

폼과 자바스크립트



소프트웨어학부
진혜진

목차

1. 폼 요소에 접근하는 여러 가지 방법
2. 폼 요소에서 입력 값 검증하기
3. 다양한 폼 요소 다루기



KMU 국민대학교
KOOKMIN UNIVERSITY

Kookmin
Total Information Systems
국민대학교 종합정보시스템입니다

■ 종합정보시스템 로그인

User ID

Password

Login ☒ ID 저장

■ 사번/학번 조회

■ 비밀번호 조회/변경

로그인 관련

- ◆ 종합정보시스템의 사용자입니다. (단, 대학원생은)
- ◆ 신(편)입생 및 신규 임용된 최초 로그인 시, [비밀번호] (단, 개인정보보호를 위해) 실 수 없으며, 안전한 비밀
- ◆ 평생교육원 학점은행제 학 사용하시는 비밀번호로 로 템에서 조회하여야 합니다
- ◆ [필독] 종합정보시스

1. 폼 요소에 접근하는 여러 가지 방법

3

□ 배송 정보 자동으로 입력하기

주문 정보

이름 :

연락처 :

주소 :

배송 정보

☐ 주문 정보와 배송 정보가 같습니다

이름 :

연락처 :

주소 :

결제하기

주문 정보

이름 :

연락처 :

주소 :

배송 정보

☒ 주문 정보와 배송 정보가 같습니다

이름 :

연락처 :

주소 :

결제하기

```
<!DOCTYPE html>
<head>
  <link rel="stylesheet" href="css/order.css">
</head>
<body>
  <div id="container">
    <form name="order">
      <fieldset>
        <legend> 주문 정보 </legend>
        <ul>
          <li>
            <label class="field" for="billingName">이름 : </label>
            <input type="text" class="input-box" id="billingName"
              name="billingName">
          </li>
          <li>
            <label class="field" for="billingTel">연락처 : </label>
            <input type="text" class="input-box" id="billingTel"
              name="billingTel">
          </li>
          <li>
            <label class="field" for="billingAddr">주소 : </label>
            <input type="text" class="input-box" id="billingAddr"
              name="billingAddr">
          </li>
        </ul>
      </fieldset>
    </form>
  </div>
</body>
</html>
```

```
<form name="ship">
  <fieldset>
    <legend> 배송 정보 </legend>
    <ul>
      <li>
        <input type="checkbox" id="shippingInfo" name="shippingInfo">
        <label for="bill_info" class="check">주문 정보와 배송
        정보가 같습니다</label>
      </li>
      <li>
        <label class="field" for="shippingName">이름 : </label>
        <input type="text" class="input-box" id="shippingName"
        name="shippingName">
      </li>
      <li>
        <label class="field" for="shippingTel">연락처 : </label>
        <input type="text" class="input-box" id="shippingTel"
        name="shippingTel">
      </li>
      <li>
        <label class="field" for="shippingAddr">주소 : </label>
        <input type="text" class="input-box" id="shippingAddr"
        name="shippingAddr">
      </li>
    </ul>
  </fieldset>
  <button type="submit" class="order">결제하기</button>
</form>
</div>
```

```
#container {  
  width:500px;  
  margin:auto;  
}  
#login_input {  
  float:left;  
}  
input[type="text"], input[type="password"]{  
  
  display:block;  
  width:300px;  
  height:35px;  
  line-height:35px;  
  border:1px solid #ccc;  
  background-color:#eee;  
}  
#login_btn > button {  
  float:left;  
  width:100px;  
  height:78px;  
  background-color:#222;  
  color:white;  
  border:none;  
}
```

1. 폼 요소에 접근하는 여러 가지 방법

7

- 폼 요소에 입력한 내용을 가져와 다른 폼 요소로 복사하는 기능을 가진 자바스크립트 프로그램 만들기
- id 값이나 class 값을 사용해 폼 요소에 접근하기
 - DOM의 다른 요소에 접근하는 것과 같음
 - `querySelector()`함수나 `querySelectorAll()`함수를 사용해 특정 id 값이나 class 값을 가진 요소에 접근할 수 있음
 - id 값을 사용할 때는 폼 요소 하나에만 접근하고, class 값을 사용할 때는 여러 요소를 가져와 배열 형태로 저장함

1. 폼 요소에 접근하는 여러 가지 방법

8

```
<li>
  <label class="field" for="billingName">이름 : </label>
  <input type="text" class="input-box" id="billingName" name="billingName">
</li>
```

- id 값이 billingName인 요소를 찾아 그 소스를 표시
- id 값을 이용해 접근할 때는 id 값 앞에 #을 붙이고, class 값을 사용해 접근할 때는 class값 앞에 .를 붙임

```
> document.querySelector("#billingName")
<
  <input type="text" class="input-box" id=
    "billingName" name="billingName">
```


1. 폼 요소에 접근하는 여러 가지 방법

9

- 텍스트 필드에 있는 값을 가져오기 위해서는 텍스트 필드에 접근하는 소스 뒤에 value 속성을 붙임
- 웹 브라우저 창의 이름 항목을 입력한 후 콘솔 창에서 확인

주문 정보

이름 :

연락처 :

주소 :

top

Filter Default

```
> document.querySelector("#billingName")  
< <input type="text" class="input-box" id="billingName" name="billingName">  
> document.querySelector("#billingName").value  
< "진혜진"
```

- 이름 항목에 입력한 값을 가져와서 표시함

1. 폼 요소에 접근하는 여러 가지 방법

10

□ name 값을 사용해 폼 요소에 접근하기

- 폼 요소에 name 속성만 있다면 name 식별자를 사용해 폼 요소에 접근할 수 있음
- 이 방법을 사용하려면 <form> 태그에 name 속성이 지정되어 있어야 하고, <form> 태그 안에 포함된 폼 요소에도 name 속성이 있어야 함
- name 속성을 사용해 폼 요소에 접근하려면 form의 name 값부터 폼 요소의 name값까지 계층을 따라 하나씩 지정해 줌

1. 폼 요소에 접근하는 여러 가지 방법

11

- 폼 안에 있는 텍스트 필드에 접근하려면 <form>의 name 값과 텍스트 필드의 name 값을 사용하면 됨

```
<form name="ship">
```

```
<fieldset>
```

```
<legend> 배송 정보
```

```
<ul>
```

```
<li>
```

```
<input type="checkbox" id="shippingInfo" name="shippingInfo">
```

```
<label for="bill_info" class="check">주문 정보와 배송  
정보가 같습니다</label>
```

```
</li>
```

```
<li>
```

```
<label class="field" for="shippingName">이름 : </label>  
<input type="text" class="input-box" id="shippingName"  
name="shippingName">
```

```
</li>
```

```
> document.ship.shippingName
```

```
< <input type="text" class="input-box" id=  
"shippingName" name="shippingName">
```

```
> document.forms["ship"].elements["shippingName"]
```

```
< <input type="text" class="input-box" id=  
"shippingName" name="shippingName">
```

1. 폼 요소에 접근하는 여러 가지 방법

12

□ 배송 정보의 이름 항목에 입력한 값을 가져옴

이름 :	<input type="text"/>
연락처 :	<input type="text"/>
주소 :	<input type="text"/>
배송 정보	
<input type="checkbox"/> 주문 정보와 배송 정보가 같습니다	
이름 :	<input type="text" value="진혜진"/>

```
> document.ship.shippingName.value  
< "진혜진"  
> document.forms["ship"].elements["shippingName"].value  
< "진혜진"  
> |
```

□ name 속성을 사용해 접근하려면 <form> 태그뿐만 아니라 접근하려는 폼 요소에 모두 name 속성이 지정되어 있는지 확인해야 함

1. 폼 요소에 접근하는 여러 가지 방법

13

□ 폼 배열을 사용해 폼 요소에 접근하기

- document 속성 중 forms 속성은 문서 안에 있는 `<form>` 태그를 모두 가져와 배열 형태로 반환함
- 이 방법은 폼 요소에 id나 class 속성도 없고, name 속성도 없을 때 사용함

```
<div id="container">
```

```
  <h3>로그인</h3>
```

```
  <form>
```

```
    <div id="login_input">
```

```
      <input type="text" autofocus placeholder="아이디">
```

```
      <input type="password" placeholder="비밀번호">
```

```
    </div>
```

```
    <div id="login_btn">
```

```
      <button type="submit" class="order">로그인</button>
```

```
    </div>
```

```
  </form>
```

```
</div>
```

로그인

아이디

비밀번호

로그인

1. 폼 요소에 접근하는 여러 가지 방법

14

- form 요소가 하나이기 때문에 숫자 없이 HTMLCollection 이라고 표시됨
- <form> 태그가 2개라면 HTMLCollection[2]처럼 괄호 안에 form 요소의 개수가 표시됨
- <form> 태그 안에 포함된 요소에 접근하려면 elements 속성을 사용함
 - ▣ 해당 폼 안에 있는 폼 요소를 모두 가져오는 속성임

```
> document.forms
< ▶ HTMLCollection [form]
> document.forms[0].elements
< ▼ HTMLFormControlsCollection(3) [input, input, button.order]
  ▶ 0: input
  ▶ 1: input
  ▶ 2: button.order
  length: 3
  ▶ __proto__: HTMLFormControlsCollection
```

로그인

아이디

비밀번호

로그인

1. 폼 요소에 접근하는 여러 가지 방법

15

- 첫 번째 텍스트 필드에 아이디, 두 번째 필드에 비밀번호 입력
 - ▣ 폼 배열을 사용한 소스 코드를 입력하면 첫 번째 폼의 첫 번째 요소, 즉 입력한 아이디 값을 가져올 수 있음

로그인

kookmin	로그인
....	

Elements Console Sources

top

```
> document.forms[0].elements[0].value  
< "kookmin"  
  
> document.forms[0].elements[1].value  
< "1234"
```

```

</li>
  <input type="checkbox" id="shippingInfo" name="shippingInfo">
  <label for="shippingInfo" class="check">주문 정보와 배송 정보가
  같습니다</label>
</li>

```

□ 배송 정보 자동 입력 프로그램 만들기

▣ 주문 정보 내용 가져오기

```
var check = document.querySelector("#shippingInfo"); // 체크박스의 id는 shippingInfo
```

```

check.addEventListener("click", function() { // check 요소에 click 이벤트가 발생했을 때 실행할 함수
  if(check.checked == true) { // 체크되었다면
    document.querySelector("#shippingName").value = document.querySelector("#billingName").value;
    // 주문 정보(이름)를 배송 정보(이름)에 복사
    document.querySelector("#shippingTel").value = document.querySelector("#billingTel").value;
    // 주문 정보(전화번호)를 배송 정보(전화번호)에 복사
    document.querySelector("#shippingAddr").value = addrInfo = document.querySelector("#billingAddr").value;
    // 주문 정보(주소)를 배송 정보(주소)에 복사
  }
  else { // 체크되어 있지 않다면 배송 정보 필드를 지움.
    document.querySelector("#shippingName").value = "";
    document.querySelector("#shippingTel").value = "";
    document.querySelector("#shippingAddr").value = "";
  }
});

```

배송 정보

☐ 주문 정보와 배송 정보가 같습니다

이름 :

연락처 :

주소 :

2. 폼 요소에서 입력 값 검증하기

17

- 자바스크립트를 여러 분야에서 사용하기 전에는 주로 폼에 입력한 값의 유효성을 검사할 때 사용했음
- Form Validation(입력 값 검증)
 - ▣ 값이 입력되었는지, 입력된 값이 정해진 조건에 맞는지 등을 검사하는 작업

회원 가입

아이디 4~15자리의 영문과 숫자로 입력

이메일

비밀번호 8자리 이상

비밀번호 확인

메일링 수신 ☐ 예 ☐ 아니오

가입하기

취소

아이디

WEB

이 페이지 내용:

4~15자리의 영문과 숫자를 사용하세요.

확인


```
<li>
  <label class="field">메일링 수신</label>
  <label class="r"><input type="radio" value="yes" name="
mailing">예</label>
  <label class="r"><input type="radio" value="no" name="
mailing">아니오</label>
</li>
</ul>
<ul id="buttons">
  <li>
    <button type="submit" class="btn btnBlack">가입하기</button>
  </li>
  <li>
    <button type="reseet" class="btn btnGray">취소</button>
  </li>
</ul>
</form>
</div>

<script src="js/register-result.js"></script>
</body>
</html>
```

2. 폼 요소에서 입력 값 검증하기

20

- 아이디 글자 수 확인하기
 - ▣ 아이디 필드를 가져와 변수에 저장하는 소스 코드

```
// '아이디' 필드를 가져와 변수에 저장  
var userId = document.querySelector("#user-id");
```

```
<li>  
  <label for="user-id" class="field">아이디</label>  
  <input type="text" id="user-id" placeholder="4~15자리의  
  영문과 숫자로 입력" required>  
</li>
```

아이디

이 페이지 내용:
4~15자리의 영문과 숫자를 사용하세요.

확인

□ 아이디 필드에 내용을 입력하면 바로 입력한 글자 수를 확인하도록 소스코드 작성

- userID 요소에서 change 이벤트가 발생하면 checkID() 함수를 실행하도록 지정

```
userId.onChange = checkId;
```

- change 이벤트

- 텍스트 필드 안의 내용이 바뀔 때, 즉 아이디 필드에 입력을 마치고 그 필드를 빠져나올 때 발생하는 이벤트

- select()함수

- 텍스트 필드에 입력한 내용을 선택함

```
function checkId() {
  // userId 필드 내용의 길이가 4 이하이거나 15 이상일 경우 실행
  if (userId.value.length < 4 || userId.value.length > 15) {
    alert("4~15자리의 영문과 숫자를 사용하세요."); // 오류 메시지 출력
    userId.select(); // 다시 입력할 수 있도록 userId 필드 선택
  }
}
```

2. 폼 요소에서 입력 값 검증하기

22

□ 비밀번호 확인하기

- ▣ 비밀번호 필드와 비밀번호 확인 필드 요소 가져와 변수에 저장

```
// '비밀번호' 필드를 가져와 변수에 저장  
var pw1 = document.querySelector("#user-pw1");  
// '비밀번호 확인' 필드를 가져와 변수에 저장  
var pw2 = document.querySelector("#user-pw2");
```

```
<li>  
  <label for="user-pw1" class="field">비밀번호</label>  
  <input type="password" id="user-pw1" placeholder="8자리 이상" required>  
</li>  
<li>  
  <label for="user-pw2" class="field">비밀번호 확인</label>  
  <input type="password" id="user-pw2" required>  
</li>
```

2. 폼 요소에서 입력 값 검증하기

23

- pw1 요소에 change 이벤트가 발생했을 때 checkPw() 함수 실행하도록 소스코드 추가

```
pw1.onChange = checkPw;
```

```
function checkPw() {  
    if (pw1.value.length < 8) {  
        alert("비밀번호는 8자리 이상이어야 합니다."); // 오류 메시지 표시  
        pw1.value = ""; // '비밀번호' 필드 지움  
        pw1.focus(); // 비밀번호를 다시 입력할 수 있게 포커싱  
    }  
}
```

- focus()

- ▣ 해당 요소에 포커스를 맞추는 자바스크립트 내장 함수임
- ▣ 텍스트 필드에서 사용할 경우 해당 텍스트 필드 안에 커서를 갖다 놓음

2. 폼 요소에서 입력 값 검증하기

24

- 텍스트 필드를 검증해서 오류가 발생했을 때 사용자가 입력한 값을 어떻게 처리할지에 따라 `select()` 함수와 `focus()` 함수를 사용할 수 있음
- `checkID()` 함수에서 사용한 `select()` 함수
 - ▣ 사용자가 기존에 입력한 값을 선택
- `checkPw()` 함수에서 사용한 `focus()` 함수
 - ▣ 기존에 입력한 값이 지워진 자리에 새로운 값을 입력하더라도 텍스트 필드에 커서를 갖다 놓음

2. 폼 요소에서 입력 값 검증하기

25

- 비밀번호 확인 필드에 입력한 내용이 비밀번호 필드에 입력한 내용과 같은 지 확인

```
pw2.onchange = comparePw;
```

```
function comparePw() {  
    if(pw1.value != pw2.value) {  
        alert("암호가 다릅니다. 다시 입력하세요.");  
        pw2.value = ""; // '비밀번호 확인' 필드 지움  
        pw2.focus();    // 비밀번호를 다시 입력할 수 있게 포커싱  
    }  
}
```

2. 폼 요소에서 입력 값 검증하기

26

이 페이지 내용:

비밀번호는 8자리 이상이어야 합니다.

확인

이메일

hjjin@kookmin.ac.kr

비밀번호

....

원

이 페이지 내용:

암호가 다릅니다. 다시 입력하세요.

확인

이메일

hjjin@kookmin.ac.kr

비밀번호

.....

비밀번호 확인

.....|

2. 폼 요소에서 입력 값 검증하기

27

□ 태그 자체에서 폼 검증하기

- HTML5의 <input> 태그에 폼 검증을 위한 속성이 추가되면서 자바스크립트 대신 type 속성 값을 사용해서 필드 값을 체크할 수 있음
- <input> 태그의 새로운 유형
 - <input type="email"> : 이메일 주소 필드
 - <input type="url"> : 사이트 주소필드

이메일

HJJIN|

비밀번호



이메일 주소에 '@'를 포함해 주세요. 'HJJIN'에 '@'가 없습니다.

사이트주소

www.kookmin.ac.kr

가입하기



URL을 입력하세요.

취소

2. 폼 요소에서 입력 값 검증하기

28

□ <input> 태그의 새로운 속성

▣ autocomplete="off"

- 자동 완성 기능을 켜고 끄는 속성

▣ required

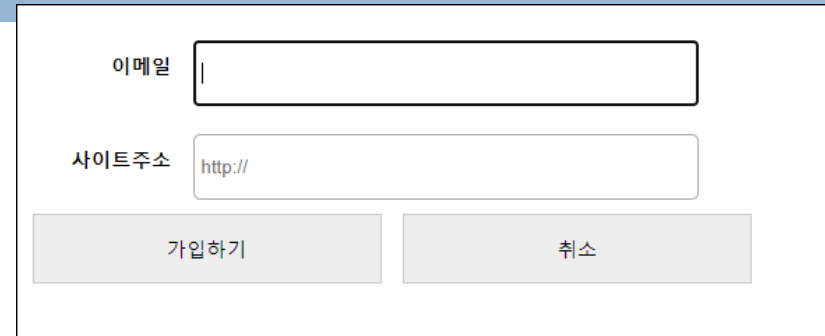
- 필수 입력항목으로 지정, 필드가 작성되지 않으면 오류 메시지를 표시하며 다음 단계로 넘어갈 수 없음

▣ placeholder

- 필드 안에 힌트가 표시되어 사용자에게 어떤 내용을 입력해야 하는지 알려줌

▣ autofocus

- 해당 필드에 마우스 커서를 자동으로 표시함



The image shows a web form with two input fields and two buttons. The first input field is labeled '이메일' (Email) and is empty. The second input field is labeled '사이트주소' (Site Address) and contains the text 'http://'. Below the input fields are two buttons: '가입하기' (Sign Up) and '취소' (Cancel).

2. 폼 요소에서 입력 값 검증하기

29

```
<form action="#" id="register">
```

```
  <ul id="user-info">
```

```
    <li>
```

```
      <label for="email" class="field">이메일</label>
```

```
      <input type="email" id="email" autocomplete="off" autofocus>
```

```
    </li>
```

```
    <li>
```

```
      <label for="url" class="field">사이트주소</label>
```

```
      <input type="url" id="url" required placeholder="http://">
```

```
    </li>
```

```
  </ul>
```

이메일

사이트주소

가입하기

취소

3. 다양한 폼 요소 다루기

30

- 자바스크립트를 사용하면 사용자가 폼에서 어떤 항목을 선택했는지 알아 낼 수 있음

- 수강 신청 페이지

- 선택목록
- 라디오버튼
- 체크박스

수강신청

학생 정보

이름

진혜진

학과

소프트웨어학부 ▼

신청 과목

2학기에 신청할 과목을 선택하세요.

☒ 웹클라이언트 ☐ 운영체제 ☐ 프로그래밍

신청 과목

2학기에 신청할 과목을 선택하세요.

☒ 웹클라이언트 ☒ 운영체제 ☒ 프로그래밍

3. 다양한 폼 요소 다루기

학과

----- 학과 선택 ----- ▼

학과

----- 학과 선택 ----- ▼
----- 학과 선택 -----
소프트웨어학부
건축공학부
기계공학부
전기전자공학부

31

□ 선택 목록 및 옵션 항목에 접근하기

▣ 선택 목록 : <select>

- ▣ 선택 목록은 <option>태그를 사용해 여러 항목을 한꺼번에 지정한 뒤 목록을 펼쳐 원하는 항목을 선택할 수 있는 요소임

```
<form name="testForm">  
  <fieldset>  
    <legend>학생 정보</legend>  
    <ul>  
      <li>  
        <label class="reg" for="userName">이름</label>  
        <input type="text" id="userName" name="userName" maxlength="50">  
      </li>  
      <li>  
        <label class="reg" for="class">학과</label>  
        <select name="major" id="major" onchange="displaySelect()">  
          <option>---- 학과 선택 ----</option>  
          <option value="software">소프트웨어학부</option>  
          <option value="archi">건축공학부</option>  
          <option value="mechanic">기계공학부</option>  
          <option value="elec">전기전자공학부</option>  
        </select>  
      </li>  
    </ul>  
  </fieldset>
```

3. 다양한 폼 요소 다루기

32

- name=major인 요소, 즉 선택 목록에 접근할 수 있음
- options
 - ▣ 옵션 항목에 접근할 수 있음

```
> document.testForm.major
< ▶ <select name="major" id="major" onchange=
    "displaySelect()">...</select>

> document.testForm.major.options
< ▶ HTMLOptionsCollection(5) [option, option, option,
    option, option, selectedIndex: 0] ⓘ
    ▶ 0: option
    ▶ 1: option
    ▶ 2: option
    ▶ 3: option
    ▶ 4: option
    length: 5
    selectedIndex: 0
    ▶ __proto__: HTMLOptionsCollection
```


3. 다양한 폼 요소 다루기

33

이름

학과

과목을 선택하세요.

☐ 운영체제 ☐ 프로그래밍

과목을 선택하세요.

☐ 운영체제 ☐ 프로그래밍

> document.testForm.major.options

< HTMLOptionsCollection(5) [option, option, option, option, option, selectedIndex: 1] ⓘ

▶ 0: option

▶ 1: option

▶ 2: option

▶ 3: option

▶ 4: option

length: 5

selectedIndex: 1

▶ __proto__: HTMLOptionsCollection

>

```
<select name="major" id="major" onchange="displaySelect()">
  <option>---- 학과 선택 ----</option>
  <option value="software">소프트웨어학부</option>
  <option value="archi">건축공학부</option>
```

- 선택 목록 중 두 번째 옵션 항목에 접근
- innerText
 - ▣ 두 번째 옵션이 화면에 표시되는 내용에 접근
- value
 - ▣ 두 번째 옵션 항목을 선택했을 때 어떤 값을 서버로 넘겨주는지 알 수 있음

```
> document.testForm.major.options[1]
< <option value="software">소프트웨어학부</option>
> document.testForm.major.options[1].innerText
< "소프트웨어학부"
> document.testForm.major.options[1].value
< "software"
```

3. 다양한 폼 요소 다루기

35

- 선택 목록에서 사용자가 선택한 옵션 항목 찾아내기
 - ▣ 자바스크립트로 선택 목록에 접근하면 배열 형태가 반환됨
 - ▣ 배열 안에 selectedIndex 속성이 있음
 - 이 속성 값을 살펴보면 사용자가 어떤 항목을 선택했는지 알 수 있음

이름 <input type="text"/>		> document.testForm.major.options.selectedIndex
학과 <input type="text" value="소프트웨어학부"/>		< 1

3. 다양한 폼 요소 다루기

이 페이지 내용:

[소프트웨어학부]를 선택했습니다.

확인

36

- ▣ <select>태그에 change이벤트가 발생했을 때 displaySelect()함수 실행
- ▣ selectMenu.selectedIndex
 - 사용자가 선택한 항목의 인덱스를 알아냄
- ▣ innerText
 - 해당 항목에서 화면에 표시할 내용을 알아냄

```
<select name="major" id="major" onchange="displaySelect()">  
  <option>---- 학과 선택 ----</option>  
  <option value="software">소프트웨어학부</option>
```

```
// 선택 메뉴를 가져와 selectMenu로 저장  
var selectMenu = document.testForm.major;  
function displaySelect() {  
  var selectedText = selectMenu.options[selectMenu.selectedIndex].innerText;  
  alert "[" + selectedText + "]를 선택했습니다.");  
}
```

3. 다양한 폼 요소 다루기

37

```
<form name="testForm">  
  <fieldset>  
    <legend>학생 정보</legend>
```

...

```
<fieldset>  
  <legend>신청 과목</legend>  
  <p>2학기에 신청할 과목을 선택하세요.</p>  
  <label><input type="radio" name="subject" value="webclient">웹클라이언트</label>  
  <label><input type="radio" name="subject" value="os">운영체제</label>  
  <label><input type="radio" name="subject" value="pgm">프로그래밍</label>  
</fieldset>
```

3. 다양한 폼 요소 다루기

38

□ 라디오 버튼과 체크 박스에 접근하기

학생 정보

이름

학과

신청 과목

2학기에 신청할 과목을 선택하십시오

☒ 웹클라이언트 ☐ 운영체제

신청 과목

2학기에 신청할 과목을 선택하십시오

☐ 웹클라이언트 ☐ 운영체제

```
> document.testForm.subject
```

```
< ▼ RadioNodeList(3) [input, input, input, value: ""] ⓘ
```

```
▶ 0: input
```

```
▶ 1: input
```

```
▶ 2: input
```

```
length: 3
```

```
value: ""
```

```
▶ __proto__: RadioNodeList
```

```
> document.testForm.subject
```

```
< ▼ RadioNodeList(3) [input, input, input, value: "webclient"]
```

```
▶ 0: input
```

```
▶ 1: input
```

```
▶ 2: input
```

```
length: 3
```

```
value: "webclient"
```

```
▶ __proto__: RadioNodeList
```

3. 다양한 폼 요소 다루기

학생 정보	
이름	<input type="text"/>
학과	<input type="text" value="---- 학과 선택"/>

신청 과목	
2학기에 신청할 과목을 선택하세요	
<input type="radio"/> 웹클라이언트 <input type="radio"/> 운영체제 <input type="radio"/> 프	

신청 과목	
2학기에 신청할 과목을 선택하세요	
<input type="checkbox"/> 웹클라이언트 <input type="checkbox"/> 운영체제 <input type="checkbox"/> 프	

```
> document.testForm.subject1
< <input type="checkbox" name="subject1" value="webclient">
> document.testForm.subject2
< <input type="checkbox" name="subject2" value="os">
> |
```

```
<fieldset>
  <legend>신청 과목</legend>
  <p>2학기에 신청할 과목을 선택하세요.</p>
  <label><input type="checkbox" name="subject1" value="webclient">
  웹클라이언트</label>
  <label><input type="checkbox" name="subject2" value="os">운영체제</label>
  >
  <label><input type="checkbox" name="subject3" value="pgm">프로그래밍</
  label>
</fieldset>
```

3. 다양한 폼 요소 다루기

40

이름 <input type="text"/>	> document.testForm.subject1.checked
학과 ----- 학과 선택	< true
신청 과목	> document.testForm.subject2.checked
2학기에 신청할 과목을 선택하세요	< false
<input type="radio"/> 웹클라이언트 <input checked="" type="radio"/> 운영체제 <input type="radio"/> 프	> document.testForm.subject[0].checked
	< false
신청 과목	> document.testForm.subject[1].checked
2학기에 신청할 과목을 선택하세요	< true
<input checked="" type="checkbox"/> 웹클라이언트 <input type="checkbox"/> 운영체제 <input type="checkbox"/> 프	>