

웹 프로그래밍 강의노트

8. 다양한 입력폼

2023년 2학기

폼 이해하기

〈form〉 요소의 사용

- 폼 요소의 사용
 - 회원가입, 상품구매, 키워드 검색 등 사용자로부터 정보를 받을 때
 - 사용자와 애플리케이션이 상호작용
 - 사용자 입력 ⇒ 전송버튼 ⇒ 애플리케이션에 전달 ⇒ 실행결과 반환
- 〈form〉 요소의 역할
 - 사용자가 입력하는 정보를 하나로 묶어서 애플리케이션에 전달
 - 다양한 입력 양식을 그룹핑하고 전송방법을 설정
 - 〈form〉 요소 내 사용자의 정보 입력 양식
 - 〈input〉, 〈textarea〉, 〈select〉, 〈button〉 등의 입력 요소를 이용

다양한 입력 폼 예제

<h3>다양한 입력 폼</h3>

<form method="get" action="form_app.js">

성명: <input type="text" name="person"/>

성별: <input type="radio" name="gender" value="male"/>남성
<input type="radio" name="gender" value="female"/>여성

직업: <select name="job" size="1">
<option>학생</option> <option>회사원</option>
<option>공무원</option> <option>기타</option>
</select>

<p> 구입희망분야(복수선택 가능)

- 분야: <input type="checkbox" name="books" value="computer"/>컴퓨터
<input type="checkbox" name="books" value="economy"/>경제
<input type="checkbox" name="books" value="common"/>상식

비고:

<textarea name="comments" rows="4" cols="40"/></textarea>

</p> <hr>

<input type="submit" value="신청"/>

<input type="reset" value="취소"/>

</form>

다양한 입력 폼

성명:

성별: ☒ 남성 ☐ 여성

직업:

구입희망분야(복수선택 가능)

분야: ☒ 컴퓨터 ☐ 경제 ☐ 상식

비고:

다양한 입력 폼

성명:

성별: ☒ 남성 ☐ 여성

직업:

구입희망분야(복수선택 가능)

분야: ☒ 컴퓨터 ☐ 경제 ☐ 상식

비고:

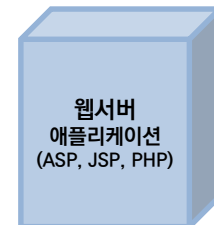
〈form〉 요소의 주요 속성

- 〈form name="이름" method="get/post" action="애플리케이션 주소"〉
 - method 속성 : 데이터 전송 방식 지정
 - get방식: 전송할 데이터를 URL 주소 뒤에 포함,
 - 예, `http://... ch04/app.js?person=%C8%AB%B1%E6&sex=male& jobs=...`
 - 간단한 데이터에는 편리, 그러나 보안에 취약
 - post방식: 프로그램의 입출력 방식을 사용, 데이터 양에 제한 없다
 - action 속성 : 데이터 처리할 애플리케이션 주소 (웹서버 경우 URL)
 - name속성 : 폼 요소에 대한 이름
- HTML 문서에서 〈form〉의 역할
 - 다양한 〈input〉 요소에 대해
 - 데이터 입력 ⇒ 전달 ⇒ 반환

GET/POST방식

요청(데이터 전송)

응답(실행결과)



기본 형식으로 입력하기

텍스트 입력

선택항목의 입력

버튼 입력

기타 입력 필드

입력 필드의 그룹핑

입력 폼의 형태

- 기본적인 입력 폼의 형태
 - <form> 요소 안에 <input>, <textarea>, <select>, <button> 등 요소
- <input> 요소의 속성
 - <input type="입력 형식" name="변수명", value="입력 값"/>
 - name 속성, value 속성
 - type 속성은 입력 폼의 유형
 - 텍스트 관련: text, password
 - 선택 관련 : radio, checkbox
 - 버튼 : submit, reset, button, image
 - 기타 : file, hidden

텍스트 입력

- 문자열 입력 필드 (한 줄)

〈input type="text" name="변수명" value="초기값" /〉

- name 속성 : 애플리케이션에 전달될 변수명
- value 속성 : 입력 받은 값을 전달 (입력창 초기문자)

- 암호 입력 필드

〈input type="password" name="변수명" /〉

- 보안이 필요한 문자 입력, ‘.’ 으로 표시 (더 이상 보호 기능 없음)

- 텍스트 영역 필드 (여러 줄): 〈textarea〉 요소

〈textarea name="이름" cols="열의 수" rows="행의 수"〉

텍스트 영역에 표시되는 초기 문장

〈/textarea〉

- 표시 영역보다 많은 텍스트를 입력하면 스크롤바

아이디와 비밀번호, 요청사항 입력

문자열, 암호 입력 및 텍스트 영역

```
<form method="post" action="form_app.js">
```

아이디 :

비밀번호:

〈p〉요청사항: 〈br〉

<textarea name="comment" cols="40" rows="5"> 전달하실 내용을 적으세요:

</textarea>

텍스트 입력 예제

아이디:

비밀번호:

요청사항:

전달하신 내용을 적으세요.

테스트 입력 예제

webclass.me/html5_3e/ch06/input2_text.html

문자열, 암호 입력 및 텍스트 영역

아이디 :

비밀번호 :

요청사항:

전달하실 내용을 적으세요:

손쉬운 경제 상식이 필요합니다. 도서관에 경제 상식을 쉽게 이해할 수 있는 교양 서적이 부족합니다. 학생들이 취업할 때 전공 상식 이외에 경제에 대한 상식도 많이 필요

요청하기

선택항목의 입력

• 라디오 버튼

`<input type="radio" name="변수명" value="선택값" checked />`

- name 속성값이 같은 그룹에서 value 값을 하나만 선택하여 전달
- 예, “member=yes&sex=male”의 형태로 전달

회원여부: `<input type="radio" name="member" value="yes" checked />`회원
`<input type="radio" name="member" value="no" />`비회원
성별 : `<input type="radio" name="sex" value="male" />`남성
`<input type="radio" name="sex" value="female" />`여성

회원여부: ☒ 회원 ☐ 비회원
성별 : ☐ 남성 ☐ 여성

• 체크박스 선택

`<input type="checkbox" name="변수명" value="선택값" checked />`

- 체크박스에 표시된 모든 value 속성값 들이 애플리케이션으로 전송

취미(중복선택) :

`<input type="checkbox" name="hobby" value="read" />`독서
`<input type="checkbox" name="hobby" value="movie" checked />`영화
`<input type="checkbox" name="hobby" value="music" />`음악
`<input type="checkbox" name="hobby" value="sports" />`스포츠

취미(중복선택) : ☐ 독서 ☒ 영화 ☒ 음악 ☒ 스포츠

- 선택목록에서 선택 : <select> 요소 내에 <option> 항목

- 드롭다운 형태(size= 1), 혹은 스크롤 박스 형태(size > 1)
- multiple 속성 : 다중 선택
- 각 항목은 <select> 요소 내에서 <option> 요소로 정의

```
<select name="job" size="1" >  
  <option value="student" selected>학생</option>  
  <option value="company">회사원</option>  
  <option value="teacher">교사</option>  
  <option value="sales">자영업</option>  
  <option value="others">기타</option>  
</select>
```

직업:

직업:
학생
회사원
교사
자영업
기타

<select size="4" multiple>

직업:
학생
회사원
교사
자영업

버튼 입력

- 전송 버튼 : 입력 데이터 전달

- `<input type="submit" value="버튼라벨" />`

- `<form>` 요소에 있는 모든 입력 데이터가 `<form>`의 `action` 속성에서 지정한 애플리케이션 프로그램으로 전송

- 초기화 버튼

- `<input type="reset" value="버튼라벨" />`

- 일반 버튼 : 다양한 용도로 사용

- `<input type="button" value="버튼라벨" />`

- 이미지 버튼

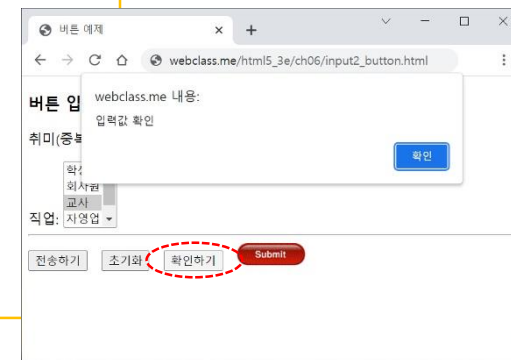
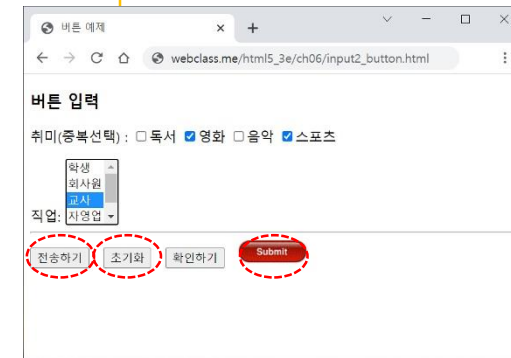
- `<input type="image" src="이미지 파일" alt="대체 문자열" />`

- `<button>` 요소 : `<button type="submit">버튼 라벨</button>`

```

<form method="get" action="form_app.js">
  <p>취미(중복선택) :
    <input type="checkbox" name="hobby" value="read"/>독서
    <input type="checkbox" name="hobby" value="movie" checked/>영화
    ...
  </p>
  직업:
  <select name="job" size="4" multiple>
    <option value="student" selected>학생</option>
    <option value="company">회사원</option>
    <option value="teacher">교사</option>
    ...
  </select>
  <hr>
  <input type="submit" value="전송하기"/> &nbsp;
  <input type="reset" value="초기화"/> &nbsp;
  <input type="button" value="확인하기" onClick="alert('입력값 확인')"/> &nbsp;
  <input type="image" src="help.gif" alt="전송 버튼"/>
</form>

```



기타 입력 필드

- 파일 선택하기 : `<input type="file" />`

파일 업로드하기 `
`
`<input type="file" name="myfile" />`

파일 업로드하기

파일 선택 선택된 파일 없음

파일 업로드하기

찾아보기...

- 데이터 숨김

`<input type="hidden" name="변수명" value=값" />`

- 시스템에서 특정 데이터를 처리하고 싶을 때, 예: 사용자 IP 주소

- 텍스트 라벨: 특정 입력 필드에 연결

`<label for="입력아이디">`

- 데이터 전달에는 영향이 없다

성별: `<label for="male">남성</label>`
`<input type="radio" name="gender" id="male" value="male">`
`<label for="female">여성</label>`
`<input type="radio" name="gender" id="female" value="female">`

텍스트 라벨을 클릭해도 됩니다.

성별: 남성 ☒ 여성 ☐

텍스트 라벨을 클릭해도 됩니다.

성별: 남성 ☐ 여성 ☒

입력 필드의 그룹핑

- 그룹핑 : <fieldset> 요소
 - 폼 양식을 그룹핑하는 범위를 지정
 - 사용자의 시각적 편의를 위해서 제공
 - 입력 필드의 그룹 주위에 기본 스타일로 테두리 선
 - name 속성 : 그룹의 이름을 지정하는
 - form 속성 : 폼과 연결
 - disabled 속성 : 그룹 내의 모든 하위 입력 요소들을 비활성화
- 그룹의 라벨 : <legend> 요소
 - 그룹을 구분하기 위한 그룹의 제목 라벨
 - <fieldset> 요소에 포함되는 첫 번째 자식 요소로서 한번만 사용
 - 그룹 라벨의 스타일은 그룹을 구분하는 선의 중간에 걸쳐서 표시

도서 검색 예제

```
<form method="post" action="form_app.js">
  <fieldset>
    <legend>로그인</legend>
    <label for="user_id">아이디 : </label>
    <input type="text" name="id" size="20" id="user_id"> <br>
    <label for="user_pw">비밀번호 :</label>
    <input type="password" name="pw" size="20" id="user_pw">
  </fieldset> <br>
  <fieldset>
    <legend>통합 검색</legend>
    <label for="book_name">도서명 : </label>
    <input type="text" name="book_search" size="50" id="book_name">
    <br>검색범위 :
    <input type="radio" name="s_type" value="keyword" id="keyword">
    <label for="keyword">키워드</label> ...
    <br>자료유형 :
    <input type="checkbox" name="d_type" value="all">전체 ...
  </fieldset> <br> ...

  <button type="submit">검색</button>
  <button type="reset">지우기</button>
</form>
```

그림핑 예제

webclass.me/html5_3e/ch06/input2_group.html

도서 검색

로그인

아이디 :

비밀번호 :

통합 검색

도서명 :

검색범위 : ☐ 키워드 ☐ 본문내용

자료유형 : ☐ 전체 ☐ 단행본 ☐ 학술지 ☐ 비도서

검색 지우기

고급 형식으로 입력하기

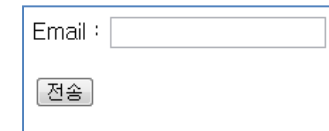
서식이 있는 텍스트 입력
날짜와 시간 입력
색상 및 숫자 입력
데이터 목록에서 선택

〈input〉 요소에 추가된 입력 형식

- text, password, radio, checkbox, button 외에 새로운 양식
 - 텍스트 관련: email, URL, tel, search (검색창)
 - 날짜와 시간 : date, month, week, time, datetime, datetime-local
 - 색상 및 숫자: number, range, color
- 〈input〉 요소에 새로운 속성
 - Autocomplete, placeholder, required, autofocus, step 속성
- 〈input〉 요소 이외에 추가된 입력 요소
 - 〈output〉 요소, 〈datalist〉 요소, 〈keygen〉 요소
- 유효성 검사
 - 예, 이메일 입력 필드, 필수적으로 입력해야 하는 필드
 - 검사 대상: 〈input〉, 〈select〉, 〈textarea〉, 〈button〉에서 submit 일 때
 - required 속성, novalidate 속성

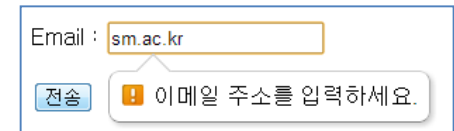
서식이 있는 텍스트 입력

- 이메일 주소 입력 : `<input type="email"/>`
 - 이메일 주소의 형식이 ****@***.***에 맞게 작성되었는지 확인
 - multiple 속성 : 여러 개의 이메일 주소, 콤마로 구분



Email :

전송

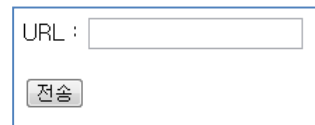


Email :

전송

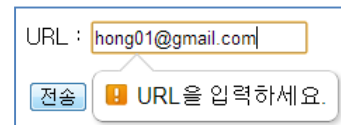
! 이메일 주소를 입력하세요.

- URL 주소 입력 : `<input type="url"/>`
 - 인터넷 주소 표기에 맞는 'http:/' 형식으로 입력되었는지 확인
 - value 속성에 "http:/"를 지정하여 미리 표시



URL :

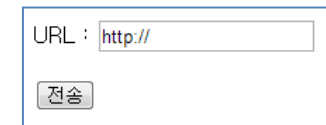
전송



URL :

전송

! URL을 입력하세요.



URL :

전송


- 전화번호 입력 : `<input type="tel" />`
 - pattern 속성 : 원하는 전화번호와 자리수에 유효한 패턴을 지정
 - 정규 표현식으로 정의, 자바스크립트에서 사용하는 패턴과 동일
 - `pattern="[A-Za-z]{no}"` , `pattern="[0-9]{no}"` , `pattern="[0-9]+"` , `pattern="[A-Za-z0-9]{min, max}"`
 - placeholder 속성 : 입력할 자리수 표시 (단순히 표시만)

Tel : `<input type="tel" placeholder="00*-000*-0000" pattern="[0-9]{2,3}-[0-9]{3,4}-[0-9]{4}" />`

Tel :

Tel :

Tel :

 요청한 형식과 일치시키세요.

- 검색창 입력 : `<input type="search" />`

Search :

Search :

날짜와 시간 입력

- 날짜 입력(일, 월, 주)

- `<input type="date" />` 연-월-일
- `<input type="month" />` 연-월
- `<input type="week" />` 연-주 (번호순서 1월 첫 주가 'W01')
 - min 속성, max 속성, value 속성
 - 속성값은 '연-월-일' 형식으로 날짜 입력

(a) type="date"

The screenshot shows a web browser window with the URL `webclass.me/html5_3e/ch06/input3_date.html`. It displays three input fields: `date (연-월-일)`, `month (연도-월)`, and `week (연도-주)`. The `date` field is active, showing a calendar for October 2021. The date 2021-10-13 is selected. The `month` field shows 2021년 10월, and the `week` field shows --, --번째 주.

(b) type="month"

The screenshot shows the same web browser window. The `month` field is active, showing a month picker for 2021. The month 10월 (October) is selected. The `date` field shows 2021-10-13, and the `week` field shows --, --번째 주.

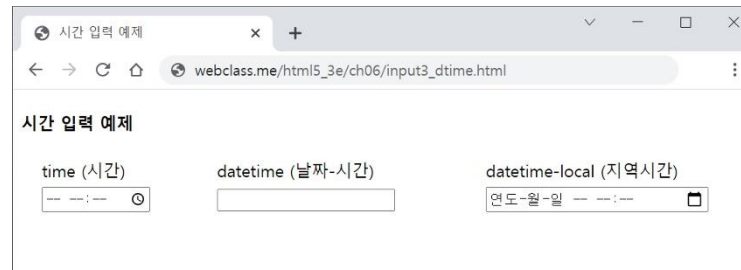
(c) type="week"

The screenshot shows the same web browser window. The `week` field is active, showing a week picker for 2021년 10월. The week 41 is selected. The `date` field shows 2021-10-13, and the `month` field shows 2021년 10월.

• 시간 입력

- `<input type="time" />` : “위/아래()” 버튼으로 시간 조정
- `<input type="datetime" />` UTC 국제표준 시간대
- `<input type="datetime-local" />` 현지 시간

time, datetime, datetime-local 형식의 예



The screenshot shows a web browser window with the title "시간 입력 예제" and the URL "webclass.me/html5_3e/ch06/input3_dtime.html". The page content is titled "시간 입력 예제" and displays three input fields:

| time (시간) | datetime (날짜-시간) | datetime-local (지역시간) |
|----------------------|--------------------------|--------------------------------|
| <input type="time"/> | <input type="datetime"/> | <input type="datetime-local"/> |

The first field, "time (시간)", shows a time selection interface with hour and minute spinners and a clock icon. The second field, "datetime (날짜-시간)", is an empty text input. The third field, "datetime-local (지역시간)", shows a date and time selection interface with a calendar icon.

날짜와 시간 입력 예제

```
<form method="get" action="form_app.js">  
  성명 : <input type="text" name="p_name"/> <br>  
  전화 : <input type="tel" name="p_tel" placeholder="00*-000*-0000"  
        pattern="[0-9]{2,3}-[0-9]{3,4}-[0-9]{4}" /> <br>  
  이메일 : <input type="email" name="p_mail" placeholder="***@***.***"/>  
  <p>  
  도서명 : <input type="text" size="25" name="book_title"/> <br>  
  예약 희망일 : <input type="date" name="last_date" min="2013-01-30"/> <br>  
  수령 시간 : <input type="time" name="time_from" min="09:00" max="18:00"/>  
  에서 <input type="time" name="time_until" min="09:00" max="18:00"/> 사이 <br>  
  <hr>  
  <input type="submit" value="예약하기"/>  
</form>
```

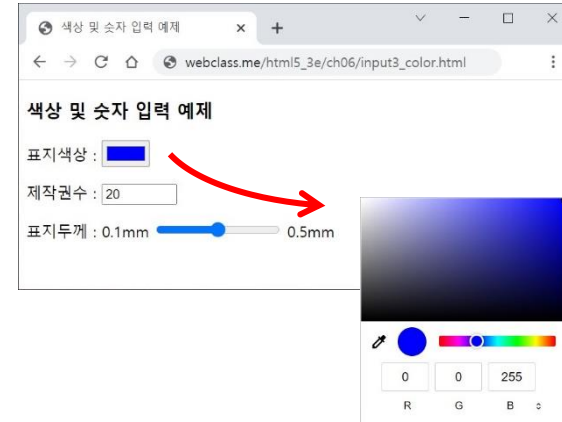
초기상태

입력오류

색상 및 숫자 입력

- 색상 입력 : `<input type="color"/>`

- 사용자가 직접 색상을 선택



- 숫자 입력 : `<input type="number"/>`

- min 속성, max 속성, value 속성, step 속성

예, `<input type="number" min="0" max="100" step="10" value="20"/>`

제작권수 :

- 범위 입력 : `<input type="range"/>`

- 스크롤바

- min 속성, max 속성, step 속성

예, 0.1mm `<input type="range" min="1" max="5" value="3" />` 0.5mm



데이터 목록에서 선택

- 데이터 목록 기능 : <datalist> 요소
 - <input> 요소의 list 속성에 <datalist> 요소의 id 속성을 지정
 - 검색어 자동완성 혹은 제시어 기능 구현 가능
 - 입력 창에 포커스가 들어오면 옵션 목록이 등장

```
도서명 : <input type="text" list="titles" name="book" />  
<datalist id="titles">  
  <option value="멀티미디어배움터" label="30,000"/>  
  <option value="인터넷배움터" label="34,000"/>  
  <option value="컴퓨터와 IT 기술의 이해" label="28,000"/>  
</datalist>
```

도서명 :

도서명 :

도서명 : 멀티미디어배움터

도서 구입 요청 예제

<form>

<h3>도서 구입 요청</h3>

도서명 : <input type="text" list="book" />

<datalist id="book">

<option value="멀티미디어배움터" label="30,000"/>

<option value="인터넷배움터" label="34,000"/>

<option value="컴퓨터와 IT 기술의 이해" label="28,000"/>

</datalist>

<p>선호도 : 1 <input type="range" min="1" max="5" value="3" /> 5

<p>가격 : <input type="number" name="price" min="0" step="100" value="10000"/> 원

<p>권수 : <input type="number" name="num" min="0" step="1" value="0" /> 권

<p><input type="submit" value="구입"/>

</form>

The image displays three sequential browser screenshots of a web form titled '도서 구입 요청' (Book Purchase Request). The browser address bar shows 'webclass.me/html5_3e/ch06/input3...'.
1. The first screenshot shows the form with the following elements: a text input field for '도서명' (Book Name), a range slider for '선호도' (Preference) set to 3, a number input for '가격' (Price) set to 10000, and another number input for '권수' (Quantity) set to 0. A '구입' (Purchase) button is at the bottom.
2. The second screenshot shows the '도서명' field replaced by a dropdown menu. The dropdown is open, showing three options: '멀티미디어배움터' (30,000), '인터넷배움터' (34,000), and '컴퓨터와 IT 기술의 이해' (28,000). The '선호도' slider is still at 3.
3. The third screenshot shows the form with the '도서명' field containing '인터넷배움터'. The '선호도' slider is now at 5. The '가격' field is set to 34000 and the '권수' field is set to 1. These three fields (book name, preference, price, and quantity) are highlighted with red dashed circles in all three screenshots.