

مقدمه و معرفی کنترل دکمه فرمان Button

[Button سایر مطالب آموزشی کنترل دکمه فرمان](#)

[Button مقدمه و معرفی کنترل دکمه فرمان](#)

[ASP.Net در Button نحوه پاسخگویی به رویدادهای دکمه فرمان](#)

[در کلاینت اسکریپت Button پاسخگویی به رویدادهای کنترل دکمه فرمان](#)

مقدمه:

یکی از کنترل های مهم و پر کاربرد در تمام محیط های برنامه نویسی و از جمله در محیط ASP.Net کنترل دکمه فرمان Button Control می باشد . از دکمه فرمان Button در ASP.Net برای ارسال یک صفحه به سرور (در اصلاح Postback کردن آن) و اجرای دستورات و توابع طراحی شده برای رویداد کلیک کنترل استفاده می شود .

هنگامی که کاربر بر روی کنترل دکمه فرمان Button کلیک کند ، صفحه مورد نظر به سرور ارسال شده و در واقع Postback می شود . در این زمان دستورات و توابع تعریف شده در رویداد Click دکمه فرمان Button و رویداد Page Load (فراخوانی صفحه) اجرا شده و نتایج به خود صفحه و یا صفحه دیگری ارسال می شود .

شکل کلی تعریف و نمایش دکمه فرمان Button در صفحه های ASP.Net به صورت زیر است :

Syntax	<pre>< asp:Button ID="Button1" runat="server" Text="Button" / ></pre> Button
--------	---

نحوه قرار دادن یک کنترل Button بر روی صفحه ASP.Net :

1. ایجاد کرده و یا صفحه از قبل طراحی شده Visual Studio در محیط ASP.Net ابتدا یک صفحه . خود را باز کنید .

2. را کشیده و روی Button یک کنترل Standard و از قسمت کنترل های Toolbox از منوی . صفحه قرار دهید
3. صفحه کد مربوط به Source همچنین می توانید به صورت مستقیم در قسمت کدنویسی . کنترل را تایپ نمایید

انواع کنترل های Button در: ASP.Net

ASP.Net شامل ۳ نوع کنترل دکمه فرمان Button می باشد که هر کدام دارای ظاهر و عملکرد متفاوتی با یکدیگر هستند :

نوع کنترل	نمایش	شرح
Button		حالت استاندارد کنترل دکمه فرمان Button می باشد . این کنترل به صورت یک تگ < Input > توسط سرور ASP.Net به مرورگر ارسال می شود
LinkButton	LinkButton	در این حالت کنترل دکمه فرمان Button به صورت یک لینک در صفحه نمایش داده می شود که در هنگام کلیک کاربر بر روی آن ، صفحه را Postback می کند .
ImageButton		در این حالت کنترل می تواند به صورت یک تصویر گرافیکی استفاده شود و یا یک تصویر گرافیکی را به آن اضافه کنید . مثلاً به وسیله آن می توانید یک دکمه Cancel طراحی کنید که یک علامت ضربدر در کنار عنوان کنترل قرار بگیرد .

در ادامه به معرفی و کاربرد هر یک از انواع این کنترل خواهیم پرداخت .
مهمترین نمونه استفاده از کنترل دکمه فرمان Button ، قرار دادن آن در فرم های ورود اطلاعات است

که کاربر می تواند پس از وارد نمودن کلیه اطلاعات لازم ، برای ارسال آنها به سرور ، روی دکمه فرمان کلیک نماید . در این حالت در اصطلاح می گویند صفحه Submit شده است .

کنترل دکمه فرمان Button و اعتبار سنجی اطلاعات در فرم های وب:

چنانچه بر روی صفحه ، کنترل های [اعتبار سنجی داده](#) وجود داشته باشد ، کلیک کردن بر روی کنترل دکمه فرمان Button باعث می شود تا ابتدا کنترل های اعتبار سنج ، عملیات چک کردن صحت اطلاعات وارد شده را انجام داده و در صورت درست بودن آنها ، صفحه را به سرور ارسال نمایند . چنانچه اطلاعات وارد شده صحیح نبوده و هر یک از کنترل های اعتبار سنج error بدهند ، صفحه به سرور ارسال نشده و پیام های لازم به کاربر برای اصلاح اطلاعات و ارسال مجدد آنها نمایش داده می شود .

در جدول به صورت مختصر به ۲ خاصیت کنترل دکمه فرمان Button اشاره می کنیم که به شما امکان می دهد بتوانید تنظیمات مربوط به چک کردن اطلاعات موجود در را توسط این کنترل انجام دهید :

خاصیت	شرح
Cause Validation	این خاصیت تعیین می کند که آیا هنگام کلیک بر روی کنترل دکمه فرمان Button و Postback شدن صفحه ، عملیات چک کردن اطلاعات فرم های موجود در صفحه انجام شود یا خیر . حالت پیش فرض این خاصیت روی مقدار True تنظیم شده و در این حالت عملیات کنترل اطلاعات انجام خواهد شد . در صورت تعیین مقدار False ، صفحه بدون کنترل اطلاعات فرم ها به سرور ارسال خواهد شد .
ValidationGroup	این خاصیت گروه کنترل های اعتبار سنجی داده ایی که باید در هنگام Postback شدن صفحه ، عملیات کنترل اطلاعات را انجام دهند را تعیین می کند . شما می توانید تعیین کنید تا فقط دسته خاصی از اطلاعات توسط کنترل های اعتبار سنج مربوط به آنها چک شوند . در صورت عدم تعیین گروه کنترل خاصی ، تمام کنترل های موجود در صفحه فراخوانی خواهند شد .

انواع حالت های Postback شدن صفحه:

در بخش قبل اشاره کردیم که هنگام کلیک کردن بر روی کنترل دکمه فرمان Button صفحه به سرور ارسال می شود . در حالت پیش فرض صفحه پس از Postback شدن به خودش بر می گردد . در این حالت چنانچه دستورات و یا توابع خاصی در رویداد کلیک کنترل دکمه فرمان Button و یا رویداد Page Load صفحه تعریف شده باشد ، اجرا خواهند شد . اما می توان در صورت نیاز ترتیبی اتخاذ کرد تا صفحه پس از Postback شدن به صفحه دیگری برود . در ادامه به آموزش این روش خواهیم پرداخت .

مدیریت رویداد های Client-Side برای کنترل دکمه فرمان: Button

کنترل کنترل دکمه فرمان Button می تواند رویدادهای Server-Side و یا Client-Side را اجرا نماید . رویدادهای Client-Side پس از ارسال صفحه و Postback شدن آن اجرا خواهند شد . اما رویدادهای Server-Side قبل از ارسال صفحه به سرور و توسط مرورگر کامپیوتر کاربر اجرا خواهند شد . یکی از پر کاربردترین زبان های برنامه نویسی Client-Side برای رویدادهای Client-Side زبان [Java Script](#) می باشد . با اضافه کردن رویدادهای Client-Side به کنترل دکمه فرمان Button ، می توان قبل از ارسال اطلاعات به سرور و Postback شدن صفحه ، پیام های هشدار ، اطلاعات تکمیلی و یا دریافت تاییدیه از کاربر را نمایش داد ، تا پس از تایید موارد فوق ، اطلاعات به سرور ارسال شوند . در ادامه به آموزش این روش نیز خواهیم پرداخت .

نحوه پاسخگویی به رویدادهای دکمه فرمان Button در ASP.Net

زمانی که بر روی یک دکمه فرمان Button از نوع معمولی ، یک دکمه فرمان لینکی LinkButton و یا یک دکمه فرمان تصویری ImageButton کلیک می کنید ، صفحه به سرور (محلی که قرار است ، اطلاعات آن پردازش شود) ارسال یا در اصطلاح Post می شود . این ارسال یا پست شدن صفحه ، باعث اجرا و لود مجدد صفحه شده و شما می توانید کدهای مورد نظر خود را برای پس از این عملیات ، اجرا نمایید . به وسیله طراحی یک مدیریت کننده رویداد (event handler) ، به یکی از روش های زیر ، می توانید این رویداد دکمه فرمان را مدیریت نمایید :

۱) **رویداد بارگذاری صفحه یا Page_Load** : هر بار که صفحه به وسیله یک دکمه فرمان به سرور ارسال می شود ، مجددا فراخوانی و بارگذاری شده و رویداد Page_Load آن اجرا می شود . از این رویداد می توانید برای قرار دادن کدهایی استفاده نمایید که مهم نیست ، پس از کلیک بر روی چه دکمه فرمانی ، می خواهید اجرا شوند . فقط مهم این است که صفحه ارسال یا Submit شده است .

مثال عملی ۱ : در مثال زیر کدی نوشته ایم که به کلیک بر روی دکمه فرمان های صفحه ، در رویداد Page_Load آن واکنش نشان می دهد . این کد ابتدا به وسیله خاصیت IsPostBack ، تست می کند که آیا صفحه برای اولین بار است اجرا شده و یا اینکه از پردازش سرور بازگشته است . سپس کدهای لازم را اجرا می کند :

```
private void Page_Load(object sender, System.EventArgs e)
{
    if (!IsPostBack)
    {
        این کد زمانی اجرا می شود ، که صفحه برای اولین بار اجرا می شود '
        کاربر هنوز بر روی هیچ دکمه فرمانی کلیک نکرده است '
    }
    else
    {
        این کد زمانی اتفاق می افتد که صفحه از سرور Postback شده و برگشته است '
        کاربر بر روی یکی از دکمه فرمان های فعال صفحه ، کلیک نموده است '
        this.Label1.Text = "You clicked a button.";
    }
}
```

۲) **رویداد کلیک Click خود دکمه فرمان** : زمانی که می خواهیم تا یکسری کدها و دستورات ، پس از کلیک بر روی یک دکمه فرمان خاص اجرا شوند ، دستورات مورد نظر را در رویداد کلیک دکمه فرمان قرار می دهیم . با دابل کلیک بر روی دکمه فرمان در محیط Design صفحه ، می توانید به event handler ، رویداد کلیک آن دسترسی پیدا کنید .

مثال عملی ۲ : مثال عملی زیر ، نحوه تعریف کد مورد نظر در رویداد کلیک دکمه فرمان Button1 را نشان می دهد . پس از کلیک کاربر بر روی آن ، کنترل Label1 ، یک پیام هشدار را در صفحه نمایش می دهد :

```
public void Button1_Click (object sender, System.EventArgs e)
{
    Label1.Text="You clicked a button.";
}
```