



Chapter 05-3

객체-3 (브라우저 객체 모델)



forms 객체

forms 객체

■ forms 객체

- <form> 태그와 그 구성요소(element)들의 정보를 추출하고 관리하는 객체
 - 자바스크립트에서의 forms 객체는 많은 하위 객체를 포함하고 있음
- <form> 객체가 가지는 하위 객체
 - text 객체, password 객체, textarea 객체, checkbox 객체, radio 객체, select 객체 등
- 각각의 하위 객체들은 유사하거나 고유한 속성과 메소드를 가짐

forms 객체의 속성

■ forms 객체의 속성

속성	설명
name	<form> 태그에서 지정한 name 속성의 값을 반환
action	<form> 태그에서 지정한 action 정보를 반환
method	<form> 태그에서 지정한 method 정보를 반환 (post 또는 get)
encoding	<form> 태그에서 지정한 인코딩 방식을 반환 - POST 방식의 데이터는 인코딩되어 전송됨 - 기본 인코딩 방식은 application/x-www-form-urlencoded임
length	<form> 태그 내에 존재하는 구성 요소들의 수를 반환 - 텍스트 입력상자, 텍스트영역 입력상자, 각종 버튼, 라디오 버튼, 체크박스 버튼, 리스트박스를 모두 포함

forms 객체의 메서드

■ submit() 메서드

- <form> 태그 내의 모든 입력 요소에 입력된 값을 action 속성이 지정하는 문서로 전송

```
document.myform.submit()
```

- :myform이라는 이름을 가진 <form>태그에 입력된 내용을 전송
 - 자바스크립트를 사용해 사용자 입력 오류를 모두 조사한 다음 데이터를 서버로 전송하기 위해 사용

■ reset() 메서드

- <form> 태그 내의 모든 입력 요소에 입력된 값을 초기화 시키기 위한 메서드

```
document.myform.reset()
```

- myform이라는 이름을 가진 <FORM>태그에 입력된 내용을 모두 지움

<form> 요소의 배열 표현

■ <form> 요소와 하위 구성요소의 배열 표현

• <form> 요소의 배열 표현

```
document.forms[ ]
```

- 현재 문서에 하나 이상의 <form> 태그가 존재하는 경우, 모든 <form> 태그를 forms라는 배열로 반환
- [예] document.forms[0]
 - 현재 문서에 존재하는 첫 번째 <form> 태그를 의미

• <form> 하위 요소의 배열 표현

```
document.elements[ ]
```

- <form> 태그에 존재하는 모든 구성요소를 나타나는 순서에 따라 element라는 배열로 반환
- [예] document.myform.element[0].value
 - 현재 문서에 존재하는 myform이라는 이름의 <form> 요소의 첫 번째 하위 요소에 입력된 값

forms 객체의 이벤트

■ forms 객체에서 주로 사용되는 이벤트

- `onSubmit`

- `<form>` 태그 내의 구성요소에 입력된 데이터가 서버로 전송될 때 실행되는 이벤트
 - `submit()` 메서드가 실행될 때 자료가 전송되기 직전에 수행됨

- `onReset`

- `<form>` 태그 내의 구성요소에 입력된 데이터가 초기화될 때 실행되는 이벤트
 - `reset()` 메서드가 실행될 때 자료가 초기화되기 직전에 수행됨

forms 객체의 예

5-forms-1.htm

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="UTF-8">
</head>
<body>

  <form name="myform" method=POST action="#" onReset="alert('데이터가 초기화됩니다');">

    <table BORDER=1 CELLSPACING=0 CELLPADDING=3>
      <tr>
        <td width=100 align=center>이름</td>
        <td width=400> <input type=text name="user_name" size=30> </td>
      </tr>
      <tr>
        <td width=100 align=center>아이디</td>
        <td width=400> <input type=text name="user_id" size=30> </td>
      </tr>
      <tr>
        <td width=100 align=center>패스워드</td>
        <td width=400> <INPUT TYPE=PASSWORD name="user_pass" size=30> </td>
      </tr>
    </table>
```


forms 객체의 예

```
<table border=0 cellspacing=0 cellpadding=3>
  <tr>
    <td width=500 align=center>
      <input type=button VALUE="전송" name="send" onClick="submit()">&nbsp;
      <input type=button VALUE="취소" name="clear" onClick="reset()">&nbsp;
      <input type=button VALUE="form정보확인" name="info" onClick="check()">
    </td>
  </tr>
</table>

</form>

<script type="text/javascript">
  function check() {
    str = "form의 action 속성 정보 : " + document.forms[0].action + "\n";
    str = str + "form의 method 속성 정보 : " + document.forms[0].method + "\n";
    str = str + "form의 name 속성 정보 : " + document.forms[0].name + "\n";
    str = str + "form의 encoding 속성 정보 : " + document.myform.encoding + "\n";
    str = str + "form의 구성요소의 수 : " + document.myform.length + "\n";
    str = str + "form의 두번째 구성요소의 이름 : " + document.myform.elements[1].name + "\n";
    alert(str);
  }
</script>
</body>
</html>
```

forms 객체의 예

이름	<input type="text" value="홍길동"/>
아이디	<input type="text" value="hong"/>
패스워드	<input type="password" value="*****"/>

localhost:8080 내용:

form의 action 속성 정보 : http://localhost:8080/JS_PRO/5-forms-1.htm#
form의 method 속성 정보 : post
form의 name 속성 정보 : myform
form의 encoding 속성 정보 : application/x-www-form-urlencoded
form의 구성요소의 수 : 6
form의 두번째 구성요소의 이름 : user_id

localhost:8080 내용:
데이터가 초기화됩니다

이름	<input type="text"/>
아이디	<input type="text"/>
패스워드	<input type="password"/>

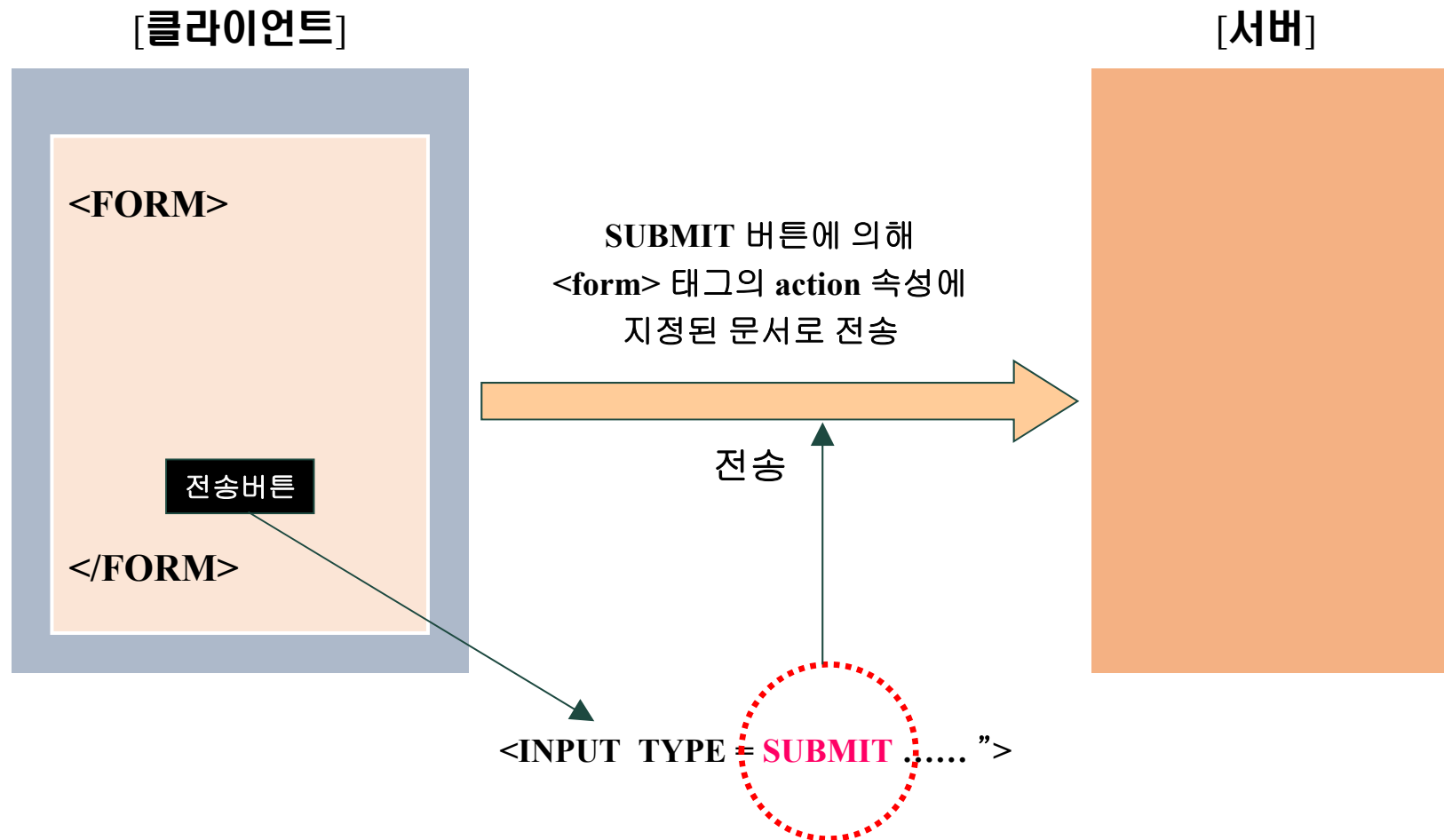
[참고] forms 객체에 자바스크립트 사용

■ 웹프로그래밍에서 자바스크립트를 사용해야 하는 이유

- 웹 프로그램을 작성할 때 프로그래밍의 주된 대상은 <FORM> 태그임
 - 사용자가 입력한 내용을 데이터베이스에 저장
 - 사용자가 요구한 내용을 데이터베이스로부터 추출해 출력
 - 사용자가 요구한 데이터를 데이터베이스로부터 제거
 - 사용자가 입력한 내용으로 기존 내용을 갱신
- 사용자가 입력한 데이터는 항상 올바른 형식은 아님
 - 패스워드의 제약, 전자우편 주소 형식, 문자와 숫자의 구분 등
- 서버의 부하를 최소화
 - 단순한 사용자 입력 오류로부터 서버의 부하를 경감시킴
 - 입력 데이터를 서버로 전송하기 전에 자바스크립트를 통해 클라이언트에서 최대한 오류 조사가 필요

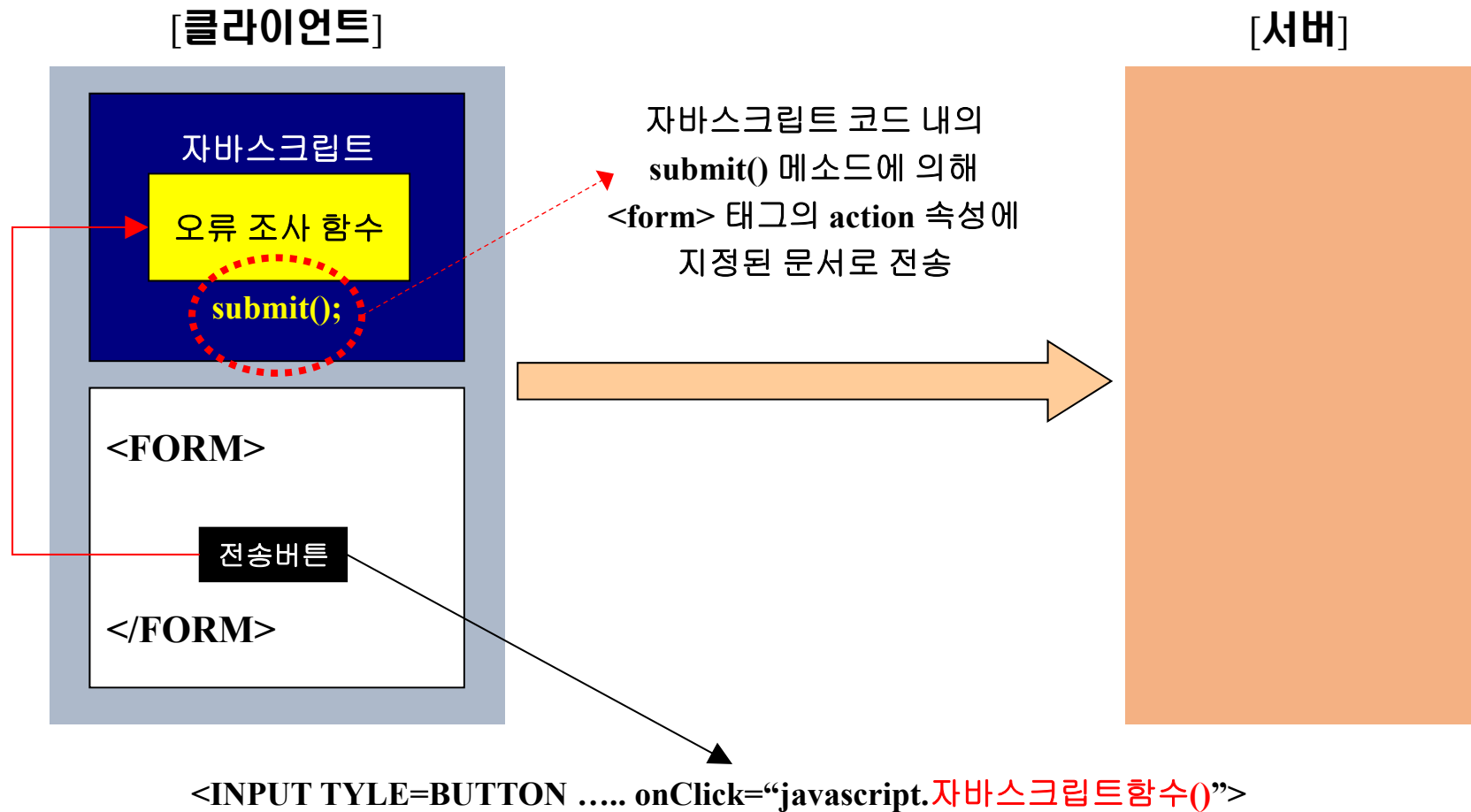
[참고] forms 객체에 자바스크립트 사용

■ 자바스크립트를 사용하지 않는 경우의 처리 방식



[참고] forms 객체에 자바스크립트 사용

■ 자바스크립트를 사용하는 경우의 처리 방식



[참고] forms 객체에 자바스크립트 사용

■ 자바스크립트를 사용한 데이터의 전달 방법 (인자를 사용하지 않는 경우)

5-forms-javascript-1.htm

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

    function check() {
      if( !myform.data.value ) {
        alert('값을 입력하세요');
        exit;
      }
      myform.submit();
    }

  </script>
</head>
<body>
  <form name="myform">
    <input type="text" name="data">
    <input type="button" value="전송" onClick="javascript:check()">
  </form>
</body>
</html>
```

[참고] forms 객체에 자바스크립트 사용

■ 자바스크립트를 사용한 데이터의 전달 방법 (form의 이름을 인자로 사용하는 경우)

5-forms-javascript-2.htm

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

    function check(form) {
      if( !form.data.value ) {
        alert('값을 입력하세요');
        exit;
      }
      form.submit();
    }

  </script>
</head>
<body>
  <form name="myform">
    <input type="text" name="data">
    <input type="button" value="전송" onClick="javascript:check(myform)">
  </form>
</body>
</html>
```

[참고] forms 객체에 자바스크립트 사용

■ 자바스크립트를 사용한 데이터의 전달 방법 (this를 인자로 사용하는 경우)

5-forms-javascript-3.htm

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

    function check(form) {
      if( !form.data.value ) {
        alert('값을 입력하세요');
        exit;
      }
      form.submit();
    }

  </script>
</head>
<body>
  <form name="myform">
    <input type="text" name="data">
    <input type="button" value="전송" onClick="javascript:check(this.form)">
  </form>
</body>
</html>
```


수고하셨습니다