

HTML5 웹 프로그래밍 입문(교수용)

4장. 다양한 입력 폼

목차

4.1 폼 이해하기

4.2 기본 형식으로 입력하기

4.3 고급 형식으로 입력하기

4.1 품 이해하기

<form> 요소의 사용

■ 폼 요소의 사용

- 회원가입, 상품구매, 키워드 검색 등 사용자로부터 정보를 받을 때
- 사용자와 애플리케이션이 상호작용
 - ▶ 사용자가 원하는 내용을 입력 => 전송버튼을 통해 데이터를 전송 => 애플리케이션으로 전달 => 작업처리 후에 실행결과를 반환

■ <form> 요소의 역할

- 사용자가 입력하는 정보를 하나로 묶어서 애플리케이션에 전달할 수 있도록 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="sex" value="male"/> 남성

<input type="radio" name="sex" 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>

<form> 요소의 주요 속성

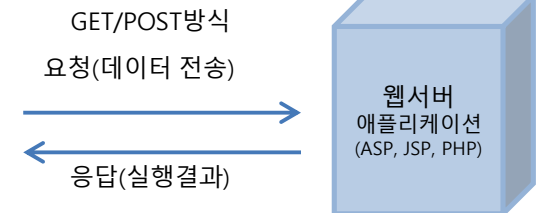
Difference with id? → see later example

<form **name="폼 이름"** method="get/post" action="애플리케이션 주소">

- method 속성 : 데이터들이 전송되는 방식을 지정하
 - ▶ 전송 방식의 종류는 **get방식과 post방식**
- action 속성 : 데이터를 처리할 애플리케이션 프로그램의 주소
 - ▶ 웹서버 프로그램인 경우에는 **URL 주소**
- name속성 : 폼 요소에 대한 이름을 지정

■ HTML 문서에서 <form> 의 역할

- 다양한 <input> 요소
 - => 입력된 데이터를
 - => 애플리케이션에 전달
 - => 그 실행 결과를 받는 것



get방식과 post방식

▶ Get방식

- 전송할 데이터를 URL 주소에 포함하여 문자열 형태로 전달
- URL 뒤에 ?에 이어서 "변수명=값", 변수가 여러 개일 경우는 &로 구분
- 한글은 16비트 유니코드로 표현

예,

...

ch04/application.js?person=%C8%AB%B1%E6%B5%BF&sex=male&job=

...

- get 방식은 간단한 데이터를 전달할 때에는 편리
- 그러나 주소 창에 전달하는 값이 노출되어 보안에 취약

▶ Post방식

- 프로그램의 입출력 방식을 사용하여 데이터의 양에 제한이 없다
- 전송하는 데이터가 드러나지 않으므로 보안이 필요한 경우에 많이 사용

4.2 기본 형식으로 입력하기

4.2.1 텍스트 입력

4.2.2 선택항목의 입력

4.2.3 버튼 입력

4.2.4 기타 입력 필드

4.2.5 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="foo"**, value="입력 값"/>

- name 속성은 입력 요소의 이름
- value 속성은 입력 요소의 값
- type 속성은 입력 폼의 유형을 결정

Refer later in the javascript
If foo == "blabla" ...
Actual handling of events (next lecture)

텍스트 관련	text (문자열 입력 필드), password (암호 문자열)
선택 관련	radio (단일 선택), checkbox (복수 선택)
버튼	submit (전송 버튼), reset (초기화 버튼), button (임의 기능 버튼), image (이미지를 전송 버튼으로 정의)
기타	file (파일 선택), hidden (숨김 필드)

텍스트 입력

■ 문자열 입력 필드

`<input type="text" name="변수명" value="초기값"/>`

- 한 줄의 문자열을 입력 받는 가장 기본적인 입력 형식
- name 속성 : 애플리케이션에 전달될 변수명을 지정
- value 속성 : 사용자로부터 입력 받은 값을 전달
 - ▶ 문자열 입력 창에 초기 문자를 보여주려면 value 속성에 값을 지정
default

■ 암호 입력 필드

`<input type="password" name="변수명"/>`

- 비밀번호나 주민등록번호 등 보안이 필요한 문자 입력
- 사용자 입력 데이터가 남들이 볼 수 없도록 ‘.’으로 표시
 - ▶ 화면에 표시되는 형식만 다를 뿐 더 이상의 보호 기능은 없다.

■ 텍스트 영역 필드 : <textarea> 요소

- 여러 줄에 걸치는 텍스트를 입력
- name 속성 : <textarea> 요소의 이름을 지정
- cols 속성 : 텍스트 영역의 한 줄에 해당하는 문자수, 즉, 열의 개수
- rows 속성 : 텍스트 영역의 행의 개수
 - ▶ 표시 영역보다 많은 텍스트를 입력하면 스크롤바
 - ▶ 텍스트 영역에 초기 문장은 <textarea>와 </textarea> 사이에 기입

<textarea name="이름" cols="열의 수" rows="행의 수">

텍스트 영역에 표시되는 초기 문장

</textarea>

아이디와 비밀번호, 요청사항 입력

<h3> 문자열, 암호 입력 및 텍스트 영역</h3>

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

`<input type="text" name = "id" value= "...ID 입력..." >`

비밀번호:

<p>요청사항:

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

</textarea>

</form>

문자열, 암호 입력 및 텍스트 영역

아이디 :

비밀번호:

요청사항:

전달하실 내용을 적으세요:

웹클래스 입력 예제

webclass.me/html5/ch04/ex402_text

문자열, 암호 입력 및 텍스트 영역

아이디 :

비밀번호 :

요청사항:

전달하실 내용을 적으세요:

손쉬운 경제상식이 필요합니다. 도서관에
경제 상식을 쉽게 이해할 수 있는 교양서적
이 부족합니다. 학생들이 취업할 때 전공
상식 이외에 경제에 대한 상식도 많이 필요

선택항목의 입력

■ 라디오 버튼

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

- 여러 항목 중에서 하나만 선택
- name속성과 value속성을 반드시 지정
 - ▶ 동일한 그룹의 라디오 버튼은 name속성을 동일하게
 - ▶ 즉, name 속성값이 같은 그룹에서 하나만 선택
 - ▶ 선택된 라디오 버튼의 value 속성값이 애플리케이션에 전달
- checked 속성 : 초기화면에서 미리 선택해 놓도록 설정
- 예,
 - ▶ 회원과 남자를 선택한 경우 “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"/>`

- 여러 항목에 대하여 개별적으로 선택, 여러개 선택 가능
- name 속성값은 모두 같은 값, value 값은 고유한 값
- checked 속성 : 초기 화면에 기본적으로 체크
 - ▶ 체크박스에 표시된 항목의 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
```

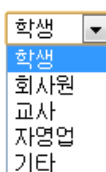
취미(중복선택) : ☐ 독서 ☒ 영화 ☒ 음악 ☒ 스포츠

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

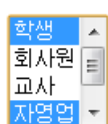
- 여러 개의 목록 중에서 하나를 선택
 - ▶ 드롭다운 형태 혹은 스크롤 박스 형태의 선택목록에서 항목을 선택
 - ▶ name 속성 : 선택목록의 변수명
 - ▶ size 속성 : 사용자들에게 보여줄 항목의 개수
 - size 속성이 1이면 드롭다운 목록, 2 이상이면 스크롤 박스의 크기
 - ▶ multiple 속성 : 하나 이상의 항목을 선택 가능하도록 설정
- 각 항목은 <select> 요소 내에서 <option> 요소로 정의
 - ▶ value 속성 : 각 항목을 구별하기 위한 값
 - ▶ selected 속성 : 초기 화면에서 특정 항목을 기본값으로 지정

```
<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: 

<select size="4" multiple>

직업: 

버튼 입력

■ 전송 버튼

`<input type="submit" value="버튼라벨"/>`

- 사용자가 입력한 값을 애플리케이션으로 전달
 - ▶ `<form>` 요소의 영역 안에 있는 모든 입력 데이터가 `<form>`의 `action` 속성에서 지정한 애플리케이션 프로그램으로 전송
- `value` 속성 :버튼에 표시하고 싶은 라벨 지정

■ 초기화 버튼

`<input type="reset" value="버튼라벨"/>`

- 사용자가 폼에 입력한 데이터 값들은 모두 초기화
- `value` 속성 : 버튼에 표시할 라벨 지정

■ 일반 버튼

`<input type="button" value="버튼라벨"/>`

- 다양한 용도로 사용할 수 있는 버튼
- value 속성값 : 버튼에 표시할 라벨 지정

■ 이미지 버튼

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

- 버튼 모양이 맘에 들지 않는 경우 원하는 이미지로 대체
- src 속성 : 이미지의 경로
- alt 속성 : 이미지에 문제가 있을 때 보여 줄 대체 텍스트

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

<p>취미(중복선택) :

☐독서☒영화☐음악☐스포츠

</p>

직업:

```
<select name="job" size="4" multiple>
```

<option value="student" selected>학생</option>

<option value="company">회사원</option>

<option value="teacher">교사</option>

<option value="sales">자영업</option>

`<option value="others">7|E|</option>`

</select>

```
<input type="button" value="확인하기" onClick="alert('입력값 확인')"/> &nbsp;
```

</form>

버튼 입력

책미(중복선택) : ☐ 독서 ☐ 영화 ☐ 음악 ☐ 스포츠

직업: 회사원 교사 자영업 기타

기타 입력 필드

■ 파일 선택하기 : `<input type="file"/>`

- 사용자가 폼 입력에서 파일을 선택
 - ▶ 초기 화면에는 파일 경로를 보여줄 문자열 필드와 “파일선택” 버튼
 - ▶ 버튼을 누르면 파일 선택창 등장, 문자열 필드에 선택된 파일 이름

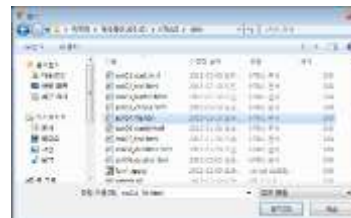
파일 업로드하기

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

크롬 브라우저

파일 업로드하기

파일 선택 선택된 파일 없음



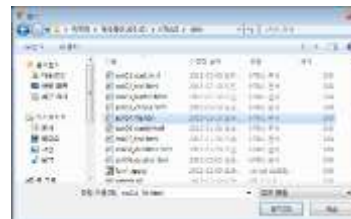
파일 업로드하기

파일 선택 ax404_file.html

파이어폭스 브라우저

파일 업로드하기

찾아보기...



파일 업로드하기

E:\HTML5\ch04\exam4... 찾아보기...

■ 데이터 숨김

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

- 사용자가 입력하거나 선택하는 데이터가 아니라 시스템에서 특정 데이터 항목을 입력 받아 처리하고 싶은 경우가
- 예, 사용자의 아이디 주소나 이미 넘겨받은 데이터를 다시 보여줄 필요가 없을 때
- 화면에 아무것도 보여주지 않지만, `name`과 `value` 속성값을 전달

■ 텍스트 라벨

<label for="입력아이디">

- 텍스트 라벨과 특정 입력 필드를 연결
 - ▶ 해당 글자를 보여주는 것 이외에 데이터 전달에는 영향이 없다
- for 속성과 이와 연결되는 입력 양식의 id 속성에 동일한 아이디 값

성별: <label for="male">남성</label>

<input type="radio" name="sex" id="male" value="male">

<label for="female">여성</label>

<input type="radio" name="sex" 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 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>
    <input type="radio" name="s_type" value="content" id="content">
    <label for="content">본문내용</label>
    <br>자료유형 :
    <input type="checkbox" name="d_type" value="all">전체
    <input type="checkbox" name="d_type" value="book">단행본
    <input type="checkbox" name="d_type" value="paper">학술지
    <input type="checkbox" name="d_type" value="non_book">비도서<br>
  </fieldset> <br> ...
```

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

도서 검색

로그인

아이디 :

비밀번호 :

통합 검색

도서명 :

검색범위 : ☒ 키워드 ☐ 본문내용

자료유형 : ☒ 전체 ☐ 단행본 ☐ 학술지 ☐ 비도서

4.3 고급 형식으로 입력하기

- 4.3.1 서식이 있는 텍스트 입력
- 4.3.2 날짜와 시간 입력
- 4.3.3 색상 및 숫자 입력
- 4.3.4 고급 입력 요소

<input> 요소에 추가된 입력 형식

■ text, password, radio, checkbox, button 외에 새로운 양식

| | |
|---------|--|
| 텍스트 관련 | email (이메일 주소), URL (URL 주소), tel (전화번호), search (검색창 입력) |
| 날짜와 시간 | date (날짜), month (연도/월), week (연도/주), time (시간), datetime (UTC time zone의 날짜/시간), datetime-local (Time zone이 없는 날짜/시간) |
| 색상 및 숫자 | number (숫자), range (일정 범위의 수), color (색상 선택) |

■ <input> 요소에 새로운 속성

- ▶ autocomplete 속성 : 자동으로 사용자의 입력을 완성
- ▶ placeholder 속성 : 입력 값에 대한 설명을 희미하게 표시
- ▶ required 속성 : 사용자가 필수적으로 입력을 하도록 검증
- ▶ autofocus 속성 : 문서가 로드될 때 입력 영역에 마우스 커서를 표시
- ▶ step 속성 : 입력 필드의 숫자나 범위를 조절하는 단계를 지정

추가 입력요소 및 유효성 검사

■ <input> 요소 이외에 추가된 입력 요소

- <output> 요소 : 폼의 처리 결과
- <datalist> 요소 : 입력 양식에 대한 내용을 옵션 리스트로 제공
- <keygen> 요소 : 암호화 키를 생성

■ 유효성 검사

- 입력값이 형식에 맞게 입력되었는지 검사하는 기능
 - ▶ 예, 이메일 입력 필드, 필수적으로 입력해야 하는 필드
- 유효성 검사 대상
 - ▶ <input> 요소, <select> 요소, <textarea> 요소
 - ▶ <button> 요소의 의 경우 submit 형식일 때
- required 속성 : 오류 발생시 메시지를 표시하고 폼 전송 중단
- novalidate 속성 : 유효성 검사 대상에서 제외

서식이 있는 텍스트 입력

■ 이메일 주소 입력 : `<input type="email"/>`

- 이메일을 입력 받기 위한 입력 창
 - ▶ 이메일 주소의 형식이 ****@***.***에 맞게 작성되었는지 확인
 - ▶ 이메일 주소가 존재하는지 까지는 미검사
- `multiple` 속성 : 여러 개의 이메일 주소를 입력, 콤마로 구분

Email :

전송

Email :

전송

! 이메일 주소를 입력하세요.

■ URL 주소 입력 : `<input type="url"/>`

- 웹 주소의 URL을 직접 입력하는 경우를 지원
 - ▶ 인터넷 주소 표기에 맞는 'http://' 형식으로 입력되었는지 확인
- `value` 속성에 "http://"를 지정하여 입력 창에 미리 표시 가능

URL :

전송

URL :

전송

! URL을 입력하세요.

URL :

전송

■ 전화번호 입력 : <input type="tel" />


- 전화번호를 입력할 수 있는 입력 창
- pattern 속성 : 원하는 전화번호와 자리수에 유효한 패턴을 지정
 - ▶ 정규 표현식으로 정의, 자바스크립트에서 사용하는 패턴과 동일
- placeholder 속성 : 입력할 자리수를 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.

[노트: pattern 속성의 예]

pattern="[A-Za-z]{no}" : no개 만큼의 영문자를 입력

pattern="[0-9]{no}" : no개 만큼의 숫자를 입력

pattern="[A-Za-z0-9]{min, max}" : 영문자와 숫자를 min~max 만큼의 글자 수 입력

pattern="[0-9]+" : 숫자를 1개 이상 입력

■ 검색창 입력 : <input type="search"/>

- 검색을 위한 문자열을 입력할 수 있는 입력 필드
- 외형적으로 사용자 인터페이스에 차이
- 예, 입력한 검색어 취소

Search :

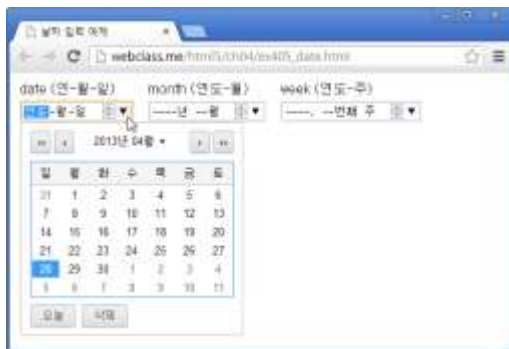
Search :

날짜와 시간 입력

■ 날짜 입력(일, 월, 주)

- `<input type="date"/>` 연-월-일의 날짜를 입력
- `<input type="month"/>` 연-월만 입력
- `<input type="week"/>` 연-주를 입력 (번호순서 1월 첫 주가 'W01')
- min 속성 : 최소값, max 속성 : 최대값, value 속성 : 초기값
 - ▶ 속성값은 '연-월-일' 형식으로 날짜 입력

(a) type="date"



The screenshot shows a web browser window with the URL `webclass.me/html/ch04/lec05_data.html`. The page contains three input fields: `date (연-월-일)`, `month (연도-월)`, and `week (연도-주)`. The `date` field is selected, and a calendar popup is displayed for the year 2013, showing the month of April. The calendar has a grid with days of the week and dates. The `date` field has a value of `2013-04-01`.

(b) type="month"



The screenshot shows the same web browser window. The `month` field is selected, and a month selection popup is displayed for the year 2013. The popup shows a list of months from January to December. The `month` field has a value of `2013-04`.

(c) type="week"



The screenshot shows the same web browser window. The `week` field is selected, and a week selection popup is displayed for the year 2013. The popup shows a list of weeks from W01 to W52. The `week` field has a value of `2013-W01`.

■ 시간 입력

- `<input type="time"/>`
 - ▶ 시간을 입력, “위/아래()” 버튼을 통해 시간과 분을 별도 조정
- `<input type="datetime"/>`
 - ▶ 날짜와 시간 선택, UTC(Universal Time Coordinated) 국제표준 시간대
- `<input type="datetime-local"/>`
 - ▶ 원하는 지역의 현지 시간 입력

time, datetime, datetime-local 형식의 예



The screenshot shows a web browser window with the address bar displaying `webclass.me/html5/ch04/ew405_dtime.html`. The page content is titled "시간 입력 예제" (Time Input Example). It contains three input fields:

| time (시간) | datetime (날짜-시간) | datetime-local (지역시간) |
|-----------|------------------|-----------------------|
| 오후 12:03 | | 2013-04-15 오후 12:03 |

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



도서 예약 대출

성명 :

전화 :

이메일 :

도서명 :

예약 희망일 :

수령 시간 : 에서 사이

입력오류



도서 예약 대출

성명 :

전화 :

이메일 :

도서명 :

예약 희망일 :

수령 시간 : 에서 오전 06:00 사이

값은 09:00 이상이어야 합니다.

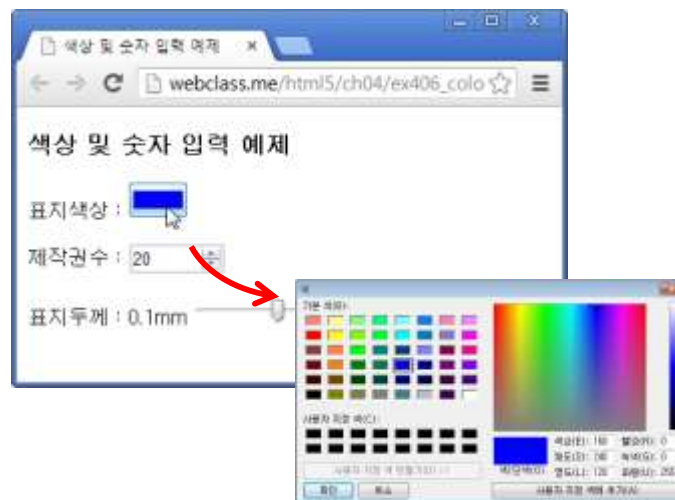
색상 및 숫자 입력

■ 색상 입력 : `<input type="color"/>`

- 사용자가 직접 색상을 선택하는 색상 선택메뉴 제공

▶ 크롬 브라우저

- 색상 입력 양식을 클릭하면 전체 색상을 보여주는 시스템 색상메뉴 등장



■ 숫자 입력 : <input type="number"/>

- 화살표 버튼으로 숫자를 조정할 수 있는 입력 형식
- 최소값 min 속성, 최대값 max 속성, 초기값 value 속성
- step 속성은 증가/감소 간격을 지정

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

제작권수 :

■ 범위 입력 : <input type="range"/>

- 스크롤바를 움직여서 일정한 범위의 숫자를 입력
- 최소값 min 속성, 최대값 max 속성
- 막대가 움직이는 칸의 간격 값은 step 속성

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

표지두께 : 0.1mm 0.5mm

고급 입력 요소

■ 데이터 목록 기능 : <datalist> 요소

- 검색어 자동완성 혹은 제시어 기능 구현 가능
 - ▶ <input> 요소의 list 속성과 <datalist> 요소의 id 속성을 동일하게 지정
- 데이터목록의 옵션 항목들이 제시어 목록으로 사용
 - ▶ 입력 창에 포커스가 들어오면 옵션 목록이 등장

```
참가국 : <input type="text" size="12" list="country" />  
<datalist id="country">  
  <option value="스페인"/>  
  <option value="영국"/>  
  <option value="독일"/>  
</datalist>
```

참가국 :



참가국 :
스페인
영국
독일



참가국 :

■ 자동 계산 출력 : <output> 요소

- 입력된 데이터로부터 계산된 결과 표현, 폼의 출력을 위한 요소
- name 속성 : 요소의 이름을 지정
- for 속성 : 변수의 이름을 지정
- 폼 전송 시 onForminput 이벤트를 이용하여 계산을 수행
 - ▶ 예, 가격 및 권수가 새로 입력되면 onForminput 이벤트 발생
=> <output> 요소의 value 값이 "price x num"으로 계산

금액 :

```
<input type="number" name="price" min="0" step="100" value="0" /> 원 X
```

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

```
<output onForminput="value=price.valueAsNumber*num.valueAsNumber+'원'"/>
```

금액 : 100 원 X 1 권 =

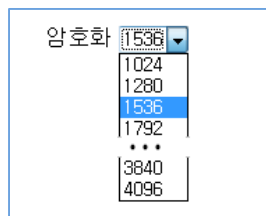


금액 : 100 원 X 1 권 = 125000원

■ 암호화 키 생성 : <keygen> 요소

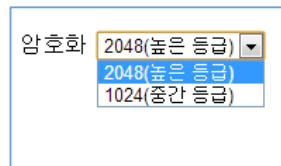
- 서버와 통신할 때의 보안을 위해 암호화 키를 생성
- 웹 보안을 위해 공개키 기반 암호화 방식에서 사용
 - ▶ 폼이 제출되면 공개키와 개인키가 쌍을 생성
=> 공개키는 서버로 전송, 개인키는 브라우저에서 사용자 키를 보관
- 브라우저 별로 지원되는 암호화 방식에는 차이

(a) 오페라 브라우저



A screenshot of the key generation dropdown menu in the Opera browser. The label '암호화' (Encryption) is to the left of the dropdown. The dropdown is open, showing a list of key sizes: 1024, 1280, 1536 (highlighted in blue), 1792, ..., 3840, and 4096.

(b) 크롬 브라우저



A screenshot of the key generation dropdown menu in the Chrome browser. The label '암호화' (Encryption) is to the left of the dropdown. The dropdown is open, showing a list of key sizes with their security levels: 2048(높은 등급) (highlighted in blue), 2048(높은 등급), and 1024(중간 등급).

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>합계 :
    <output onForminput="value=price.valueAsNumber*num.valueAsNumber+'원'"/>
  <p>암호화 <keygen name="key"/>
  <p><input type="submit" value="구입"/>
</form>
```

Opera

추가된 입력 요소 예제

도서 구입 요청

도서명 :

선호도 : 1 5

가격 : 원

권수 : 권

합계 :

암호화 1536

구입

Opera

추가된 입력 요소 예제

도서 구입 요청

도서명 :

선호도 : 5

가격 : 원

권수 : 권

합계 :

암호화 1536

구입

Opera

추가된 입력 요소 예제

도서 구입 요청

도서명 :

선호도 : 1 5

가격 : 원

권수 : 권

합계 : 170000원

암호화 1536

구입