



## Chapter 06-1

# jQuery 기초



# jQuery의 기초

# jQuery의 기초

## ■ jQuery

- 복잡한 Javascript를 쉽게 사용하기 위한 목적으로 생성된 라이브러리 중 하나
  - 2006년 자바스크립트 개발자인 존 레식(John Resig)에 의해 개발
- 자바스크립트 프로그래밍을 쉽게 하기 위한 다양한 함수를 제공
  - 지속적으로 업데이트되고 있음
- 브라우저 간의 호환성 문제 해결
  - 자바스크립트의 DOM 처리에서 발생하는 크로스 브라우징 문제를 해결
- jQuery는 자바스크립트보다는 쉬움
  - Javascript 기초를 이해하고 있으면 쉽게 배울 수 있음

# jQuery의 기초

## ■ jQuery의 특징

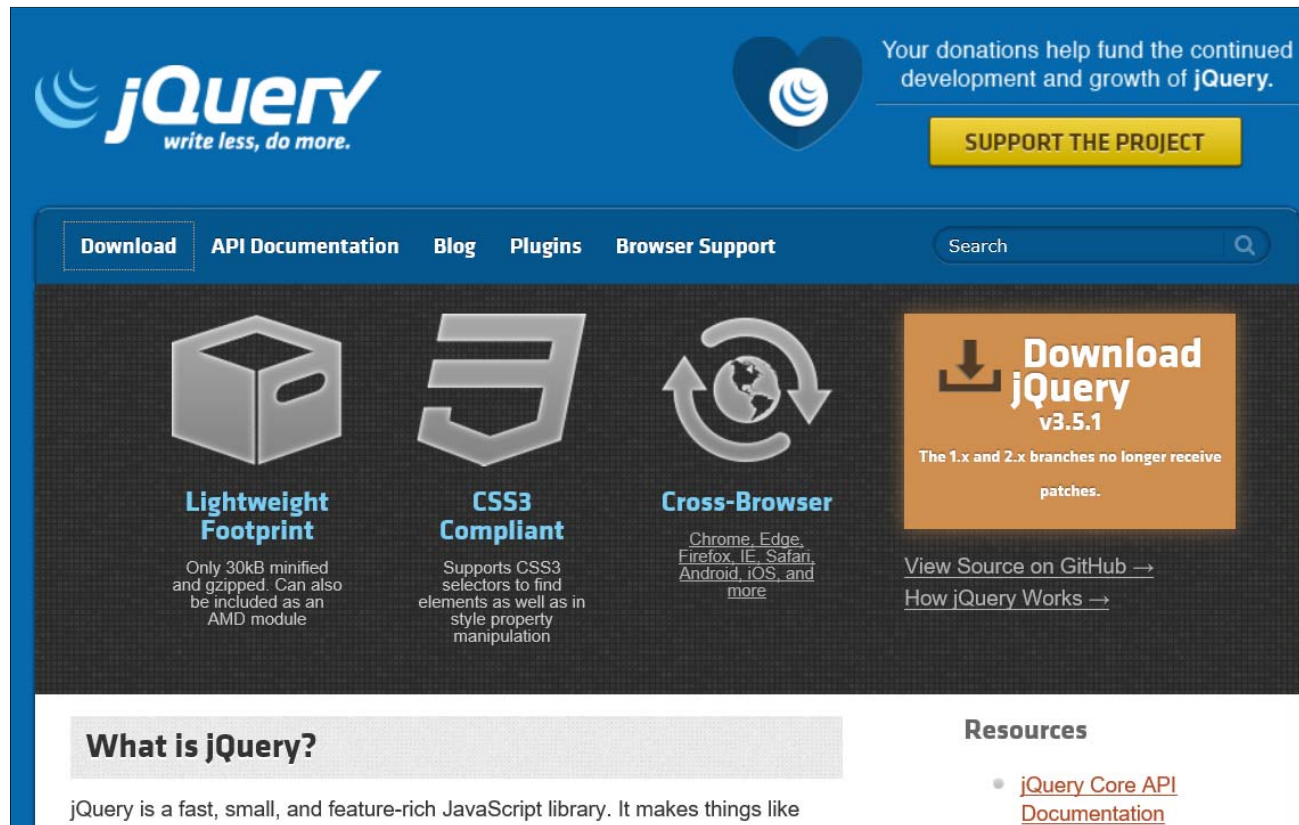
- Write Less, Do More
  - 적은 코드로 많은 일을 할 수 있다는 의미
- 오픈 소스( Open Source )
  - 오픈 소스 커뮤니티의 지원을 받으며, 충분한 테스트를 통해 지속적으로 업데이트되고 있음
  - jQuery는 무료이며 jQuery 플러그인 제작자의 이름만 명시하면 해당 코드를 사용할 수 있음
- 문서화( Documentation )
  - jQuery 함수의 사용 방법에 대한 문서화가 잘 되어있으며 많은 예제도 지원하고 있음
- 크로스 브라우징( Cross Browsing )
  - 자바스크립트는 브라우저마다 모두 동일한 형태로 나타나지 않음
  - jQuery는 모든 브라우저를 지원함
    - 버전에 따라 일부 기능의 수행에 차이는 있음 ( 점진적으로 지원 영역이 넓어지고 있음 )

# jQuery 라이브러리 연동

# jQuery 라이브러리 다운로드

## ■ jQuery의 다운로드

- jQuery 공식 사이트(<http://jquery.com>)에서 다운로드 가능



The screenshot shows the jQuery website homepage. At the top, the jQuery logo is on the left, and a heart icon with the text "Your donations help fund the continued development and growth of jQuery." is on the right, with a yellow "SUPPORT THE PROJECT" button below it. A navigation bar contains links for "Download", "API Documentation", "Blog", "Plugins", and "Browser Support", along with a search bar. The main content area features three icons: a box for "Lightweight Footprint", a stylized 'E' for "CSS3 Compliant", and a circular arrow for "Cross-Browser". To the right is a large orange box for "Download jQuery v3.5.1" with a download arrow icon. Below this box are links for "View Source on GitHub" and "How jQuery Works". At the bottom, there is a "What is jQuery?" section and a "Resources" section with a link to "jQuery Core API Documentation".

**jQuery**  
write less, do more.

Your donations help fund the continued development and growth of jQuery.

SUPPORT THE PROJECT

Download API Documentation Blog Plugins Browser Support Search

**Lightweight Footprint**  
Only 30kB minified and gzipped. Can also be included as an AMD module

**CSS3 Compliant**  
Supports CSS3 selectors to find elements as well as in style property manipulation

**Cross-Browser**  
Chrome, Edge, Firefox, IE, Safari, Android, iOS, and more

**Download jQuery v3.5.1**  
The 1.x and 2.x branches no longer receive patches.

View Source on GitHub →  
How jQuery Works →

**What is jQuery?**  
jQuery is a fast, small, and feature-rich JavaScript library. It makes things like

**Resources**  
• [jQuery Core API Documentation](#)

# jQuery 라이브러리 연동 방식

## ■ 다운로드 방식

- jQuery 라이브러리 파일을 다운받아 웹 문서에 연결하는 방법
  - `<script>` 태그를 사용해 다운로드한 라이브러리를 직접 지정

```
<head>  
  <script src="script/jquery-3.5.1.js"> </script>  
</head>
```

- 라이브러리 파일을 다운받아 `script` 폴더에 저장한 경우의 예
- 직접 지정 방식의 특징
  - 네트워크가 없는 상태에서도 사용 가능
  - 최신의 버전 유지에 어려움

# jQuery 라이브러리 연동 방식

## ■ 네트워크 전송 방식

- CDN(Content Delivery Network)을 통해 원격 서버에 있는 라이브러리를 참조하는 방법
  - <script> 태그를 사용해 라이브러리가 있는 서버의 URL을 지정

```
<head>  
  <script src="https://code.jquery.com/jquery-3.5.1.js"> </script>  
</head>
```

- Google과 Microsoft와 같이 외부에서도 CDN을 제공하고 있음
  - Google : <https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.js>
  - Microsoft : <https://ajax.aspnetcdn.com/ajax/jQuery/jquery-3.5.1.js>
- CDN 방식의 특징
  - 네트워크가 반드시 연결된 상태에서 사용 가능
  - 최신 버전 유지에 편의성이 있음



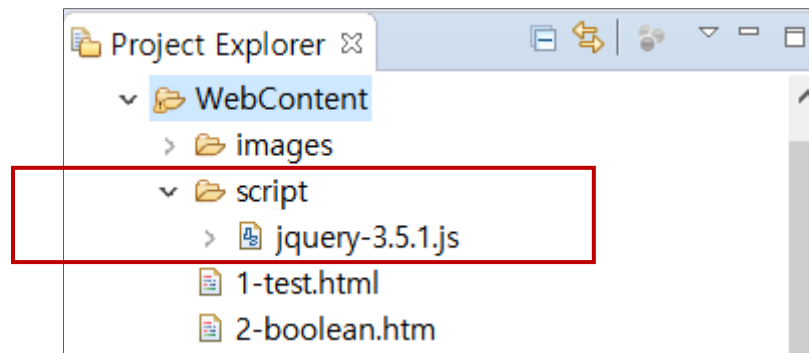
# jQuery 라이브러리 연동 방식

## ■ jQuery 라이브러리 파일의 종류

- Minified version
  - ○○.min.js 파일 형식
  - 용량을 최소화하기 위해 지핑(zipping)한 버전
  - 파일의 크기는 작지만 라이브러리 내용 참조에 어려움
- Uncompressed version
  - ○○.js 파일 형식
  - 파일의 크기는 크지만 라이브러리 내용 참조가 쉬움
- Minified version과 Uncompressed version은 기능면에서 동일함

## ■ 연동 방식

- 다운로드 방식을 사용
  - 사용할 버전은 현재(2020년 7월)의 최신 버전인 3.5.1을 사용
- 라이브러리 파일
  - Uncompressed versio을 다운받아 사용
  - WebContent에 script 폴더를 생성하고 라이브러리 파일을 다운로드 함



# 문서 준비 이벤트

## ■ 문서 준비 이벤트

- 브라우저가 문서 객체를 모두 로드했을 때 발생하는 이벤트

- ready() 메서드를 사용해 표현

```
<script>

    $(document).ready( function() {
        jQuery 코드
    });

</script>
```

- 브라우저가 현재 문서에 존재하는 모든 요소를 읽었다면 매개변수로 지정된 콜백 함수(callback function)을 실행하라는 의미

- 자바스크립트의 다음 구문과 유사

```
<script>
    window.onload = function() {

    };
</script>
```

# 문서 준비 이벤트

## ■ 문서 준비 이벤트 사용의 형식

```
jQuery(document).ready( function() {  
    jQuery 구문  
} );
```

- jQuery(document)
  - 현재의 웹 페이지를 의미
- ready()
  - 준비가 되었는지(문서가 모두 로드 되었는지)를 나타내는 메서드
- function() { 구문 }
  - 준비가 되면 실행되는 부분으로 콜백 함수(callback function)로 표현함

# 문서 준비 이벤트

## ■ 문서 준비 이벤트를 사용하는 이유

- 브라우저는 <HEAD> 부분을 먼저 읽어 들인 다음 <BODY> 부분을 읽어 들어 요소를 로드 함
  - jQuery 구문이 <HEAD>에 존재하는 존재할 경우
  - <BODY>를 읽지 않아 요소를 로드하지 않은 상태이므로 jQuery 구문을 수행할 수 없음
- 정상적인 jQuery 동작을 위해서는 요소를 모두 로드했다는 것을 알려주는 이벤트가 필요
  - 이를 문서 준비 이벤트라고 함
  - 모든 요소가 읽혀져 생성된 후 jQuery 구문이 수행되도록 지정함
  - jQuery 구문을 <HEAD> 내에 유지할 경우 ready() 메서드 내에 표현해야 함
- jQuery 코드를 <body> 태그 내에서 가장 마지막에 위치시킬 경우
  - 문서 준비 이벤트를 사용하지 않아도 됨

## ■ jQuery코드가 <HEAD> 태그 내에 위치할 경우

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="UTF-8">
  <script src="script/jquery-3.5.1.js"></script>
  <style>
    div {width:200px; height:200px; background-color:#00ffff}
  </style>

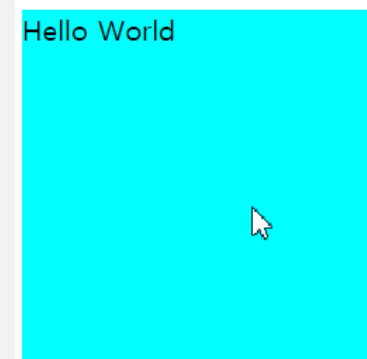
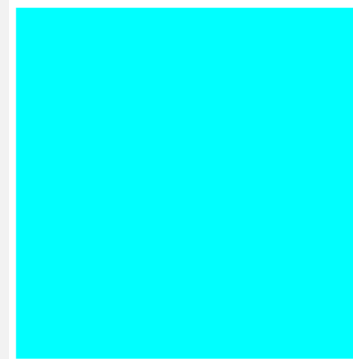
  <script type="text/javascript">

    jQuery(document).ready( function() {
      $('div').click(function(e) {
        $(this).text("Hello World");
      });
    });

  </script>

</head>
<body>
  <div> </div>
</body>
</html>
```

6-jquery-1.htm



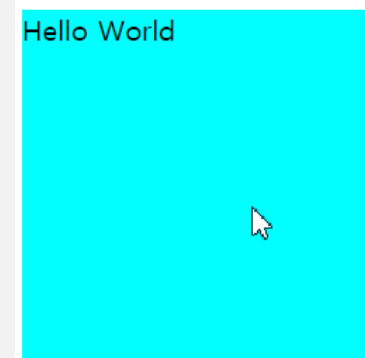
## ■ jQuery코드가 <BODY> 태그 마지막 부분에 위치할 경우

6-jquery-2.htm

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="UTF-8">
  <script src="script/jquery-3.5.1.js"></script>
  <style>
    div {width:200px; height:200px; background-color:#00ffff}
  </style>
</head>
<body>

  <div> </div>

  <script>
    $('div').click(function(e) {
      $(this).text("Hello World");
    });
  </script>
</body>
</html>
```



# 문서 준비 이벤트

## ■ 문서 준비 이벤트의 축약 표현

```
$(document).ready( function( ) {  
    jQuery 구문  
} );
```

=

```
$( function( ) {  
    jQuery 구문  
} );
```

```
<script>  
    jQuery(document).ready( function() {  
        $('div').click(function(e) {  
            $(this).text("Hello World");  
        });  
    });  
</script>
```

=

```
<script>  
    $( function() {  
        $('div').click(function(e) {  
            $(this).text("Hello World");  
        });  
    });  
</script>
```



## ■ jQuery 구문 HTML 문서 외부에 존재할 경우

- jQuery 구문을 별도의 자바스크립트 파일로 작성하고 문서에 연결하여 사용할 수 있음
  - jQuery 구문을 별도의 자바스크립트 파일로 작성
  - <script> 태그를 사용해 HTML 문서에 연결

```
<script src = "jQuery파일"></script>
```

- jQuery 구문을 연결하는 <script>를 <HEAD> 내에 위치시킬 경우
  - 연결하는 .js 파일 내에 반드시 문서 준비 이벤트를 사용해야 함
- jQuery 구문을 연결하는 <script>를 <BODY>의 마지막에 위치시킬 경우
  - 연결하는 .js 파일 내에 문서 준비 이벤트를 사용하지 않음

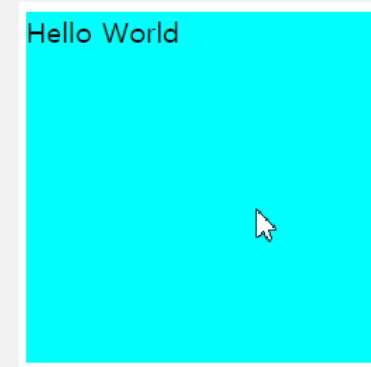
# 문서 준비 이벤트

## 6-jquery-3-sub.htm

```
$(document).ready( function() {  
    $('div').click(function(e) {  
        $(this).text("Hello World");  
    });  
});
```

## 6-jquery-3.htm

```
<!DOCTYPE html>  
<html>  
<head>  
    <meta charset="UTF-8">  
    <style>  
        div {width:200px; height:200px; background-color:#00ffff}</style>  
  
    <script src="script/jquery-3.5.1.js"></script>  
    <script src="6-jquery-3-sub.js"></script>  
  
</head>  
<body>  
    <div> </div>  
</body>  
</html>
```



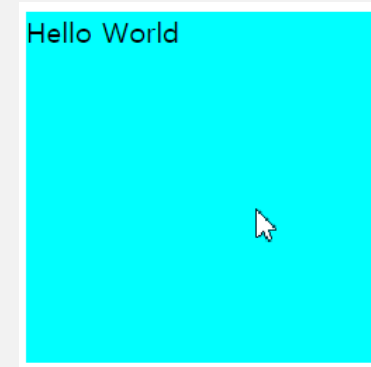
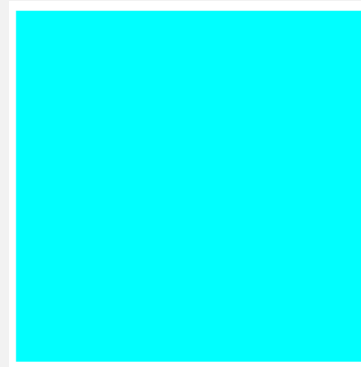
# 문서 준비 이벤트

6-jquery-4-sub.htm

```
$('#div').click(function(e) {  
    $(this).text("Hello World");  
});
```

6-jquery-4.htm

```
<!DOCTYPE html>  
<html>  
<head>  
    <meta charset="UTF-8">  
    <style>  
        div{width:200px; height:200px; background-color:#00ffff}  
    </style>  
  
    <script src="script/jquery-3.5.1.js"></script>  
</head>  
<body>  
    <div> </div>  
  
    <script src="6-jquery-3-sub.js"> </script>  
  
</body>  
</html>
```



# 자바스크립트와 jquery 사용 비교의 예

6-javascript-vs-jquery-1.htm

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="UTF-8">
  <script type="text/javascript">

    window.onload = function(){

      var main = document.getElementsByTagName("div");
      var ulList = main[0].getElementsByTagName("ul");
      var liList = ulList[0].getElementsByTagName("li");

      for(var i=0;i<liList.length;i++) {

        if(liList[i].getAttribute("class") == "current") {
          liList[i].style.backgroundColor = "#FF0000";
          break;
        }
      }
    }

  </script>
</head>
```

# 자바스크립트와 jquery 사용 비교의 예

```
<body>

  <h2> Lorem Ipsum </h2>

  <div id = "main">
    <p> This is paragraph </p>
    <ul id = "navi">
      <li> item 1 </li>
      <li class="current"> item 2 </li>
      <li> item 3 </li>
    </ul>
  </div>

</body>
</html>
```

## Lorem Ipsum

This is paragraph

- item 1
- item 2
- item 3

# 자바스크립트와 jquery 사용 비교의 예

6-javascript-vs-jquery-1.htm

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="UTF-8">
  <script src="script/jquery-3.5.1.js"></script>
  <script type="text/javascript">
    $(function() {
      jQuery("div li.current").css("background-color","#F00");
    });
  </script>
</head>

<body>
  <h2> Lorem Ipsum </h2>
  <div id = "main">
    <p> This is paragraph </p>
    <ul id = "navi">
      <li> item 1 </li>
      <li class="current"> item 2 </li>
      <li> item 3 </li>
    </ul>
  </div>
</body>
</html>
```

## Lorem Ipsum

This is paragraph

- item 1
- **item 2**
- item 3

**수고하셨습니다**