21 jQuery Ajax

모던 웹을 위한 Javascript jQuery 입문

한빛미디어 Hanbit Media, Inc.

21.1 기본적인 jQuery Ajax

❖ 기본적인 jQuery Ajax

- 예제 코드 21-1]
  - Index.cshtml 파일에 jQuery 추가
  - 버전 문제로 CDN 사용하는 편이 좋음

코드 21-1 Index.cshtml :

```
@{ Layout = null; }
<!DOCTYPE html>
<html>
<head>
    <title>Index</title>
    <script src="http://code.jquery.com/jquery-1.7.js"></script>
    <script>
        $(document).ready(function () {

        });
    </script>
</head>
<body>

</body>
</html>
```

21.1 기본적인 jQuery Ajax

❖ $.ajax() 메서드

- $.ajax(options);
- $.ajax(url, options);
- 예제 코드 21-2]
  - $.ajax() 메서드의 첫 번째 매개 변수에 문자열 /Home/MyFirstStringAction
  - 두 번째 매개 변수에는 옵션 객체

코드 21-2 Index.cshtml: $.ajax( ) 메서드

```
<script>
    $(document).ready(function () {
        $.ajax('/Home/MyFirstStringAction', {
            success: function () { }
        });
    });
</script>
```

21.1 기본적인 jQuery Ajax

❖ $.ajax() 메서드 (2)

- Ajax가 성공했을 때 자동으로 success 이벤트 실행
- success 이벤트 핸들러의 첫 번째 매개 변수는 Ajax가 성공했을 때 받은 데이터 – 문서 객체에 추가

코드 21-3 Index.cshtml: $.ajax() 메서드

```
<script>
    $(document).ready(function () {
        $.ajax('/Home/MyFirstStringAction', {
            success: function (data) {
                $('body').append(data);
            }
        });
    });
</script>
```

그림 21-1 실행 결과

Hello ASP.NET MVC

❖ $.ajax() 메서드 (3)

- $.ajax() 메서드의 매개 변수 url
  - 예제 코드 21-4처럼 옵션 속성으로 지정 가능
  - 보통은 자동 지정

코드 21-4 Index.cshtml: url 옵션 속성을 사용한 $.ajax( ) 메서드

```
<script>
    $(document).ready(function () {
        $.ajax({
            url: '/Home/MyFirstStringAction',
            success: function (data) {
                $('body').append(data);
            }
        });
    });
</script>
```

21.1 기본적인 jQuery Ajax

❖ $.ajax() 메서드의 옵션

표 21-1 jQuery.Ajax( ) 메서드의 옵션

옵션 속성 이름	설명	자료형
async	동기, 비동기를 지정합니다.	Boolean
complete(jqXHR, textStatus)	Ajax 완료 이벤트 핸들러를 지정합니다.	Function
data	요청 매개 변수를 지정합니다.	Object, String
error(jqXHR, textStatus, errorThrown)	Ajax 실패 이벤트 핸들러를 지정합니다.	Function
jsonp	JSONP 매개 변수 이름을 지정합니다.	String
jsonpCallback	JSONP 콜백 함수 이름을 지정합니다.	String, Function
success(data, textStatus, jqXHR)	Ajax 성공 이벤트 핸들러를 지정합니다.	Function, Array
timeout	만료 시간을 지정합니다.	Number
type	'GET' 또는 'POST'를 지정합니다.	String
url	대상 URL을 지정합니다.	String

21.1 기본적인 jQuery Ajax

❖ Ajax 요청 수행

- 표 21-1의 옵션 속성 중 data 속성 사용
  - 요청 매개 변수를 쉽게 지정 가능
- 예제 코드 21-5]
  - $.ajax() 메서드의 옵션객체에 url 속성 입력
  - data 속성에 객체 입력
  - http://localhost:Port/Home/ActionWithData?name=rintiantta&age=21 로 Ajax 요청 수행

코드 21-5 Index.cshtml: data 옵션 속성을 사용한 $.ajax() 메서드

```
<script>
    $(document).ready(function () {
        $.ajax({
            url: '/Home/ActionWithData',
            data: { name: 'rintiantta', age: 21 },
            success: function (data) {
                $('body').append(data);
            }
        });
    });
</script>
```

그림 21-2 실행 결과

21.2 추가적인 jQuery Ajax 메서드

❖ jQuery가 지원하는 추가적 Ajax 메서드

표 21-2 jQuery의 Ajax 관련 메서드

메서드 이름	설명
$.get( )	get 방식으로 Ajax를 수행합니다.
$.post( )	post 방식으로 Ajax를 수행합니다.
$.getJSON( )	get 방식으로 Ajax를 수행해 JSON 데이터를 가져옵니다.
$.getScript( )	get 방식으로 Ajax를 수행해 Script 데이터를 가져옵니다.
$(selector).load( )	Ajax를 수행한 후에 선택자로 선택한 문서 객체 안에 응답받을 문자열을 넣습니다.

```
$.get(url, function (data, textStatus, jqXHR) { });
$.post(url, function (data, textStatus, jqXHR) { });

$.get(url, data, function (data, textStatus, jqXHR) { });
$.post(url, data, function (data, textStatus, jqXHR) { });
```

21.2 추가적인 jQuery Ajax 메서드

❖ $.get() 메서드 사용

코드 21-6 Index.cshtml: $.get() 메서드

```
<script>
    $(document).ready(function () {
        $.get('/Home/MyFirstStringAction', function (data) {
            $('body').html(data);
        });
    });
</script>
```

그림 21-3 실행 결과

Hello ASP.NET MVC

21.2 추가적인 jQuery Ajax 메서드

❖ $.post() 메서드 사용

- HTML 태그 가져와 문서 객체에 출력할 때
  - $(selector).load() 메서드 사용하면 훨씬 간단

코드 21-7 Index.cshtml : $.post() 메서드

```
<script>
    $(document).ready(function () {
        $.post('/Home/MyFirstStringAction', function (data) {
            $('body').html(data);
        });
    });
</script>
```

코드 21-8 Index.cshtml : $(selector).load() 메서드

```
<script>
    $(document).ready(function () {
        $('body').load('/Home/MyFirstStringAction');
    });
</script>
```

21.2 추가적인 jQuery Ajax 메서드

❖ $.getJSON() 메서드 사용

코드 21-9 Index.cshtml: $.getJSON() 메서드

```
<script>
    $(document).ready(function () {
        $.getJSON('/Home/MyFirstJsonAction', function (data) {
            $.each(data, function (key, value) {
                $('body').append('<h1>' + key + ' : ' + value + '</h1>');
            });
        });
    });
</script>
```

그림 21-4 getJSON() 메서드를 사용한 실행 결과

name : 윤인성

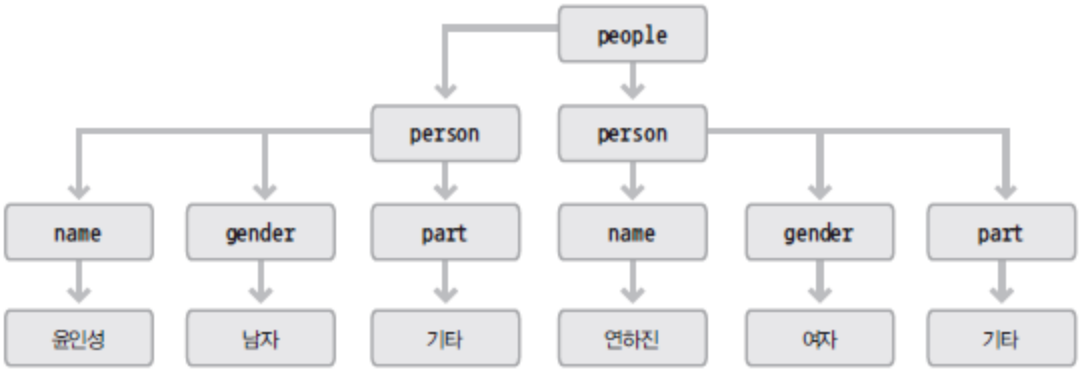
gender : 남자

part : 세컨드기타

21.3 XML 다루기

❖ 액션 MyFirstXMLAction0이 제공하는 XML 문서

그림 21-5 XML 문서의 DOM



21.3 XML 다루기

❖ jQuery를 사용해 XML 문서 객체 다루기

- find() 메서드 사용
  - find() 메서드의 매개 변수에 찾고자 하는 태그의 이름 입력

코드 21-10 Index.cshtml : find( ) 메서드

```
<script>
    $(document).ready(function () {
        $.ajax('/Home/MyFirstXMLAction', {
            success: function (data) {
                $(data).find('person').each(function () {

                });
            }
        });
    });
</script>
```

21.3 XML 다루기

❖ jQuery를 사용해 XML 문서 객체 다루기 (2)

- find() 메서드 사용
  - person 태그가 두 개이므로 each() 메서드 사용해 내부 탐색
  - find() 메서드를 다시 한번 사용해 name, gender, part 태그를 가져옴
  - text() 메서드로 각 내용을 끄집어 냄

코드 21-11 Index.cshtml : XML 문서의 요소 추출

```
$.ajax('/Home/MyFirstXMLAction', {
    success: function (data) {
        $(data).find('person').each(function () {
            $(this).find('name').text()
            $(this).find('gender').text()
            $(this).find('part').text()
        });
    }
});
```

21.3 XML 다루기

❖ 예제 코드 21-12]

- 가져온 XML 문서를 간단하게 HTML 태그로 구성해 출력

코드 21-12 Index.cshtml : XML 문서를 HTML 태그로 구성

```
$.ajax('/Home/MyFirstXMLAction', {
    success: function (data) {
        $(data).find('person').each(function () {
            var name = '<li>' + $(this).find('name').text() + '</li>';
            var gender = '<li>' + $(this).find('gender').text() + '</li>';
            var part = '<li>' + $(this).find('part').text() + '</li>';

            $('#wrap').append('<ul>' + name + gender + part + '</ul>');
        });
    }
});
```

그림 21-6 실행 결과

- 윤인성
- 남자
- 기타

- 연하진
- 여자
- 기타

21.4 jQuery Ajax 보조 메서드

❖ jQuery Ajax 보조 메서드

표 21-3 jQuery Ajax 보조 메서드

메서드 이름	설명
serialize( )	입력 양식의 내용을 요청 매개 변수 문자열로 만듭니다.
serializeArray( )	입력 양식의 내용을 객체로 만듭니다.
$.param( )	객체의 내용을 요청 매개 변수 문자열로 만듭니다.

❖ $.param() 메서드

- 객체를 쿼리 문자열로 바꿈

코드 21-13 Index.cshtml : $.param() 메서드

```
<script>
    $(document).ready(function () {
        // 변수를 선언합니다.
        var data = {
            name: 'RintIanTta',
            gender: 'Man',
            part: 'Guitar'
        };

        // 출력합니다.
        $('body').html('<h1>' + $.param(data) + '</h1>');
    });
</script>
```

그림 21-7 param() 메서드 사용 결과

name=RintIanTta&gender=Man&age=21

21.4 jQuery Ajax 보조 메서드

❖ body 태그 구성

- serialize() 메서드와 serializeArray() 메서드 테스트
  - Name과 Age 입력
  - get Ajax String 버튼을 누르면 '/Home/ActionWithData'로 Ajax 요청 수행

코드 21-14 Index.cshtml: body 태그 구성

```
<body>
    <form id="my_form">
        <table>
            <tr>
                <td><label for="name">Name</label></td>
                <td><input id="name" name="name" type="text"/></td>
            </tr>
            <tr>
                <td><label for="age">Age</label></td>
                <td><input id="age" name="age" type="text"/></td>
            </tr>
        </table>
        <input type="submit" value="Get Ajax String"/>
    </form>
    <div id="wrap">

    </div>
</body>
```

그림 21-8 생성된 입력 양식

21.4 jQuery Ajax 보조 메서드

❖ script 태그

- 입력 양식의 submit 이벤트 연결
- 쿼리 전송 후 페이지 전환을 막기 위해 preventDefault() 메서드 사용

코드 21-15 Index.cshtml : submit 이벤트 연결

```
<script>
    $(document).ready(function () {
        $('#my_form').submit(function (event) {
            event.preventDefault();
        });
    });
</script>
```

21.4 jQuery Ajax 보조 메서드

❖ jQuery Ajax를 사용해 입력 양식을 전송하는 방법 (1)

- 각각의 입력 양식에서 value 속성 직접 가져온 뒤 URL 생성
- 버튼의 click 이벤트 실행될 때 사용

코드 21-16 Index.cshtml : 형태1

```
$('#my_form').submit(function (event) {
    // 입력 양식의 value 속성을 가져옵니다.
    var name = $('#name').val();
    var age = $('#age').val();

    // ActionWithData에서 자료를 가져옵니다.
    var url = '/Home/ActionWithData?name=' + name + '&age=' + age;
    $('#wrap').load(url);

    // 기본 이벤트를 제거합니다.
    event.preventDefault();
});
```

21.4 jQuery Ajax 보조 메서드

❖ jQuery Ajax를 사용해 입력 양식을 전송하는 방법 (2)

- 각각의 입력 양식에서 value 속성 가져옴
- Value 속성을 이용해 객체 만들고 param() 메서드 사용해 쿼리로
- Ajax와 관련된 메서드의 data 속성에 넣음
  - param() 메서드 사용하지 않고 문자열 직접 생성해 입력도 가능
  - 추천하지 않는 방법

코드 21-17 Index.cshtml: 형태 2

```
$('#my_form').submit(function (event) {
    // 입력 양식의 value 속성을 가져옵니다.
    var name = $('#name').val();
    var age = $('#age').val();

    // ActionWithData에서 자료를 가져옵니다.
    var url = '/Home/ActionWithData';
    var data = { name: name, age: age };
    var params = $.param(data);
    $('#wrap').load(url, params);

    // 기본 이벤트를 제거합니다.
    event.preventDefault();
});
```

21.4 jQuery Ajax 보조 메서드

❖ jQuery Ajax를 사용해 입력 양식을 전송하는 방법 (3)

- 각 입력 양식에서 value 속성 가져온 후 객체 생성
- Ajax와 관련된 메서드의 data 속성에 집어넣는 방법
- 많이 사용되지만 submit 이벤트와 연결하는 경우는 많지 않음

코드 21-18 Index.cshtml: 형태 3

```
$('#my_form').submit(function (event) {
    // 입력 양식의 value 속성을 가져옵니다.
    var name = $('#name').val();
    var age = $('#age').val();

    // ActionWithData에서 자료를 가져옵니다.
    var url = '/Home/ActionWithData';
    var data = { name: name, age: age };
    $('#wrap').load(url, data);

    // 기본 이벤트를 제거합니다.
    event.preventDefault();
});
```

21.4 jQuery Ajax 보조 메서드

❖ jQuery Ajax를 사용해 입력 양식을 전송하는 방법 (4)

- serialize() 메서드 사용
  - 입력 양식에 적힌 값을 쿼리 문자열로 바꿈

코드 21-19 Index.cshtml: 형태 4

```
$('#my_form').submit(function (event) {
    // ActionWithData에서 자료를 가져옵니다.
    var url = '/Home/ActionWithData';
    $('#wrap').load(url, $(this).serialize());

    // 기본 이벤트를 제거합니다.
    event.preventDefault();
});
```

21.4 jQuery Ajax 보조 메서드

❖ jQuery Ajax를 사용해 입력 양식을 전송하는 방법 (5)

- serializeArray() 메서드 사용
  - 입력 양식에 적힌 값 객체 생성
  - Ajax 관련 메서드의 data 속성에 넣음

코드 21-20 Index.cshtml: 형태 5

```
$('#my_form').submit(function (event) {
    // ActionWithData에서 자료를 가져옵니다.
    var url = '/Home/ActionWithData';
    $('#wrap').load(url, $(this).serializeArray());

    // 기본 이벤트를 제거합니다.
    event.preventDefault();
});
```

그림 21-9 실행 결과

21.4 jQuery Ajax 보조 메서드

❖ serialize() 메서드와 serializeArray() 메서드의 차이

코드 21-21 Index.cshtml: serialize() 메서드와 serializeArray() 메서드

```
$('#my_form').submit(function (event) {
    // ActionWithData에서 자료를 가져옵니다.
    var url = '/Home/ActionWithData';
    var serialize = $(this).serialize();
    var serializeArray = $(this).serializeArray();
    // 기본 이벤트를 제거합니다.
    event.preventDefault();
});
```

그림 21-10 실행 결과

name=RintIanTta&age=21

[object Object],[object Object]

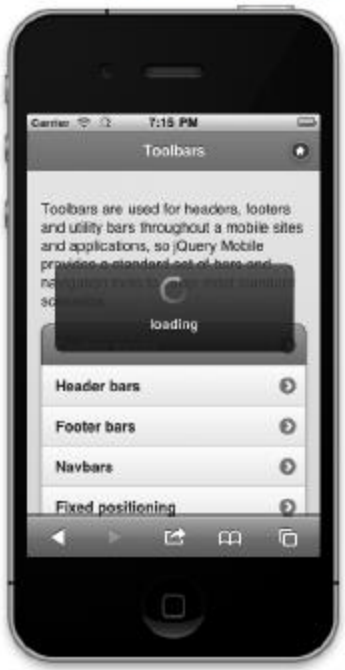
serializeArray()가 리턴한 객체를 확인해보면 다음과 같은 형태입니다.

```
[
    { name: 'name', value: 'RintIanTta' },
    { name: 'age', value: '21' }
]
```

21.5 jQuery Ajax 이벤트 핸들링

❖ 스피너 (Spinner)

그림 21-11 jQuery Mobile의 스피너



21.5 jQuery Ajax 이벤트 핸들링

❖ 스피너를 구현하는 방법

- 표 21-4의 이벤트 연결 메서드 사용

표 21-4 jQuery Ajax 이벤트 연결 메서드

메서드 이름	설명
ajaxComplete( )	Ajax 요청이 완료될 때 이벤트 핸들러를 실행합니다.
ajaxError( )	Ajax 요청이 실패할 때 이벤트 핸들러를 실행합니다.
ajaxSend( )	Ajax 요청을 보낼 때 이벤트 핸들러를 실행합니다.
ajaxStart( )	Ajax 요청을 시작할 때 이벤트 핸들러를 실행합니다.
ajaxStop( )	Ajax 요청을 중지할 때 이벤트 핸들러를 실행합니다.
ajaxSuccess( )	Ajax 요청이 성공할 때 이벤트 핸들러를 실행합니다.

코드 21-22 Index.cshtml : body 태그 구성

```
<body>
    <div id="ajax_event">

    </div>
    <div id="wrap">

    </div>
</body>
```

21.5 jQuery Ajax 이벤트 핸들링

❖ Ajax 이벤트 연결 메서드

- Ajax 수행하면 자동으로 해당 이벤트 핸들러 호출
- 테스트를 위해 마지막 줄에 간단한 Ajax 요청 수행

코드 21-23 Index.cshtml: jQuery의 Ajax 이벤트 메서드

```
<script>
    $(document).ready(function () {
    // Ajax 이벤트 연결
        $('#ajax_event').ajaxComplete(function () {
            $(this).append('<h1>AjaxComplete<h1>')
        }).ajaxError(function () {
            $(this).append('<h1>AjaxError<h1>')
        }).ajaxSend(function () {
            $(this).append('<h1>AjaxSend<h1>')
        }).ajaxSuccess(function () {
            $(this).append('<h1>AjaxSuccess<h1>')
        }).ajaxStart(function () {
            $(this).append('<h1>AjaxStart<h1>')
        });

        // Ajax 수행
        $('#wrap').load('/Home/MyFirstStringAction');
    });
</script>
```

그림 21-12 Ajax 이벤트

Thank You !

모던 웹을 위한 Javascript jQuery 입문

한빛미디어 Hanbit Media, Inc.