

jQuery 시작하기

#01. jQuery 초기화

예전 스타일

```
<!DOCTYPE html>
<html lang="ko">
<head>
    ... meta태그 설정 및 CSS 처리 ...
</head>

<body>
    ... HTML 태그 ...

    <!-- jQuery 참조 -->
    <script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.6.4/jquery.min.js">
</script>
    <!-- 개발자 작성 영역 -->
    <script type="text/javascript">
/** 모든 곳에서 인식할 변수 및 함수 정의 영역 */

/** jQuery 구현 부분 */
$(function() {
    // 이 영역이 페이지 로딩이 완료된 후 실행된다.
    // <script> 태그가 body 종료 직전에 있을 경우 생략 가능
});
</script>
</body>
</html>
```

화살표 함수 적용

```
<!DOCTYPE html>
<html lang="ko">
<head>
    ... meta태그 설정 및 CSS 처리 ...
</head>

<body>
    ... HTML 태그 ...

    <!-- jQuery 참조 -->
    <script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.6.4/jquery.min.js">
</script>
```

```

<!-- 개발자 작성 영역 -->
<script type="text/javascript">
/** 모든 곳에서 인식할 변수 및 함수 정의 영역 */

/** jQuery 구현 부분 ==> 화살표 함수 사용 */
$(() => {
    // 이 영역이 페이지 로딩이 완료된 후 실행된다.
});
</script>
</body>
</html>

```

초기화 함수 없이 바로 적용하기

javascript 코드를 body태그 종료 직전에 삽입하는 경우 HTML에 대한 로딩이 모두 이루어진 이후에 Javascript가 동작하기 때문에 별도의 초기화 함수 없이 바로 코드를 진행해도 된다.

```

<!DOCTYPE html>
<html lang="ko">
<head>
    ... meta태그 설정 및 CSS 처리 ...
</head>

<body>
    ... HTML 태그 ...

    <!-- jQuery 참조 -->
    <script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.6.4/jquery.min.js">
</script>
    <!-- 개발자 작성 영역 -->
    <script type="text/javascript">
        // 이 영역이 페이지 로딩이 완료된 후 실행된다.
    </script>
</body>
</html>

```

#02. HTML 요소 접근

HTML요소를 jQuery객체로 생성

```
const obj = $("CSS셀렉터");
```

- 대상 요소를 지정할 수 있는 CSS셀렉터를 사용한다.
- jQuery에서 사용한 CSS셀렉터가 반드시 <style>태그에 정의되어 있어야 하는 것은 아니다.

- 혼란을 피하기 위해 CSS에서는 class형식으로만 셀렉터를 구성하고 Javascript 에서는 가급적 id속성만으로 대상을 지정한다.

HTML요소 내의 내용 제어

`html()` 함수는 파라미터가 있을 경우 설정(setter), 파라미터가 없을 경우 리턴(getter)의 기능을 한다.

요소 안에 내용을 설정하기

```
$("#foo").html(설정할 내용);
```

요소 안의 내용을 가져오기

```
const content = $("#foo").html();
```