

ASP.NET 2005 ページ間データ受け渡し

[ASP.NET 2005][ASP.NET][Visual Studio][C#][Java EE(SCWCD)]

- ・ <http://msdn.microsoft.com/ja-jp/netframework/ee692374.aspx>

Cookie

- ・ Web アプリケーションは履歴やユーザー設定などのユーザー固有の情報を格納できます
- ・ ブラウザは、Cookie をクライアント コンピュータで管理します
- ・ Cookies というプロパティを公開する `HttpResponse` オブジェクトを使用してクライアントに送信されます
- ・ Web アプリケーションからブラウザに送信するすべての Cookie は、このコレクションに追加する必要があります
- ・ Cookie は、ASP.NET ページをクライアントにレンダリングする前に作成する必要があります

Cookie を記述するには、2 つの方法

Cookie のプロパティを直接 Cookies コレクションに設定

```
Response.Cookies["UserSettings"]["Font"] = "Arial";
```

HttpCookie オブジェクトのインスタンスを作成してから Cookies コレクションに追加

```
HttpCookie myCookie = new HttpCookie("UserSettings");  
myCookie["Font"] = "Arial";  
Response.Cookies.Add(myCookie);
```

Session

- ・ 複数の要求に対して一意のブラウザ セッションに関連付けられた情報を格納するためのセッション状態管理機能を提供
- ・ `HttpContext` の `Session` プロパティまたは `Page` の `Session` プロパティを使用してアクセス可能な `HttpSessionState` クラスを使用してアクセス

Session データの関連付け

- ・ 一意の識別子を使用して特定のブラウザ セッションに関連付けられます
- ・ 既定では、この識別子はブラウザの無期限のセッション Cookie に格納
- ・ セッション識別子を URL に格納するようにアプリケーションを構成することもできる
 - ・ `cookieless` 属性の `UseDeviceProfile` の値を指定すると、ブラウザが Cookie をサポートしているかどうかを確認する。
 - ・ `cookieless` 属性の `AutoDetect` の値を指定すると、ブラウザで Cookie が有効になっているかどうかを確認できる。
 - ・ `UseDeviceProfile` を指定し Cookie がサポートされている、または `AutoDetect` を指定したときに Cookie が有効ならセッション識別子を Cookie に格納
 - ・ それ以外の場合はセッション識別子を URL に格納します。

値を読む

```
string firstName = (string)(Session["First"]);
```

値を保存

```
Session["FirstName"] = firstName;
```

QueryString

- ・ `HttpRequest.QueryString` プロパティに、クライアントから送信されたクエリ文字列変数のコレクションを格納

HttpRequest.Params

- ・ `QueryString`、`Form`、`ServerVariables`、`Cookies` の各項目が組み合わされたコレクションを取得

Session と Cookie と QueryString を利用する例

Default.aspx

画面



コントロール	プロパティ	値	内容
HyperLink1	<code>NavigateUrl</code>	<code>~/Page2.aspx?state=1</code>	QueryString の例
HyperLink2	<code>NavigateUrl</code>	<code>~/Page2.aspx?state=2</code>	QueryString の例

ソース

```
protected void Page_Load(object sender, EventArgs e)
{
    // Page(Web フォーム自身) の IsPostBack プロパティを調べて、
    // ポストバックによる再表示か、初回表示かを確認
    if (!IsPostBack)
    {
        // Cookie の設定
        Response.Cookies.Add(new HttpCookie("Data1", "Hello"));
    }
}
```

```

        // Session の設定
        Session["Data2"] = " こんにちは ";
    }
    Label1.Text = Response.Cookies["Data1"].Value;
    Label2.Text = (String)Session["Data2"];
}
protected void Button1_Click(object sender, EventArgs e)
{
    // Cookie に値を設定
    Response.Cookies["Data1"].Value = TextBox1.Text;
    // QueryString を付加してリダイレクト
    Response.Redirect(" /Page2.aspx?state=1");
}
protected void Button2_Click(object sender, EventArgs e)
{
    // Session に値を設定
    Session["Data2"] = TextBox1.Text;
    // QueryString を付加してリダイレクト
    Response.Redirect(" /Page2.aspx?state=2");
}
}

```

Page2.aspx

画面



ソース

```

protected void Page_Load(object sender, EventArgs e)
{
    // QueryString として 受け渡された値を表示
    Label1.Text = Request.QueryString["state"];

    // Cookie を利用して受け渡された値を表示
    Label2.Text = Request.Cookies["Data1"].Value;

    // Session を利用して受け渡された値を表示
    Label3.Text = (String)Session["Data2"];

    // すべてのパラメータを含む Request.Params の内容を確認
    StringBuilder sb = new StringBuilder();
    NameValueCollection p = Request.Params;
    for (int i=0; i < p.Count; i++) {
        sb.Append(String.Format("{0} = {1}<br/>", p.GetKey(i), p.Get(i)));
    }
    Literal1.Text = sb.ToString();
}

```

実行

Default.aspx



Page2.aspx

