The simplest formula to calculate page count?

Ask Question



I have an array and I want to divide them into page according to preset page size.

64

This is how I do:



16

```
private int CalcPagesCount()
{
    int totalPage = imagesFound.Length / PageSize;

    // add the last page, ugly
    if (imagesFound.Length % PageSize != 0) totalPage++;
    return totalPage;
}
```

I feel the calculation is not the simplest (I am poor in math), can you give one simpler calculation formula?

c# pagination



asked Oct 23 '09 at 3:29



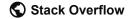
Join Stack Overflow to learn, share knowledge, and build your career.

Email Sign Up





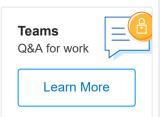




Tags

Users

Jobs





totalPage = (imagesFound.Length + PageSize - 1) / PageSize



Or use floating point math:



totalPage = (int) Math.Ceiling((double) imagesFound.Length

answered Oct 23 '09 at 3:32



John Kugelman **254k** 56 412

- 2 Danger ... this results in an overflow when PageSize = int.MaxValue . I added an answer that's not vulnerable to an overflow. Jeremy Nov 11 '17 at 15:03
- To avoid the overflow you could just refactor the formula to get: ((imagesFound.Length 1) / PageSize) + 1 AdrianRM Jan 21 at 16:15



Actually, you are close to the best you can do. About the only thing that I can think of that might be "better" is something like this:



20

totalPage = (imagesFound.Length + PageSize - 1) / PageSize

And the only reason that this is any better is that you avoid the if statement.

answered Oct 23 '09 at 3:33



Join Stack Overflow to learn, share knowledge, and build your career.

Email Sign Up

OR SIGN IN WITH



Facebook



17

NOTE: you will always get at least 1 page, even for 0 count, if the page size is > 1, which is what I needed but may not be what you need. A page size of 1(silly but technically valid) and a count of 0 would be zero pages. Depending on your needs you may want to check for a zero value for count & page size of 1

int pages = ((count - 1) / PAGESIZE) + 1;

edited May 23 at 12:11

answered Dec 5 '14 at 19:30



30 44

This seems much simpler than the accepted answer. – Greg Chabala Apr 27 '17 at 16:16

Yes, simpler. Simpler but Wrong. if count = 0 then pages = 1 -Pavel Melnikov Aug 4 '17 at 15:04

- @Pevel Melnikov . I suppose in a purely pedantic sense it is wrong, but it depends on context. In the code I used this in I always wanted at least 1 page (0 pages was invalid even if there were 0 items). I should at least add a note. - Booji Boy Aug 14 '17 at 19:46 🧪
- @PavelMelnikov, it's not exactly wrong. Just depends on how you want to look at it. The accumption that it always returns at

Join Stack Overflow to learn, share knowledge, and build your career.

Email Sign Up or sign in with



Faceboo



The OP contains a valid answer. If I wanted to turn off paging then I could set PageSize = int.MaxValue.



Several answers here add to PageSize (imagesFound.Length + PageSize) and that would cause an overflow. Which then leads to an incorrect result.

This is the code I am going to use:

```
int imageCount = imagesFound.Length;
// include this if when you always want at least 1 page
if (imageCount == 0)
    return 1;
return imageCount % PageSize != 0
    ? imageCount / PageSize + 1
    : imageCount / PageSize;
```

edited Sep 26 '18 at 21:09

answered Nov 11 '17 at 15:01



To avoid having arrors with page numbering the heat way I

Join Stack Overflow to learn, share knowledge, and build your career.







This should not give you page 2 when PageSize == imagesFound.Length

edited Apr 11 '16 at 9:06

answered May 28 '14 at 6:46



I don't think this is valid. If you have a length of 3 and a pagesize of 2, you would expect 2 pages (first page has 2 items, second page has 1). Doing this will give you (3-1)/2 = 1 page. - joshhendo Sep 1 '15 at 0:23

Altered the original post to accommodate for that aswell, thanks for noticing - Clayton C Apr 11 '16 at 9:07

Join Stack Overflow to learn, share knowledge, and build your career.

Email Sign Up

OR SIGN IN WITH



Facebook