Server.MapPath("."), Server.MapPath("~"), Server.MapPath(@"\"), Server.MapPath("/"). What is the difference?

Ask Question

▲

433

▼

Can anyone explain the difference between `Server.MapPath(".")`, `Server.MapPath("~")`, `Server.MapPath(@"\")` and `Server.MapPath("/")` ?

asp.net   path   directory   mapping   filesystems

★

238

edited Dec 15 '13 at 21:15

dav_i
19.7k   13   84   115

asked Nov 9 '08 at 9:48

Manu
16.1k   25   68   82

3 Answers

▲

783

Server.MapPath specifies the relative or virtual path to map **to a physical directory**.

Teams
Q&A for work

Home

PUBLIC

🌐 **Stack Overflow**

Tags

Users

Jobs

- `Server.MapPath(".")` [1] returns the current physical directory of the file (e.g. aspx) being executed
- `Server.MapPath("..")` returns the parent directory
- `Server.MapPath("~")` returns the physical path to the root of the application
- `Server.MapPath("/")` returns the physical path to the root of the domain name (is not necessarily the same as the root of the application)

An example:

Let's say you pointed a web site application (`http://www.example.com/`) to

  `C:\Inetpub\wwwroot`

and installed your shop application (sub web as virtual directory in IIS, marked as application) in

  `D:\WebApps\shop`

For example, if you call `Server.MapPath()` in following request:

  http:*//www.example.com/shop/products/GetProduct.aspx?id=2342*

then:

- `Server.MapPath(".")` [1] returns `D:\WebApps\shop\products`
- `Server.MapPath("..")` returns `D:\WebApps\shop`
- `Server.MapPath("~")` returns `D:\WebApps\shop`
- `Server.MapPath("/")` returns `C:\Inetpub\wwwroot`
- `Server.MapPath("/shop")` returns `D:\WebApps\shop`

If Path starts with either a forward slash (/) or backward slash (\ ), the `MapPath()` returns a path as if Path was a full, virtual path.

If Path doesn't start with a slash, the `MapPath()` returns a path relative to the directory of the request being processed.

Note: in C#, `@` is the verbatim literal string operator meaning that the string should be used "as is" and not be processed for escape sequences.

Footnotes

1. `Server.MapPath(null)` and `Server.MapPath("")` will produce this effect too.

edited May 23 '17 at 12:10

[icon] Community ♦
1   1

answered Nov 9 '08 at 10:01

[icon] splattne
84.3k   45   195   242

---

44   Excellent. We've been battling with Server.Bloody.MapPath. Thanks – gbn May 10 '10 at 15:16

---

26   You will be better off using `HostingEnvironment.MapPath` as it doesn't require `HttpContext` : stackoverflow.com/q/944219/3205 – skolima Jul 25 '12 at 14:57

▲

4

1) `Server.MapPath(".")` -- Returns the "Current Physical Directory" of the file (e.g. `aspx`) being executed.

Ex. Suppose `D:\WebApplications\Collage\Departments`

2) `Server.MapPath("..")` -- Returns the "Parent Directory"

Ex. `D:\WebApplications\Collage`

3) `Server.MapPath("~")` -- Returns the "Physical Path to the Root of the Application"

Ex. `D:\WebApplications\Collage`

4) `Server.MapPath("/")` -- Returns the physical path to the root of the Domain Name

Ex. `C:\Inetpub\wwwroot`

edited Apr 28 '18 at 12:23
user9711416

answered Jan 8 '18 at 10:49
Vaibhav_Welcomes_You
67   1   1

Just to expand on @splattne's answer a little:

24

`MapPath(string virtualPath)` calls the following:

```
public string MapPath(string virtualPath)
{
    return this.MapPath(VirtualPath.CreateAllowNull(virtualPath));
}
```

`MapPath(VirtualPath virtualPath)` in turn calls `MapPath(VirtualPath virtualPath, VirtualPath baseVirtualDir, bool allowCrossAppMapping)` which contains the following:

```
//...
if (virtualPath == null)
{
    virtualPath = VirtualPath.Create(".");
```

```
  }
  //...
```

So if you call `MapPath(null)` or `MapPath("")` , you are effectively calling `MapPath(".")`

<div align="right">

answered Jul 12 '13 at 13:49

dav_i
19.7k   13   84   115

</div>

---

protected by Jeff Atwood Jun 7 '10 at 7:30

Thank you for your interest in this question. Because it has attracted low-quality or spam answers that had to be removed, posting an answer now requires 10 reputation on this site (the association bonus does not count).

Would you like to answer one of these unanswered questions instead?