



# 21 jQuery Ajax

모던 웹을 위한 Javascript & jQuery 입문(개정판)

## ❖ 기본

- \$.ajax ( ) 메서드는 다음 두 가지 형태

```
$.ajax(options);  
$.ajax(url, options);
```

### 코드 21-2 \$.ajax ( ) 메서드(1)

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



## ❖ 기본

- jQuery를 활용한 Ajax

코드 21-3 jQuery를 활용한 Ajax

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



## ❖ 기본

### ■ \$.ajax( ) 메서드(2)

코드 21-4 \$.ajax( ) 메서드(2)

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



## ❖ 기본

### ■ \$.ajax( ) 메서드의 옵션

표 21-1 \$.ajax( ) 메서드의 옵션

옵션 속성 이름	설명	자료형
async	동기, 비동기를 지정합니다.	Boolean
complete(xhr, status)	Ajax 완료 이벤트 리스너를 지정합니다.	Function
data	요청 매개변수를 지정합니다.	Object, String
error(xhr, status, error)	Ajax 실패 이벤트 리스너를 지정합니다.	Function
jsonp	JSONP 매개변수 이름을 지정합니다.	String
jsonpCallback	JSONP 콜백 함수 이름을 지정합니다.	String, Function
success(data, status, xhr)	Ajax 성공 이벤트 리스너를 지정합니다.	Function, Array
timeout	만료 시간을 지정합니다.	Number
type	'GET' 또는 'POST' 등을 지정합니다.	String
url	대상 URL을 지정합니다.	String



## ❖ 추가적인 jQuery Ajax 메서드

### ■ 추가적인 jQuery Ajax 메서드

표 21-2 추가적인 jQuery Ajax 메서드

메서드 이름	설명
<code>\$.get( )</code>	get 방식으로 Ajax를 수행합니다.
<code>\$.post( )</code>	post 방식으로 Ajax를 수행합니다.
<code>\$.getJSON( )</code>	get 방식으로 Ajax를 수행해 JSON 데이터를 가져옵니다.
<code>\$.getScript( )</code>	get 방식으로 Ajax를 수행해 Script 데이터를 가져옵니다.
<code>\$(selector).load( )</code>	Ajax를 수행하고 선택자로 선택한 문서 객체 안에 집어넣습니다.

### ■ \$.get ( ) 메서드와 \$.post ( ) 메서드 사용 형태

```
$.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

## ❖ 추가적인 jQuery Ajax 메서드

### ■ get( ) 메서드

코드 21-6 \$.get( ) 메서드

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



# 21.2 추가적인 jQuery Ajax 메서드

## ❖ 추가적인 jQuery Ajax 메서드

### ■ load( ) 메서드

코드 21-7 load( ) 메서드

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





# 21.2 추가적인 jQuery Ajax 메서드

## ❖ 추가적인 jQuery Ajax 메서드

### ■ \$.getJSON( ) 메서드

코드 21-8 \$.getJSON( ) 메서드

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



### ❖ XML 조작

- jQuery의 find ( ) 메서드로 XML 문서를 다루는 방법

코드 21-9 find( ) 메서드

```
<script>
$(document).ready(function () {
    $.ajax({
        url: 'data.xml',
        success: function (data) {
            $(data).find('product').each(function () {

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



## ❖ XML 조작

### ■ XML 문서의 요소 추출

코드 21-10 XML 문서의 요소 추출

```
$.ajax({  
  url: 'data.xml',  
  success: function (data) {  
    $(data).find('product').each(function () {  
      $(this).find('name').text()  
      $(this).find('price').text()  
    });  
  }  
});
```



## ❖ XML 조작

### ■ XML 조작

코드 21-11 XML 조작

```
$.ajax({  
  url: 'data.xml',  
  success: function (data) {  
    $(data).find('product').each(function () {  
      // 변수를 선언합니다.  
      var name = $(this).find('name').text();  
      var price = $(this).find('price').text();  
  
      // 출력합니다.  
      $('<h1></h1>').text(name + ':' + price).appendTo('body')  
    });  
  }  
});
```



## ❖ GET 요청

### ■ GET 요청

코드 21-13 GET 요청

```
$('#get').click(function () {  
    $.ajax({  
        url: '/products',  
        type: 'get',  
        dataType: 'text',  
        success: function (data) {  
            $('#output').val(data);  
        }  
    });  
});
```



## ❖ POST 요청

- POST 요청
- data 속성을 넣어 데이터를 전달한다는 것만 빼면 GET 요청과 다른 점이 없음

코드 21-14 POST 요청

```
$('#post').click(function () {  
    $.ajax({  
        url: '/products',  
        type: 'post',  
        dataType: 'text',  
        data: {  
            name: $('#name').val(),  
            price: $('#price').val()  
        },  
        success: function (data) {  
            $('#output').val(data);  
        }  
    });  
});
```



## ❖ PUT 요청

### ■ PUT 요청

코드 21-15 PUT 요청

```
$('#put').click(function () {  
    $.ajax({  
        url: '/products/0',  
        type: 'put',  
        dataType: 'text',  
        data: {  
            name: $('#name').val(),  
            price: $('#price').val()  
        },  
        success: function (data) {  
            $('#output').val(data);  
        }  
    });  
});
```



## ❖ DELETE 요청

### ■ DELETE 요청

코드 21-16 DELETE 요청

```
$('#delete').click(function () {  
    $.ajax({  
        url: '/products/0',  
        type: 'delete',  
        dataType: 'text',  
        success: function (data) {  
            $('#output').val(data);  
        }  
    });  
});
```





## ❖ 보조 메서드

### ■ 보조 메서드

표 21-3 보조 메서드

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



## ❖ 보조 메서드

- \$.param ( ) 메서드는 객체를 쿼리 문자열로 바꿈
- 다음 코드는 data 객체를 \$.param ( ) 메서드를 사용해 쿼리 문자열로 바꾼 뒤 출력

코드 21-17 \$.param() 메서드

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

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



## ❖ 보조 메서드

- serialize ( ) 메서드와 serializeArray ( ) 메서드는 모두 입력 양식과 함께 사용하는 메서드
- body 태그 구성

코드 21-18 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="region">Region</label></td>
        <td><input id="region" name="region" type="text" /></td>
      </tr>
    </table>
    <input type="submit" value="Get Ajax String" />
  </form>
  <div id="wrap">

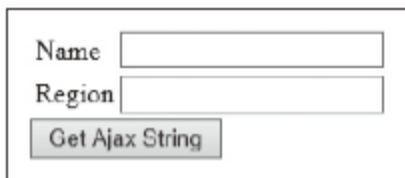
  <div>
</body>
```



## ❖ 보조 메서드

### ■ 입력 양식

그림 21-12 입력 양식



Name

Region

### ■ submit 이벤트 연결

코드 21-19 submit 이벤트 연결

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



## ❖ 보조 메서드

### ■ 요청 매개변수 전달 형태(1)

코드 21-20 요청 매개변수 전달 형태(1)

```
<script>
    $(document).ready(function () {
        $('#my-form').submit(function (event) {
            // 입력 양식의 value 속성을 가져옵니다.
            var name = $('#name').val();
            var region = $('#region').val();

            // Ajax 요청을 수행합니다.
            var url = '/parameter?name=' + name + '&region=' + region;
            $('#wrap').load(url);

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



## ❖ 보조 메서드

### ■ 요청 매개변수 전달 형태(2)

코드 21-21 요청 매개변수 전달 형태(2)

```
<script>
$(document).ready(function () {
    $('#my-form').submit(function (event) {
        // 입력 양식의 value 속성을 가져옵니다.
        var name = $('#name').val();
        var region = $('#region').val();

        // Ajax 요청을 수행합니다.
        var url = '/parameter';
        var data = { name: name, region: region };
        var params = $.param(data);
        $('#wrap').load(url, params);

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



## ❖ 보조 메서드

### ■ 요청 매개변수 전달 형태(3)

코드 21-22 요청 매개변수 전달 형태(3)

```
<script>
$(document).ready(function () {
    $('#my-form').submit(function (event) {
        // 입력 양식의 value 속성을 가져옵니다.
        var name = $('#name').val();
        var region = $('#region').val();

        // Ajax 요청을 수행합니다.
        var url = '/parameter';
        var data = { name: name, region: region };
        $('#wrap').load(url, data);

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



## ❖ 보조 메서드

### ■ 요청 매개변수 전달 형태(4)

코드 21-23 요청 매개변수 전달 형태(4)

```
<script>
$(document).ready(function () {
    $('#my-form').submit(function (event) {
        // Ajax 요청을 수행합니다.
        $('#wrap').load('/parameter', $(this).serialize());

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





## ❖ 보조 메서드

### ■ 요청 매개변수 전달 형태(5)

코드 21-24 요청 매개변수 전달 형태( 5)

```
<script>
$(document).ready(function () {
    $('#my-form').submit(function (event) {
        // Ajax 요청을 수행합니다.
        $('#wrap').load('/parameter', $(this).serializeArray());

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



## ❖ 보조 메서드

- serialize( ) 메서드와 serializeArray( ) 메서드
- serialize ( ) 메서드는 쿼리 문자열을 리턴
- serializeArray ( ) 메서드는 배열로 리턴

코드 21-25 serialize( ) 메서드와 serializeArray( ) 메서드

```
<script>
    $(document).ready(function () {
        $('#my-form').submit(function (event) {
            // 변수를 선언합니다.
            var serialize = $(this).serialize();
            var serializeArray = $(this).serializeArray();

            // 출력합니다.
            $('<h1></h1>').text(serialize).appendTo('#wrap');
            $('<h1></h1>').text(serializeArray).appendTo('#wrap');

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



## ❖ 이벤트 관리

### ■ jQuery Ajax 이벤트 연결 메서드

표 21-4 jQuery Ajax 이벤트 연결 메서드

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





# Thank You !

모던 웹을 위한 Javascript jQuery 입문(개정판)