storing state across postback

Asked 16 years, 3 months ago Modified 16 years, 3 months ago

Viewed 906 times

2

What is the best way to store string data across postback. I need to store an ID and name for multiple entities. I was thinking of using a datatable in viewstate, but would that make viewstate grow too large? I can't use a database yet because I'll be inserting a record that those other records need to be related to. So I'll just be storing them temporarily until the user submits the form.

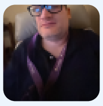
asp.net

Share

Improve this question

Follow

asked Sep 18, 2008 at 18:06



Chris Westbrook

2,040 ● 6 ● 25 ● 36

6 Answers

Sorted by: Highest score (default)

4

You actually have a lot of options - the one you choose will entirely depend on the requirements of your own application.

1. ViewState - you can add the data to the page's viewstate. The advantages of this is that the data *will*

live only for the lifetime of the user being on the page and posting it back to the server. Another advantage of it over hidden fields is that it is harder for users to hack into it and alter your values (I believe, in fact, that you can encrypt your viewstate). The disadvantage, of course, lies in page sizes - everything you add to the view state is one more thing that gets dropped on a user's page and then gets posted back to the server. This makes it non-optimal for storing large amounts of data.

2. Cookies - You can toss the information back to the user in the form of cookies. In this case, you can declare how long the information will last - for the scope of the user having their browser open, or for a specific calendar time. The information will be available to any page of your application each time the user hits that page. The bad news is that you are limited in the amount of information you can store, and that users can very readily alter their own cookies.

3. Session - You're storing the user's information on your own server's memory (I'll leave aside the discussion of various types of session storage). In this case the information will live as long as your user's session lives, and will be available to all pages of your application. There is no risk of user's modifying those values directly, though session hijacking is a risk you may want to explore. The disadvantage, though, is that you are using precious server resources in this case - if your application has

a large load, it may affect your scalability in the future.

As I said - what you choose to do will entirely depend on the needs and requirements of your application.

Share Improve this answer Follow

answered Sep 18, 2008 at 18:16

John Christensen
5,030 ● 1 ● 29 ● 26

Great answer but could I ask for clarity on one item please? I have been using `static string` to persist across postbacks, but if I have two different users filling in their unique data (i.e first name) at the same time, both their records get the same name, not their unique name. Slightly confused still as to which of the above will be local to each client and not affected by the other clients data. Cookies and Session both look good but are cookies unique to each client? are Sessions unique to each client even though they are stored in 'your servers own memory'? – FlashTrev Apr 1, 2015 at 21:16

1

several ways (though not an exhaustive list):

- ViewState
- hidden fields
- session
- query string
- cookies

Share Improve this answer

Follow

answered Sep 18, 2008 at 18:09

Danimal

7,710 ● 8 ● 48 ● 57

1

ViewState is fine. If you are storing it across postbacks, a client-side solution is best. So, you'd be adding size somewhere--either in ViewState or hidden fields.

If you want to do this server-side, you can use the Session, but remember to clean it up when you can.

Share Improve this answer

Follow

answered Sep 18, 2008 at 18:14

EndangeredMassa

17.5k ● 8 ● 56 ● 80

0

you could just store them to a cookie, this would allow you to access them from Javascript too. Alternatively you could store a simple string array to the view state. A lot depends on what and how much information you wish to store.

Share Improve this answer

Follow

answered Sep 18, 2008 at 18:09

Toby Mills

1,016 ● 2 ● 10 ● 13

0

When I have this scenario I create a structure for my fields that I stuff into Viewstate. I'm okay with having a small structure added into the page size and lifecycle considering the entire page's controls set is there already :)

Furthermore it cleans up after itself after you're done with the page, so there's no worrying about filling your Session with crap.

Share Improve this answer

Follow

answered Sep 18, 2008 at 19:07

nathaniel
1,145 ● 2 ● 8 ● 20

0

I concur with the accepted answer but I would also add that if you only want to keep track of a simple key/value collection you would be better putting a generic Dictionary into either ViewState or Session:

```
Dictionary<int, string> myValues = new Dictionary<int,
myValues.Add(1, "Apple");
maValues.Add(2, "Pear");
```

Share Improve this answer

Follow

edited May 23, 2017 at 12:19

Community Bot
1 ● 1

answered Sep 19, 2008 at 11:31

rohancragg
5,106 ● 5 ● 38 ● 47