

# ASP.NET Core 1.0

## 헬퍼 메서드

# 헬퍼 메서드

- 헬퍼 메서드를 호출하는 HTML 개체는 폼, 링크, 입력 요소 등을 생성
- 순수 HTML로 작성할 때보다 MVC의 라우팅 및 모델 클래스와 연동되는 부분에 대한 유용한 기능을 제공

## 헬퍼 메서드의 종류

- Form 확장: HTML 폼 생성
- Input 확장: HTML 폼 입력 요소 생성
- Label 확장: HTML 레이블 요소 생성
- Link 확장: HTML 하이퍼링크 생성
- Select 확장: 드롭다운 리스트/리스트 생성
- TextArea 확장: TextArea 요소 생성

# 헬퍼 메서드

## @Html.ActionLink()

- 링크를 하나 생성 해주는 헬퍼 메서드 (HTML의 <a>태그 생성)
- @Html.ActionLink("텍스트", "액션 메서드")
- @Html.ActionLink("텍스트", "액션 메서드", "컨트롤러", "파라미터")
- <a href="/컨트롤러/액션메서드/파라미터">텍스트</a>

```
@Html.ActionLink("ActionLink 테스트", "Index") <br />  
@Html.ActionLink("ActionLink 테스트", "Test")
```

```
<a href="/">ActionLink 테스트</a>  
<br>  
<a href="/Home/Test">ActionLink 테스트</a>
```

```
@{Html.BeginForm();}  
<h3>폼1의 내용 들어오는 곳</h3>  
@{Html.EndForm();}  
  
@using(Html.BeginForm()) {  
    <h3>폼2의 내용 들어오는 곳</h3>  
}
```

## @Html.BeginForm()

- Form 태그를 생성 해주는 헬퍼 메서드

CSRF(Cross Site Request Forgery)  
공격을 예방 하기 위한 토큰

- @using (Html.BeginForm()) { }

```
<form action="/Home/About" method="post">  
    <h3>폼1의 내용 들어오는 곳</h3>  
    <input name="__RequestVerificationToken" type="hidden" value="CfDJ8CDEBwxS969DtVn53LLvg7RG3mtQ_-HQ58h60ioCYs01VaHkLW0a-uS0" />  
</form>  
<form action="/Home/About" method="post">  
    <h3>폼2의 내용 들어오는 곳</h3>  
    <input name="__RequestVerificationToken" type="hidden" value="CfDJ8CDEBwxS969DtVn53LLvg7RG3mtQ_-HQ58h60ioCYs01VaHkLW0a-uS0" />  
</form>
```

# 링크 헬퍼 메서드

## @Url.Action()

- 특정 액션 메서드의 경로를 반환
- `Url.Action("액션 메서드", "컨트롤러");`

```
@Url.Action("Index") <br />  
@Url.Action("Index", "Home") <br />  
@Url.Action("About", "Home")
```

```
"/ "  
/ "  
<br>  
"/ "  
<br>  
"/Home/About  
"
```

# 입력 요소 헬퍼 메서드

## @Html.TextBox

- 싱글라인 텍스트 박스 생성

## @Html.Password

- 패스워드 텍스트박스 생성

## @Html.RadioButton

- 라디오 버튼 생성

## @Html.CheckBox

- 체크박스 생성

## @Html.Hidden

- 히든 필드 생성

## @Html.TextArea

- 멀티라인 텍스트 박스 생성

## @Html.DropDownList

- 드롭다운 리스트 생성

## @Html.ListBox

- 리스트 박스 생성

# 입력 요소 헬퍼 메서드

```
@Html.TextBox("InputTextBox", "TextBox Value") <br />  
@Html.Password("InputPassword", "Password Value") <br />  
@Html.RadioButton("InputRadio", "Radio Value") <br />  
@Html.CheckBox("InputCheckBox", "CheckBox Value") <br />  
@Html.Hidden("InputHidden", "Hidden Value") <br />  
@Html.TextArea("TextArea", "TextArea Value") <br />  
@Html.DropDownList("Select Box", new SelectList(new[] { "item1", "item2" })) <br />  
@Html.ListBox("List Box", new MultiSelectList(new[] { "item1", "item2" })) <br />
```

```
<input id="InputTextBox" name="InputTextBox" type="text" value="TextBox Value">  
<br>  
<input id="InputPassword" name="InputPassword" type="password" value="Password Value">  
<br>  
<input id="InputRadio" name="InputRadio" type="radio" value="Radio Value">  
<br>  
<input id="InputCheckBox" length="14" name="InputCheckBox" type="checkbox" value="true">  
<input name="InputCheckBox" type="hidden" value="false">  
<br>  
<input id="InputHidden" name="InputHidden" type="hidden" value="Hidden Value">  
<br>  
<textarea id="TextArea" name="TextArea">TextArea Value</textarea>  
<br>  
▼<select id="Select_Box" name="Select Box">  
  <option>item1</option>  
  <option>item2</option>  
</select>  
<br>  
▼<select id="List_Box" multiple="multiple" name="List Box">  
  <option>item1</option>  
  <option>item2</option>  
</select>
```

# 데이터 표시

```
참조 5개
public class Item
{
    참조 1개 | 0 예외
    public int ItemId { get; set; }

    [Required(ErrorMessage = "Category is required")]
    [Range(1, int.MaxValue, ErrorMessage = "Category is required")]
    참조 1개 | 0 예외
    public int CategoryId { get; set; }

    [Display(Name = "Item Name")]
    [Required(ErrorMessage = "Name is required")]
    [StringLength(160)]
    참조 3개 | 0 예외
    public string Name { get; set; }

    [Required(ErrorMessage = "Price is required")]
    [Range(0.01, 100.00,
        ErrorMessage = "Price must be between 0.01 and 100.00")]
    참조 1개 | 0 예외
    public decimal Price { get; set; }
}
```

```
@model Practice2.Models.Item
```

```
<h3>@Html.DisplayFor(m => m.Name)</h3>
```

```
<h3>@Html.LabelFor(m => m.Name)</h3>
```

# 유효성 검사(Validation)

## ValidationMessageFor()

- 특정 텍스트 박스 다음에 에러메시지를 출력

## ValidationSummary()

- 모든 에러 메시지를 한곳에 모아서 출력

UserID	<input type="text"/>
	UserID 값은 필수입니다.
Password	<input type="text"/>
	Password 값은 필수입니다.

```
[Required(ErrorMessage = "User ID 값은 필수입니다.")]
```

참조 4개 | 0 예외

```
public string userId { get; set; }
```

```
[Required(ErrorMessage = "Password 값은 필수입니다.")]
```

참조 4개 | 0 예외

```
public string pwd { get; set; }
```

- UserID 값은 필수입니다.
- Password 값은 필수입니다.

UserID	<input type="text"/>
Password	<input type="text"/>

# Html.Raw()

```
public IActionResult About() {  
    ViewBag.strHtml = "안녕하세요?<br /> 잘부탁드립니다."  
    return View();  
}
```

```
<h2>@ViewBag.strHtml</h2>  
<h2>@Html.Raw(ViewBag.strHtml)</h2>
```

안녕하세요?<br /> 잘부탁드립니다.

안녕하세요?  
잘부탁드립니다.

# 기타 헬퍼 메서드

```
public IActionResult About()
{
    ViewBag.tag = "<script>alert('.Net Core');</script>";
    return View();
}
```

```
<h3>@Html.Raw(ViewBag.tag)</h3><br />
<h3>@Html.Encode(ViewBag.tag)</h3><br />
<h3>@ViewBag.tag</h3>
```



@Html.Encode();

- 문자열을 인코딩하여 출력

localhost:56852 내용:  
.Net Core

확인

&lt;script&gt;alert(&#x27;.Net Core&#x27;);&lt;/script&gt;

<script>alert('.Net Core');</script>

© 2017 - Practice2

@Html.Partial();

- 특정 페이지를 현재 페이지에 포함 시킨다
- @Html.Partial("특정 페이지");

# CSS class와 같은 속성 부여

```
<h4>Html Input Text</h4>  
<input name="txtName" class="form-control" id="txtNmae" type="text" />  
  
<h4>헬퍼 메서드</h4>  
@Html.TextBox("txtName", null, new { @class = "form-control" })
```

Html Input Text

헬퍼 메서드

```
<h4>Html Input Text</h4>  
<input name="txtName" class="form-control" id="txtNmae" type="text">  
<h4>헬퍼 메서드</h4>  
<input class="form-control" id="txtName" name="txtName" type="text" value>
```