

در ASP.Net، ما user-control سفارشی را جهت استفاده مجدد و مستقل در صفحات ASPX ایجاد می‌کنیم. هر user-control دارای properties عمومی، متدها و یا delegateهای خاص خود است و زمانی که user-control در یک صفحه وب جاسازی (embedded) یا فراخوانی (load) می‌شود بوسیله صفحه وب قابل استفاده است. بعد از درج user-control در صفحه وب و فراخوانی آن، ممکن است نیاز باشد مثلاً با کلیک بر روی دکمه‌ای از user-control متدی از صفحه اجرا شود. اما یک مشکل، زمانی که در حال ایجاد user-control هستید هیچ اطلاعی از صفحه‌ای که قرار است user-control در آن قرار بگیرد ندارید پس چگونه می‌توانیم به متدهای آن دسترسی داشته باشیم؟! در کلاس Delegate، متدی بنام [DynamicInvoke](#) وجود دارد که برای فراخوانی (Invoke) متد اشاره شده در delegate استفاده می‌شود. ما از این متد برای صدا زدن یک متد صفحه وبی که user-control در آن قرار دارد استفاده می‌کنیم. مثال:

```
public partial class CustomUserCtrl : System.Web.UI.UserControl
{
    private System.Delegate _delWithParam;
    private System.Delegate _delNoParam;

    // برای فراخوانی متدهایی از صفحه که دارای پارامتر هستند
    public Delegate PageMethodWithParamRef
    {
        set { _delWithParam = value; }
    }

    // برای فراخوانی متدهایی از صفحه که بدون پارامتر هستند
    public Delegate PageMethodWithNoParamRef
    {
        set { _delNoParam = value; }
    }

    protected void Page_Load(object sender, EventArgs e)
    {
    }

    protected void BtnMethodWithParam_Click(object sender, System.EventArgs e)
    {
        //Parameter to a method is being made ready
        object[] obj = new object[1];
        obj[0] = "Parameter Value" as object;
        _delWithParam.DynamicInvoke(obj);
    }

    protected void BtnMethowWithoutParam_Click(object sender, System.EventArgs e)
    {
        //Invoke a method with no parameter
        _delNoParam.DynamicInvoke();
    }
}
```

فرض کنید در user-control بالا، دو دکمه وجود دارد که متد BtnMethodWithParam_Click را به رویداد کلیک یک دکمه، و متد BtnMethowWithoutParam_Click به رویداد کلیک دکمه دیگر منتسب می‌کنیم، سپس دو عامل خصوصی (Private) را تعریف می‌کنیم و متد DynamicInvoke این عامل‌های خصوصی را در متدهای BtnMethodWithParam_Click و BtnMethowWithoutParam_Click فراخوانی می‌کنیم حال کفایت عامل‌هایی در صفحه تعریف کنیم که این عامل‌ها به متدهای مورد نظر صفحه اشاره کنند و این عامل‌های صفحه را در عامل‌های عمومی user-control قرار دهیم. در ادامه به پیاده سازی صفحه می‌پردازیم: ابتدا دو عامل تعریف می‌کنیم:

```
public partial class _Default : System.Web.UI.Page
{
    delegate void DelMethodWithParam(string strParam);
    delegate void DelMethodWithoutParam();
```

در رویداد Page_Load، یک وهله از هر کدام از عامل‌های بالا که به متد (توجه: امضاء متدها با امضاء عامل‌ها یکسان است) مورد نظر ما در صفحه اشاره می‌کند ایجاد می‌کنیم:

```
protected void Page_Load(object sender, EventArgs e)
{
    DelMethodWithParam delParam = new DelMethodWithParam(MethodWithParam);

    // عامل صفحه را به عامل عمومی تعریف شده در یوزر کنترل تخصیص می‌دهیم
    this.UserCtrl.PageMethodWithParamRef = delParam;
    DelMethodWithoutParam delNoParam = new DelMethodWithoutParam(MethodWithNoParam);

    // عامل صفحه را به عامل عمومی تعریف شده در یوزر کنترل تخصیص می‌دهیم
    this.UserCtrl.PageMethodWithNoParamRef = delNoParam;
}
```

در زیر متدهایی خصوصی که در صفحه وجود دارند و قرار است با کلیک بر روی دکمه‌های user-control فراخوانی شوند را مشاهده می‌کنید:

```
// متد دارای پارامتری که قرار است در کنترل فراخوانی شود
private void MethodWithParam(string strParam)
{
    Response.Write("It has parameter: " + strParam);
}

// متد بدون پارامتری که قرار است در کنترل فراخوانی شود
private void MethodWithNoParam()
{
    Response.Write("It has no parameter.");
}
```

و در نهایت کد پیاده سازی نهایی صفحه ما بشکل زیر خواهد شد:

```
public partial class _Default : System.Web.UI.Page
{
    delegate void DelMethodWithParam(string strParam);
    delegate void DelMethodWithoutParam();

    protected void Page_Load(object sender, EventArgs e)
    {
        DelMethodWithParam delParam = new DelMethodWithParam(MethodWithParam);

        // عامل صفحه را به عامل عمومی تعریف شده در یوزر کنترل تخصیص می‌دهیم
        this.UserCtrl.PageMethodWithParamRef = delParam;
        DelMethodWithoutParam delNoParam = new DelMethodWithoutParam(MethodWithNoParam);

        // عامل صفحه را به عامل عمومی تعریف شده در یوزر کنترل تخصیص می‌دهیم
        this.UserCtrl.PageMethodWithNoParamRef = delNoParam;
    }

    // متد دارای پارامتری که قرار است در کنترل فراخوانی شود
    private void MethodWithParam(string strParam)
    {
        Response.Write("It has parameter: " + strParam);
    }

    // متد بدون پارامتری که قرار است در کنترل فراخوانی شود
    private void MethodWithNoParam()
    {
        Response.Write("It has no parameter.");
    }
}
```

برداشتی آزاد از [این مقاله](#) .