

The results are in! See what nearly 90,000 developers picked as their most loved, dreaded, and desired coding languages and more in the 2019 Developer Survey.

[Home](#)[PUBLIC](#)[Stack Overflow](#)[Tags](#)[Users](#)[Jobs](#)**Teams**

Q&A for work

[Learn More](#)

How to get the URL of the current page in C# [duplicate]

[Ask Question](#)

604

This question already has an answer here:

[Get URL of ASP.Net Page in code-behind \[duplicate\]](#) 10 answers



Can anyone help out me in getting the URL of the current working page of ASP.NET in C#?



150

c#

asp.net

edited Nov 3 '11 at 21:40

[Mark Byers](#)

600k 129 1371
1349

asked Feb 27 '09 at 6:20


apekshabs


marked as duplicate by [Gordon](#) Aug 3 '13 at 7:51

This question has been asked before and already has an answer. If those answers do not fully address your question, please [ask a new question](#).

11 @Gordon So which is it? You marked this question as being the duplicate... OF A QUESTION THAT YOU MARKED AS BEING A

DUPLICATE TOO, ONE MINUTE LATER. Maybe we need a feature request to get StackOverflow to add "no follow" to marked duplicates, because all I wanted was an answer and the search engine brought me here first. I won't make such a request however, lest it be marked duplicate. :(– [Mark Allen](#) Feb 12 '14 at 0:57

- 2 @MarkAllen not sure why you are yelling at me? if you follow the dupes, you'll end up at the oldest question (i could find then) asking this. Which of it is it? Well, all of them. It doesn't matter where you ended up searching. This page gives you 14 answers, the closed against one gives you 15 and the original one gives you another 11. If I wouldn't close them questions, you'd only get 14, not 40. Note that you have less than 10k rep so you dont see the deleted ones. – [Gordon](#) Feb 12 '14 at 7:00 
- 8 @Gordon Sorry if the caps offended you. It would be nice if dupes could be exposed in a single merged page. – [Mark Allen](#) Feb 12 '14 at 20:45
- 4 Seems to me that the primary should be the one which is either the best asked (highest vote) or the one with the best answer. Often that will be the oldest but not always. Maybe a conversion for meta... – [Liath](#) Jan 8 '15 at 16:04

I agree - it would be nice if there were a way to merge duplicate questions into a single question... – [Taraz](#) Jul 24 '18 at 18:43 

9 Answers



You may at times need to get different values from URL.

437

Below example shows different ways of extracting different parts of URL



EXAMPLE: (Sample URL)

```
http://localhost:60527/WebSite1test/Default2.aspx?  
QueryString1=1&QueryString2=2
```

CODE

```
Response.Write("<br/> " + HttpContext.Current.Request.Url.Host);  
Response.Write("<br/> " + HttpContext.Current.Request.Url.Author  
Response.Write("<br/> " + HttpContext.Current.Request.Url.Port);  
Response.Write("<br/> " + HttpContext.Current.Request.Url.Absolu  
Response.Write("<br/> " + HttpContext.Current.Request.Applicatio  
Response.Write("<br/> " + HttpContext.Current.Request.Url.Absolu  
Response.Write("<br/> " + HttpContext.Current.Request.Url.PathAn
```

OUTPUT

```
localhost  
localhost:60527  
60527  
/WebSite1test/Default2.aspx  
/WebSite1test  
http://localhost:60527/WebSite1test/Default2.aspx?QueryString=1  
/WebSite1test/Default2.aspx?QueryString=1&QueryString=2
```

You can copy paste above sample code & run it in asp.net web form application with different URL.

I also recommend reading ASP.Net Routing in case you may use ASP Routing then you don't need to use traditional URL with query string.

<http://msdn.microsoft.com/en-us/library/cc668201%28v=vs.100%29.aspx>

edited May 11 '16 at 10:55



mattmanser

4,977 1 31 47

answered May 22 '13 at 13:46



Learning

9,745 31 127 254



the request.rawurl will gives the content of current page it gives the exact path that you required

USE `HttpContext.Current.Request.RawUrl`

[edited Jan 8 '15 at 15:59](#)

community wiki
[2 revs, 2 users 86%](#)
[R Square](#)



a tip for people who needs the path/url in global.asax file;

If you need to run this in **global.asax > Application_Start** and you app pool mode is **integrated** then you will receive the error below:

Request is not available in this context exception in Application_Start.

In that case you need to use this:

System.Web.HttpRuntime.AppDomainAppVirtualPath

Hope will help others..

answered May 23 '13 at 15:00



[dvidmn](#)

3,840 2 34 49



If you want to get

`localhost:2806`

11

from

`http://localhost:2806/Pages/`

then use:

`HttpContext.Current.Request.Url.Authority`

answered Jan 17 '13 at 14:31



Thomas

210 8 19



I guess its enough to return absolute path..

6

`Path.GetFileName(Request.Url.AbsolutePath)`

using System.IO;

edited Dec 20 '12 at 16:36



G Jeny Ramirez

506 1 5 14

answered Sep 19 '12 at 13:36



Mayank Pathak

2,968 5 33 61

-
- 1 I don't think that would satisfy OP because it only gives the page name :(
- clamchoda Nov 5 '12 at 16:39
-



100



Just sharing as this was my solution thanks to Canavar's post.

If you have something like this:

```
"http://localhost:1234/Default.aspx?un=asdf&somethingelse=fdsa"
```

or like this:

```
"https://www.something.com/index.html?a=123&b=4567"
```

and you only want the part that a user would type in then this will work:

```
String strPathAndQuery = HttpContext.Current.Request.Url.PathAndQuery;  
String strUrl = HttpContext.Current.Request.Url.AbsoluteUri.Replace(  
    "/");
```

which would result in these:

```
"http://localhost:1234/"  
"https://www.something.com/"
```

answered Apr 26 '12 at 16:56



Soenhay

2,712 4 21 50

-
- 1 Just to state the obvious: HttpContext is located in System.Web, i.e System.Web.HttpContext. msdn: msdn.microsoft.com/en-us/library/system.web.httpcontext.aspx – Sindre Jul 6 '12 at 9:17
-
- 12 Even simpler is: Request.Url.GetLeftPart(UriPartial.Authority)
See here: msdn.microsoft.com/en-us/library/... – Rosdi Kasim Aug 9 '15 at 18:51
-



A search landed me at this page, but it wasn't quite what I was looking for. Posting here in case someone else looking for what I was lands at this page too.



There is two ways to do it if you only have a string value.

.NET way:

Same as @Canavar, but you can instantiate a new Uri Object

```
String URL = "http://localhost:1302/TESTERS/Default6.aspx";  
System.Uri uri = new System.Uri(URL);
```

which means you can use the same methods, e.g.

```
string url = uri.AbsoluteUri;  
// http://localhost:1302/TESTERS/Default6.aspx  
  
string host = uri.host  
// localhost
```

Regex way:

[Getting parts of a URL \(Regex\)](#)

edited May 23 '17 at 11:55



Community ♦

1 1

answered Aug 5 '11 at 11:13



Ben Pearson

5,185 2 25 48

-
- 1 Also worth noting that if the string can not be parsed it will throw a System.FormatException. However, the System.Uri.TryCreate method was introduced in .NET 4. – Ben Pearson Aug 5 '11 at 11:21
-

Try this :

848

```
string url = HttpContext.Current.Request.Url.AbsoluteUri;  
// http://localhost:1302/TESTERS/Default6.aspx  
  
string path = HttpContext.Current.Request.Url.AbsolutePath;  
// /TESTERS/Default6.aspx  
  
string host = HttpContext.Current.Request.Url.Host;  
// localhost
```

edited Feb 27 '09 at 13:59


answered Feb 27 '09 at 6:23



Canavar

41.3k 17 77 114

- 4 AbsolutePath appears to return "/" in .NET 3.5 (haven't tested other versions) for paths similar to twitter user accounts, such as twitter.com/#!/user. You can use the Fragment method to get anything after the pound (#). – Ben Pearson Aug 5 '11 at 12:11
- 3 Actually these are not correct when used with a rewriting mechanism like [UrlRewriter.net](#) because this will give the rewritten URL. You can then use: `Request.Url.GetLeftPart(UriPartial.Authority) + Request.RawUrl` – Rody van Sambeek Apr 20 '12 at 12:18
- what do you do if `HttpContext.Current` is null? – drzaus May 7 '12 at 18:50
- 8 @drzaus if `HttpContext.Current` is null then you are not processing a page request or you are attempting to access it before it is set. If you need additional help please start a new question. – Trisped May 23 '12 at 3:19
- 2 If you need to run this in `global.asax > Application_Start` and you app pool mode is "integrated" then you will receive "Request is not available in this context exception in `Application_Start`" error. In that case you need to use

System.Web.HttpRuntime.AppDomainAppVirtualPath – [dvdmn](#) May 23 '13 at 14:56 



if you just want the part between http:// and the first slash

44

```
string url = Request.Url.Host;
```



would return stackoverflow.com if called from this page

Here's the [complete breakdown](#)

answered Feb 27 '09 at 6:27



[roman m](#)

17k 24 96 129

12 Does not include port. Need: `HttpContext.Current.Request.Url.Port` .
– [JohnB](#) Apr 8 '12 at 0:53

2 @JohnB or rather `HttpContext.Current.Request.Url.Authority` ,
fetching `host:port` with one call... – [T_D](#) Jun 15 '17 at 14:20

As @JohnB says, `Authority` is safer than `Host` if the port is different
than 80 or 443. – [vapcguy](#) Dec 4 '18 at 16:09
