn.microsoft.com/en-us/dotnet/core/install/linux-ubuntu#2204-microsoft-package-feed).

### Install the SDK

The .NET SDK allows you to develop apps with .NET. If you install the .NET SDK, you don't need to install the corresponding runtime. To install the .NET SDK, run the following commands:

BashCopy

sudo apt-get update && \

sudo apt-get install -y dotnet-sdk-6.0

### Install the runtime

The ASP.NET Core Runtime allows you to run apps that were made with .NET that didn't provide the runtime. The following commands install the ASP.NET Core Runtime, which is the most compatible runtime for .NET. In your terminal, run the following commands:

BashCopy

sudo apt-get update && \

sudo apt-get install -y aspnetcore-runtime-6.0

As an alternative to the ASP.NET Core Runtime, you can install the .NET Runtime, which doesn't include ASP.NET Core support: replace aspnetcore-runtime-6.0 in the previous command with dotnet-runtime-6.0:

BashCopy

sudo apt-get install -y dotnet-runtime-6.0

## 22.04 (Microsoft package feed)

**Important**

.NET 6 is included in the Ubuntu 22.04 package manager feeds, but .NET 7 isn't. To install .NET 7 you must use the Microsoft package feed. If you've previously installed .NET from the Ubuntu package manager feed, you may run into issues swapping to the Microsoft package manager feed for .NET. For more information, see the [**Advisory on installing .NET on Ubuntu**](https://github.com/dotnet/core/issues/7699) and [**Troubleshoot .NET package mixups**](https://learn.microsoft.com/en-us/dotnet/core/install/linux-package-mixup#whats-going-on).

Installing with APT can be done with a few commands. Before you install .NET, run the following commands to add the Microsoft package signing key to your list of trusted keys and add the package repository.

Open a terminal and run the following commands:

BashCopy

wget https://packages.microsoft.com/config/ubuntu/22.04/packages-microsoft-prod.deb -O packages-microsoft-prod.deb

sudo dpkg -i packages-microsoft-prod.deb

rm packages-microsoft-prod.deb

### Install the SDK

The .NET SDK allows you to develop apps with .NET. If you install the .NET SDK, you don't need to install the corresponding runtime. To install the .NET SDK, run the following commands:

BashCopy

sudo apt-get update && \

sudo apt-get install -y dotnet-sdk-7.0

**Important**

If you receive an error message similar to **Unable to locate package dotnet-sdk-7.0**, see the [**troubleshooting**](https://learn.microsoft.com/en-us/dotnet/core/install/linux-ubuntu#troubleshooting) section.

### Install the runtime

The ASP.NET Core Runtime allows you to run apps that were made with .NET that didn't provide the runtime. The following commands install the ASP.NET Core Runtime, which is the most compatible runtime for .NET. In your terminal, run the following commands:

BashCopy

sudo apt-get update && \

sudo apt-get install -y aspnetcore-runtime-7.0

**Important**

If you receive an error message similar to **Unable to locate package aspnetcore-runtime-7.0**, see the [**troubleshooting**](https://learn.microsoft.com/en-us/dotnet/core/install/linux-ubuntu#troubleshooting) section.

As an alternative to the ASP.NET Core Runtime, you can install the .NET Runtime, which doesn't include ASP.NET Core support: replace aspnetcore-runtime-7.0 in the previous command with dotnet-runtime-7.0:

BashCopy

sudo apt-get install -y dotnet-runtime-7.0

**Note**

[**Ubuntu 22.04 includes OpenSSL 3**](https://discourse.ubuntu.com/t/openssl-3-0-transition-plans/24453) as the baseline version. Versions of .NET prior to .NET 6 don't support OpenSSL 3. Microsoft doesn't test or support using OpenSSL 1.x on Ubuntu 22.10. For more information, see [**.NET 6 Security Improvements**](https://devblogs.microsoft.com/dotnet/announcing-net-6/#security).

## 20.04

Installing with APT can be done with a few commands. Before you install .NET, run the following commands to add the Microsoft package signing key to your list of trusted keys and add the package repository.

Open a terminal and run the following commands:

BashCopy

wget https://packages.microsoft.com/config/ubuntu/20.04/packages-microsoft-prod.deb -O packages-microsoft-prod.deb

sudo dpkg -i packages-microsoft-prod.deb

rm packages-microsoft-prod.deb

### Install the SDK

The .NET SDK allows you to develop apps with .NET. If you install the .NET SDK, you don't need to install the corresponding runtime. To install the .NET SDK, run the following commands:

BashCopy

sudo apt-get update && \

sudo apt-get install -y dotnet-sdk-6.0

**Important**

If you receive an error message similar to **Unable to locate package dotnet-sdk-6.0**, see the [**troubleshooting**](https://learn.microsoft.com/en-us/dotnet/core/install/linux-ubuntu#troubleshooting) section.

### Install the runtime

The ASP.NET Core Runtime allows you to run apps that were made with .NET that didn't provide the runtime. The following commands install the ASP.NET Core Runtime, which is the most compatible runtime for .NET. In your terminal, run the following commands:

BashCopy

sudo apt-get update && \

sudo apt-get install -y aspnetcore-runtime-6.0

**Important**

If you receive an error message similar to **Unable to locate package aspnetcore-runtime-6.0**, see the [**troubleshooting**](https://learn.microsoft.com/en-us/dotnet/core/install/linux-ubuntu#troubleshooting) section.

As an alternative to the ASP.NET Core Runtime, you can install the .NET Runtime, which doesn't include ASP.NET Core support: replace aspnetcore-runtime-6.0 in the previous command with dotnet-runtime-6.0:

BashCopy

sudo apt-get install -y dotnet-runtime-6.0

## 18.04

Installing with APT can be done with a few commands. Before you install .NET, run the following commands to add the Microsoft package signing key to your list of trusted keys and add the package repository.

Open a terminal and run the following commands:

BashCopy

wget https://packages.microsoft.com/config/ubuntu/18.04/packages-microsoft-prod.deb -O packages-microsoft-prod.deb

sudo dpkg -i packages-microsoft-prod.deb

rm packages-microsoft-prod.deb

### Install the SDK

The .NET SDK allows you to develop apps with .NET. If you install the .NET SDK, you don't need to install the corresponding runtime. To install the .NET SDK, run the following commands:

BashCopy

sudo apt-get update && \

sudo apt-get install -y dotnet-sdk-6.0

**Important**

If you receive an error message similar to **Unable to locate package dotnet-sdk-6.0**, see the [**troubleshooting**](https://learn.microsoft.com/en-us/dotnet/core/install/linux-ubuntu#troubleshooting) section.

### Install the runtime

The ASP.NET Core Runtime allows you to run apps that were made with .NET that didn't provide the runtime. The following commands install the ASP.NET Core Runtime, which is the most compatible runtime for .NET. In your terminal, run the following commands:

BashCopy

sudo apt-get update && \

sudo apt-get install -y aspnetcore-runtime-6.0

**Important**

If you receive an error message similar to **Unable to locate package aspnetcore-runtime-6.0**, see the [**troubleshooting**](https://learn.microsoft.com/en-us/dotnet/core/install/linux-ubuntu#troubleshooting) section.

As an alternative to the ASP.NET Core Runtime, you can install the .NET Runtime, which doesn't include ASP.NET Core support: replace aspnetcore-runtime-6.0 in the previous command with dotnet-runtime-6.0:

BashCopy

sudo apt-get install -y dotnet-runtime-6.0

## 16.04

Installing with APT can be done with a few commands. Before you install .NET, run the following commands to add the Microsoft package signing key to your list of trusted keys and add the package repository.

Open a terminal and run the following commands:

BashCopy

wget https://packages.microsoft.com/config/ubuntu/16.04/packages-microsoft-prod.deb -O packages-microsoft-prod.deb

sudo dpkg -i packages-microsoft-prod.deb

rm packages-microsoft-prod.deb

### Install the SDK

The .NET SDK allows you to develop apps with .NET. If you install the .NET SDK, you don't need to install the corresponding runtime. To install the .NET SDK, run the following commands:

BashCopy

sudo apt-get update && \

sudo apt-get install -y dotnet-sdk-6.0

**Important**

If you receive an error message similar to **Unable to locate package dotnet-sdk-6.0**, see the [**troubleshooting**](https://learn.microsoft.com/en-us/dotnet/core/install/linux-ubuntu#troubleshooting) section.

### Install the runtime

The ASP.NET Core Runtime allows you to run apps that were made with .NET that didn't provide the runtime. The following commands install the ASP.NET Core Runtime, which is the most compatible runtime for .NET. In your terminal, run the following commands:

BashCopy

sudo apt-get update && \

sudo apt-get install -y aspnetcore-runtime-6.0

**Important**

If you receive an error message similar to **Unable to locate package aspnetcore-runtime-6.0**, see the [**troubleshooting**](https://learn.microsoft.com/en-us/dotnet/core/install/linux-ubuntu#troubleshooting) section.

As an alternative to the ASP.NET Core Runtime, you can install the .NET Runtime, which doesn't include ASP.NET Core support: replace aspnetcore-runtime-6.0 in the previous command with dotnet-runtime-6.0:

BashCopy

sudo apt-get install -y dotnet-runtime-6.0

## How to install other versions

All versions of .NET are available for download at <https://dotnet.microsoft.com/download/dotnet>, but require [manual installation](https://learn.microsoft.com/en-us/dotnet/core/install/linux-scripted-manual). You can try and use the package manager to install a different version of .NET. However, the requested version may not be available.

The packages added to package manager feeds are named in a hackable format, for example: {product}-{type}-{version}.

* **product**  
  The type of .NET product to install. Valid options are:
  + dotnet
  + aspnetcore
* **type**  
  Chooses the SDK or the runtime. Valid options are:
  + sdk
  + runtime
* **version**  
  The version of the SDK or runtime to install. This article will always give the instructions for the latest supported version. Valid options are any released version, such as:
  + 7.0
  + 5.0
  + 3.1
  + 2.1

It's possible the SDK/runtime you're trying to download is not available for your Linux distribution. For a list of supported distributions, see [Install .NET on Linux](https://learn.microsoft.com/en-us/dotnet/core/install/linux).

### Examples

* Install the ASP.NET Core 7.0 runtime: aspnetcore-runtime-7.0
* Install the .NET Core 2.1 runtime: dotnet-runtime-2.1
* Install the .NET 5 SDK: dotnet-sdk-5.0
* Install the .NET Core 3.1 SDK: dotnet-sdk-3.1

### Package missing

If the package-version combination doesn't work, it's not available. For example, there isn't an ASP.NET Core SDK, the SDK components are included with the .NET SDK. The value aspnetcore-sdk-7.0 is incorrect and should be dotnet-sdk-7.0. For a list of Linux distributions supported by .NET, see [.NET dependencies and requirements](https://learn.microsoft.com/en-us/dotnet/core/install/linux).

## Use APT to update .NET

When a new patch release is available for .NET, you can simply upgrade it through APT with the following commands:

BashCopy

sudo apt-get update

sudo apt-get upgrade

If you've upgraded your Linux distribution since installing .NET, you may need to reconfigure the Microsoft package repository. Run the installation instructions for your current distribution version to upgrade to the appropriate package repository for .NET updates.

## Troubleshooting

Starting with Ubuntu 22.04 you may run into a situation where it seems only a piece of .NET is available. For example, when you've installed the runtime and the SDK, but when running dotnet --info the SDK isn't listed. This can be related to using two different package sources. The official Ubuntu 22.04 and Ubuntu 22.10 package feeds include .NET, but you may have also installed .NET from the Microsoft feeds. For more information about how to fix this problem, see [Troubleshoot fxr, libhostfxr.so, and FrameworkList.xml errors](https://learn.microsoft.com/en-us/dotnet/core/install/linux-package-mixup)

### APT problems

This section provides information on common errors you may get while using APT to install .NET.

#### Unable to find package

**Important**

Package manager installs are only supported on the **x64** architecture. Other architectures, such as **Arm**, must install .NET by some other means such as with Snap, an installer script, or through a manual binary installation.

For more information on installing .NET **without a package manager**, see one of the following articles:

* [Alternatively install .NET with Snap.](https://learn.microsoft.com/en-us/dotnet/core/install/linux-snap)
* [Alternatively install .NET with install-dotnet script.](https://learn.microsoft.com/en-us/dotnet/core/install/linux-scripted-manual#scripted-install)
* [Manually install .NET](https://learn.microsoft.com/en-us/dotnet/core/install/linux-scripted-manual#manual-install)

#### Unable to locate \ Some packages could not be installed

**Note**

This information only applies when .NET is installed from the Microsoft package feed.

If you receive an error message similar to **Unable to locate package {dotnet-package}** or **Some packages could not be installed**, run the following commands.

There are two placeholders in the following set of commands.

* {dotnet-package}  
  This represents the .NET package you're installing, such as aspnetcore-runtime-7.0. This is used in the following sudo apt-get install command.
* {os-version}  
  This represents the distribution version you're on. This is used in the wget command below. The distribution version is the numerical value, such as 20.04 on Ubuntu or 10 on Debian.

First, try purging the package list:

BashCopy

sudo dpkg --purge packages-microsoft-prod && sudo dpkg -i packages-microsoft-prod.deb

sudo apt-get update

Then, try to install .NET again. If that doesn't work, you can run a manual install with the following commands:

BashCopy

sudo apt-get install -y gpg

wget -O - https://packages.microsoft.com/keys/microsoft.asc | gpg --dearmor -o microsoft.asc.gpg

sudo mv microsoft.asc.gpg /etc/apt/trusted.gpg.d/

wget https://packages.microsoft.com/config/ubuntu/{os-version}/prod.list

sudo mv prod.list /etc/apt/sources.list.d/microsoft-prod.list

sudo chown root:root /etc/apt/trusted.gpg.d/microsoft.asc.gpg

sudo chown root:root /etc/apt/sources.list.d/microsoft-prod.list

sudo apt-get update && \

sudo apt-get install -y {dotnet-package}

#### Failed to fetch

While installing the .NET package, you may see an error similar to Failed to fetch ... File has unexpected size ... Mirror sync in progress?. This error could mean that the package feed for .NET is being upgraded with newer package versions, and that you should try again later. During an upgrade, the package feed shouldn't be unavailable for more than 30 minutes. If you continually receive this error for more than 30 minutes, please file an issue at <https://github.com/dotnet/core/issues>.

## Dependencies

When you install with a package manager, these libraries are installed for you. But, if you manually install .NET or you publish a self-contained app, you'll need to make sure these libraries are installed:

* libc6
* libgcc1
* libgcc-s1 (for 22.x)
* libgssapi-krb5-2
* libicu55 (for 16.x)
* libicu60 (for 18.x)
* libicu66 (for 20.x)
* libicu70 (for 22.04)
* libicu71 (for 22.10)
* liblttng-ust1 (for 22.x)
* libssl1.0.0 (for 16.x)
* libssl1.1 (for 18.x, 20.x)
* libssl3 (for 22.x)
* libstdc++6
* libunwind8 (for 22.x)
* zlib1g

If the .NET app uses the System.Drawing.Common assembly, libgdiplus will also need to be installed. Because [System.Drawing.Common is no longer supported on Linux](https://learn.microsoft.com/en-us/dotnet/core/compatibility/core-libraries/6.0/system-drawing-common-windows-only), this only works on .NET 6 and requires setting the System.Drawing.EnableUnixSupport runtime configuration switch.

You can install a recent version of libgdiplus by [adding the Mono repository to your system](https://www.mono-project.com/download/stable/#download-lin-ubuntu).

## Next steps

* [How to enable TAB completion for the .NET CLI](https://learn.microsoft.com/en-us/dotnet/core/tools/enable-tab-autocomplete)
* [Tutorial: Create a console application with .NET SDK using Visual Studio Code](https://learn.microsoft.com/en-us/dotnet/core/tutorials/with-visual-studio-code)

## Recommended content

### [Install .NET on Linux distributions - .NET](https://learn.microsoft.com/en-us/dotnet/core/install/linux?source=recommendations)

Learn about how to install .NET on Linux. .NET is not only available at package.microsoft.com, but also the official package archives for various Linux distributions.

### [Install .NET on Linux without using a package manager - .NET](https://learn.microsoft.com/en-us/dotnet/core/install/linux-scripted-manual?source=recommendations)

Demonstrates how to install the .NET SDK and the .NET Runtime on Linux without a package manager. Use the install script or manually extract the binaries.

### [Install .NET on Linux with Snap - .NET](https://learn.microsoft.com/en-us/dotnet/core/install/linux-snap?source=recommendations)

Demonstrates how to install either the .NET SDK or the .NET Runtime on Linux with Snap.

### [Install .NET on Debian - .NET](https://learn.microsoft.com/en-us/dotnet/core/install/linux-debian?source=recommendations)

Demonstrates the various ways to install .NET SDK and .NET Runtime on Debian.

Show more

## Feedback

Submit and view feedback for

[**This product**](https://aka.ms/feedback/report?space=61) [**This page**](https://github.com/dotnet/docs/issues/new?title=&body=%0A%0A%5BEnter%20feedback%20here%5D%0A%0A%0A---%0A%23%23%23%23%20Document%20Details%0A%0A%E2%9A%A0%20*Do%20not%20edit%20this%20section.%20It%20is%20required%20for%20learn.microsoft.com%20%E2%9E%9F%20GitHub%20issue%20linking.*%0A%0A*%20ID%3A%2080707602-f283-e796-d3b8-3287fbc011dc%0A*%20Version%20Independent%20ID%3A%201d732b15-9e64-633a-000c-ad96d9d3818a%0A*%20Content%3A%20%5BInstall%20.NET%20on%20Ubuntu%20-%20.NET%5D(https%3A%2F%2Flearn.microsoft.com%2Fen-us%2Fdotnet%2Fcore%2Finstall%2Flinux-ubuntu)%0A*%20Content%20Source%3A%20%5Bdocs%2Fcore%2Finstall%2Flinux-ubuntu.md%5D(https%3A%2F%2Fgithub.com%2Fdotnet%2Fdocs%2Fblob%2Fmain%2Fdocs%2Fcore%2Finstall%2Flinux-ubuntu.md)%0A*%20Product%3A%20**.net**%0A*%20Technology%3A%20**dotnet-installation**%0A*%20GitHub%20Login%3A%20%40adegeo%0A*%20Microsoft%20Alias%3A%20**adegeo**)

[View all page feedback](https://github.com/dotnet/docs/issues?utf8=%E2%9C%93&q=%221d732b15-9e64-633a-000c-ad96d9d3818a%22&in=body)

In this article

* [Supported distributions](https://learn.microsoft.com/en-us/dotnet/core/install/linux-ubuntu#supported-distributions)
* [Install preview versions](https://learn.microsoft.com/en-us/dotnet/core/install/linux-ubuntu#install-preview-versions)
* [Remove preview versions](https://learn.microsoft.com/en-us/dotnet/core/install/linux-ubuntu#remove-preview-versions)
* [22.10](https://learn.microsoft.com/en-us/dotnet/core/install/linux-ubuntu#2210)

Show more

[**English (United States)**](https://learn.microsoft.com/en-us/locale?target=https://learn.microsoft.com/en-us/dotnet/core/install/linux-ubuntu)

Theme