

این	QueryString	متغیرهای Session	Server.Transform برای	مقادیر
-----	-------------	------------------	-----------------------	--------

بین	های
-----	-----

فرمهای وب ASP.NET نویسی رویدادگرای شگرفی را برای توسعه‌گران فراهم می‌کنند. این موضوع طراحی سرتاسر برنامه کاربردی شما را ساده می‌کند ولی مسائل و مشکلات خاص خود را نشان می‌دهد. برای مثال، در ASP کلاسیک شما به آسانی می‌توانید مقادیر را با استفاده از POST از یک صفحه ASP ای دیگر ارسال نمایید. اما اگر می‌خواهید در مدل فرمهای وب (یا همان مدل برنامه‌نویسی ASP.NET) نویسی کنید، همان چیز در ASP.NET ممکن نمی‌باشد. اما راههایی برای غلبه بر این وضعیت وجود دارند که می‌توانند مورد استفاده قرار بگیرند. در این مقاله موضوعات زیر را مورد بررسی قرار خواهیم داد:

- چگونگی ارسال مقادیر با استفاده از QueryString
- چگونگی استفاده از متغیرهای Session برای ارسال مقادیر
- چگونگی استفاده از مند Server.Transfer برای ارسال مقادیر

QueryString

QueryString یک مکانیسم قدیمی برای ارسال مقادیر در بین صفحات است. مزیت اصلی این متد سادگی است. اما عیب آن این است که پس از ارسال، مقادیر در نوار آدرس مرورگر قابل مشاهده می‌نمی‌توان آبجکت‌ها را از این طریق ارسال کرد. این متد مناسب‌ترین راه برای ارسال تعداد کمی از مقادیری است که نیازی به محافظت از دید دیگران ندارند. برای اعمال کردن این متد مراحل زیر را انجام دهید:

- یک فرم وب با کنترل‌هایش را درست کنید.
- یک کنترل دکمه‌ای Button یا LinkButton برای ارسال فرم به سرور بر روی فرم قرار دهید.
- در رویداد کلیک دکمه یک متغیر از نوع String تعریف کنید که URL را برای فرم دیگر که مقادیر قرار جا ارسال شوند، نگه می‌دارد.

مقادیر کنترل‌ها را در قالب پارامترهای QueryString در متغیر از نوع String قرار دهید.
Response.Redirect که از متغیر String تعریف شده استفاده می‌کند برای هدایت کاربر به صفحه دیگر استفاده نمایید.

قطعه کد زیر چگونگی انجام این مراحل را نشان می‌دهد:

```
private void Button1_Click(object sender, System.EventArgs e)
{
    string url;
    url="anotherwebform.aspx?name=" +
        TextBox1.Text + "&email=" +
        TextBox2.Text;
    Response.Redirect(url);
}
```

```
private void Page_Load(object sender, System.EventArgs e)
{
    Label1.Text=Request.QueryString["name"];
    Label2.Text=Request.QueryString["email"];
}
```

استفاده از متغیرهای Session

در این روش باید مقادیر کنترل‌ها را در متغیرهای Session ذخیره کنیم و در فرم وب دیگری به آنها دسترسی داشته باشیم. همانطور که می‌دانید ذخیره داده‌های زیاد در Session ممکن است اختلالاتی را در سرور بوجود آورد، بنابراین باید از این متد بدرستی استفاده شود. البته هر وقت که خواستید می‌توانید متغیرهای Session را از بین ببرید. مراحل اصلی برای استفاده از این متد به ترتیب زیر می :

یک فرم وب با کنترل‌هایش را درست کنید.

یک کنترل دکمه‌ای Button یا LinkButton برای ارسال فرم به سرور بر روی فرم قرار دهید.

در رویداد کلیک دکمه، متغیرهای Session را تعریف کرده و مقادیر کنترل‌ها را در آنها قرار دهید.

کاربر را با استفاده از Server.Transfer ای دیگر هدایت کنید.

در فرم وب دیگر متغیرهای Session را دریافت کرده و پس از دریافت اگر لازم باشد آنها را پاک کنید.

کد زیر این مراحل را در عمل نشان می‌دهد:

```
private void Button1_Click(object sender, System.EventArgs e)
{
    // textbox1 and textbox2 are webform controls
    Session["name"]=TextBox1.Text;
    Session["email"]=TextBox2.Text;
    Server.Transfer("anotherwebform.aspx");
}
```

```
private void Page_Load(object sender, System.EventArgs e)
{
    Label1.Text=Session["name"].ToString();
    Label2.Text=Session["email"].ToString();
    Session.Remove("name");
    Session.Remove("email");
}
```

Server.Transfer

این روش متدی پیچیده ولی روش ماهرانه‌ای برای ارسال مقادیر بین . در اینجا مقادیری را که

می‌خواهید در صفحات دیگر به آنها دسترسی داشته باشید به عنوان خصوصیات کلاس صفحه بیان

می‌کنید. در کل این متد واضحتر و شیء‌گراتر از متدهای قبلی است. مراحل زیر را برای استفاده از این متد بترتیب دنبال کنید:

یک فرم وب با کنترل‌هایش را درست کنید.

رویدادهای خصوصیت Get که مقادیر کنترل‌ها را برخواهند گرداند را تعریف کنید.

یک کنترل دکمه‌ای Button یا LinkButton برای ارسال فرم به سرور بر روی فرم قرار دهید.

در رویداد کلیک دکمه متد Server.Transfer که اجرای برنامه را به فرم تعیین شده انتقال می‌دهد فراخوانی کنید.

در فرم دوم شما می‌توانید با استفاده از خصوصیت Context.Handler به یک نمونه از فرم اول

دسترسی داشته باشید. سپس می‌توانید از خصوصیات Get که برای دسترسی به مقادیر کنترل‌ها ایجاد کرده‌ایم استفاده کنید.

کد زیر برای اجرای یک نمونه از مراحل بالا تدارک دیده شده است. خصوصیات زیر را به فرم وب اضافه کنید:

```
public string Name
{
    get
    {
        return TextBox1.Text;
    }
}
```

```
public string EMail
{
    get
    {
        return TextBox2.Text;
    }
}
```

```
}
```

Server.Transfer را فراخوانی کنید .

```
private void Button1_Click(object sender, System.EventArgs e)
{
    Server.Transfer("anotherwebform.aspx");
}
```

```
private void Page_Load
(object sender, System.EventArgs e)
{
    //create instance of source web form
    WebForm1 wf1;

    //get reference to current handler instance
    wf1=(WebForm1)Context.Handler;
    Label1.Text=wf1.Name;
    Label2.Text=wf1.Email;
}
```