

웹 서비스 플랫폼

Hybrid Web

제이쿼리 대화상자와 폼



1.1 대화상자

- 대화상자(dialog box)

- 제이쿼리 모바일이 지원하는 팝업 창 형태의 특별한 페이지
- 둥근 모서리와 바깥 여백, 그림자 등의 스타일이 적용되어 기존 페이지 위에 표시
- 확인 창이나 경고 창, 액션시트 스타일 등의 유형을 만들 때 사용

- 대화상자 생성 2가지 방법

1) <a> 링크 태그 안에 data-rel 속성값 'dialog'를 설정하는 방법

- 연결된 일반 페이지를 읽어 오면서 대화상자로 변환해서 표시

```
<a href="#" data-rel="dialog"> </a>
```

```
<!-- 링크 페이지의 유형을 대화상자로 선언 -->
```

2) 일반 페이지가 아닌 대화상자로 처음부터 생성하는 방법

- <div> 태그 안에 data-role 속성값을 'page'가 아닌 'dialog'로 설정

```
<div data-role="dialog" id=" ">
```

```
<!-- 페이지 유형을 대화상자로 선언 -->
```

- data-role="dialog" 속성을 설정하는 방법이 더 바람직

- 대화상자를 연결하는 버튼마다 data-rel="dialog" 속성값을 모두 설정할 필요없이 단 한 번만 페이지 유형을 대화상자로 설정하면 되기 때문



```
<!DOCTYPE html>
<html>
<head>
    <meta charset="utf-8"/>
    <meta name="viewport" content="width=device-width, initial-scale=1"/>
    <title>jQuery Mobile</title>
    <link rel="stylesheet" href="http://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css"/>
    <script src="http://code.jquery.com/jquery-1.11.1.min.js"></script>
    <script src="http://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.js"></script>
</head>
<body>

<div data-role="page" id="page1" data-position="fixed">
    <div data-role="header">
        <h1>header 영역</h1>
    </div>
    <div data-role="content">
        <p>기본형 페이지 </p>
        <a href="#dialog1" data-role="button" data-rel="dialog">대화상자형 페이지1</a>
        <a href="#dialog2" data-role="button">대화상자형 페이지2</a>
    </div>
    <div data-role="footer" data-position="fixed">
        <h4>footer 영역</h4>
    </div>
</div>
```

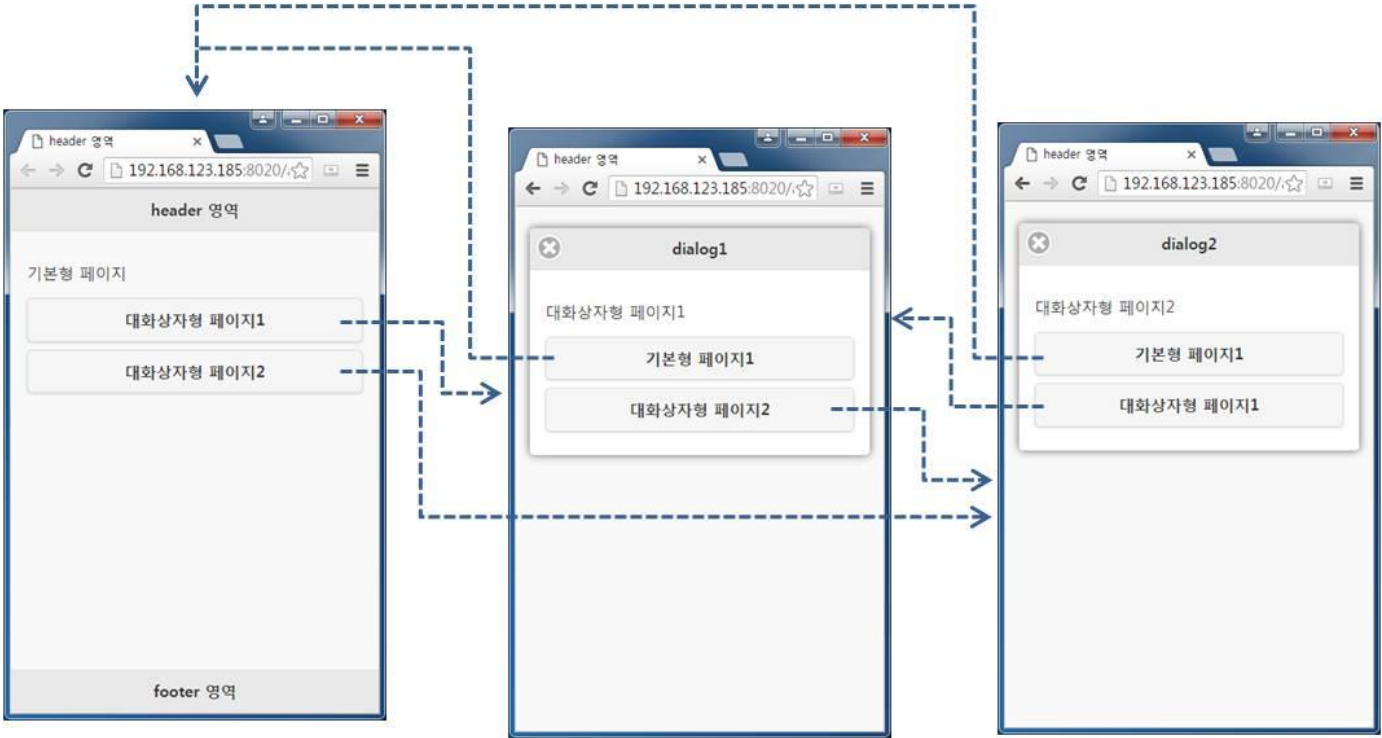


```
<div data-role="page" id="dialog1" data-position="fixed">
  <div data-role="header">
    <h1>dialog1</h1>
  </div>
  <div data-role="content">
    <p> 대화상자형 페이지1 </p>
    <a href="#page1" data-role="button">기본형 페이지1</a>
    <a href="#dialog2" data-role="button">대화상자형 페이지2</a>

  </div>
</div>
<div data-role="dialog" id="dialog2" data-position="fixed">
  <div data-role="header">
    <h1>dialog2</h1>
  </div>
  <div data-role="content">
    <p> 대화상자형 페이지2 </p>
    <a href="#page1" data-role="button">기본형 페이지1</a>
    <a href="#dialog1" data-role="button">대화상자형 페이지1</a>

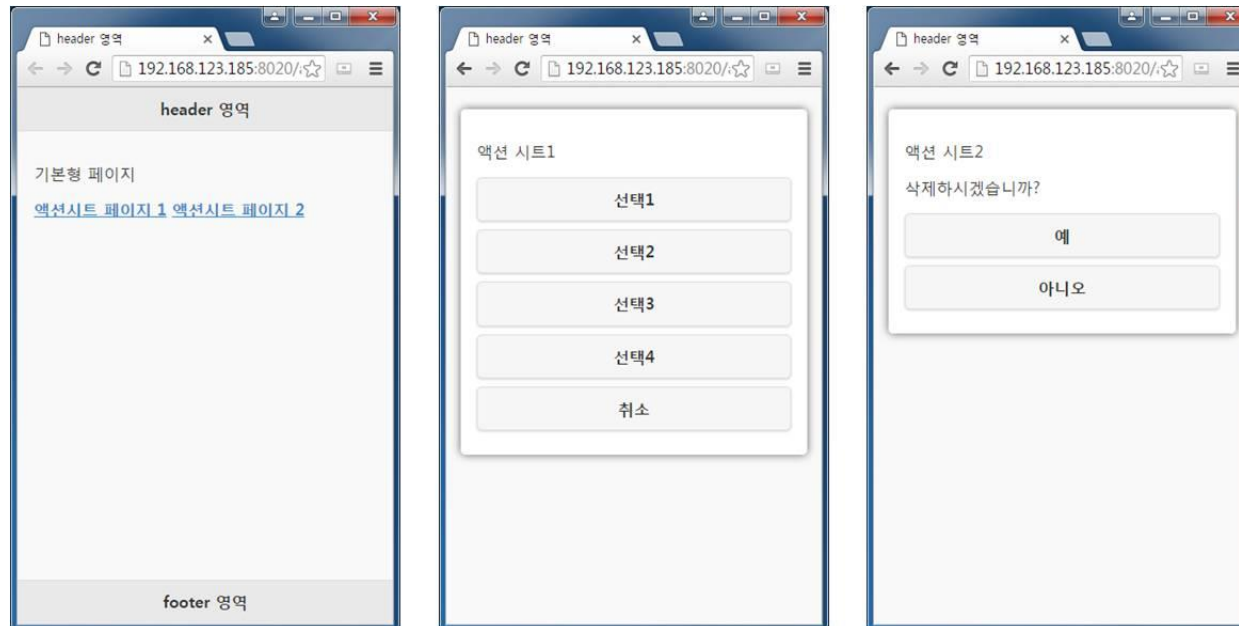
  </div>
</div>
</body>
</html>
```





액션시트 생성하기

- 액션시트(action sheets)
 - 헤더와 푸터가 없는 특별한 대화상자
 - 사용자에게 의견을 묻고 응답을 요청하는 경우에 사용.
 - 취소를 포함한 선택 가능한 작업 항목들을 제시할 목적으로 보통 둘 이상의 버튼을 포함
- action-sheet.html의 실행 결과(예제8-2)



```
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8"/>
  <meta name="viewport" content="width=device-width, initial-scale=1"/>
  <title>jQuery Mobile</title>
  <link rel="stylesheet" href="http://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css"/>
  <script src="http://code.jquery.com/jquery-1.11.1.min.js"></script>
  <script src="http://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.js"></script>
</head>
<body>

<div data-role="page" id="page1" >
  <div data-role="header" >
    <h1>header 영역</h1>
  </div>
  <div data-role="content" >
    <p> 기본형 페이지 </p>
    <a href="#dialog1" data-transition="slidedown">액션시트 페이지 1</a>
    <a href="#dialog2" data-transition="slidedown">액션시트 페이지 2</a>
  </div>
  <div data-role="footer" data-position="fixed">
    <h4>footer 영역</h4>
  </div>
</div>
```



```
<div data-role="dialog" id="dialog1">
  <div data-role="content">
    <p>액션 시트1</p>
    <a href="#" data-role="button">선택1</a>
    <a href="#" data-role="button">선택2</a>
    <a href="#" data-role="button">선택3</a>
    <a href="#" data-role="button">선택4</a>
    <a href="#" data-role="button" data-rel="back">취소</a>
  </div>
</div>
<div data-role="dialog" id="dialog2">
  <div data-role="content">
    <p>액션 시트2</p>
    <p>삭제하시겠습니까?</p>
    <a href="#" data-role="button" data-rel="back">예</a>
    <a href="#dialog1" data-role="button" data-rel="back">아니오
  </div>
</div>
</body>
</html>
```



1.2 페이지 테마

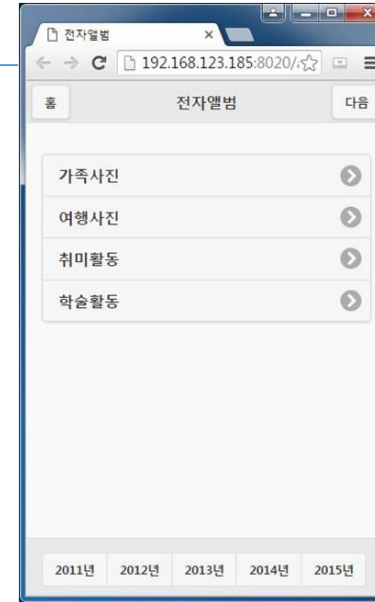
- 디자인 테마(theme)
 - 페이지와 페이지 구성 요소들에 적용되기 위해 미리 작성된 스타일시트 템플릿이다.
 - 색상, 배경, 글꼴, 테두리, 음영 등 스타일의 전체적인 설정 유형을 말하며 하나의 CSS3 파일을 구성
 - 페이지, 대화상자, 헤더, 푸터 등에 테마가 적용되면 색상과 음영, 질감 등이 자동 변화됨
 - 일부 요소만 테마를 설정 가능, 테마를 별도로 정의하지 않으면 부모 엘리먼트의 테마를 상속
- 대상 영역에 data-theme 속성값으로 'a'~'b'까지 테마 속성값 중 하나를 설정
 - 'a'는 은색 계열, 'b'는 검정계열
 - 기본 테마는 'a'
- 기본 테마(디자인 견본)

테마 견본	색상 계열	특징
a	은색	일반적인 기본 테마
b	검정	선명한 테마

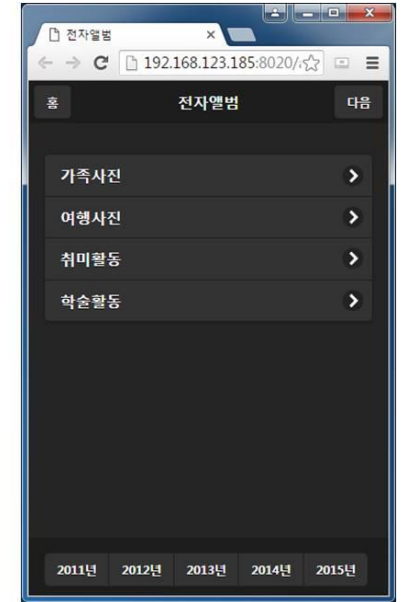


테마 적용하기

```
<div data-role="page" id="page1" data-theme="a">
  <div data-role="header" data-theme="b">
    <a href="#page1">홈</a>
    <h1>전자앨범</h1>
    <a href="#page2">다음</a>
  </div>
  <div data-role="content">
    <ul data-role="listview" data-inset="true">
      <li><a href="#">가족사진</a></li>
      <li><a href="#">여행사진</a></li>
      <li><a href="#">취미활동</a></li>
      <li><a href="#">학술활동</a></li>
    </ul>
  </div>
  <div data-role="footer" data-position="fixed" class="ui-bar" data-theme="b">
    <div data-role="controlgroup" data-type="horizontal" data-theme="a">
      <a href="#">2011년</a>
      <a href="#">2012년</a>
      <a href="#">2013년</a>
      <a href="#">2014년</a>
      <a href="#">2015년</a>
    </div>
  </div>
</div>
```



테마-a



테마-b

themeab.html



2.1 기본 레이아웃 형식

- 콘텐츠 레이아웃

- 스마트폰은 화면 크기가 작고 폭이 좁기 때문에 모바일 페이지의 레이아웃 구성이 중요

- 바 레이아웃(bar layout)

- 콘텐츠 영역 안에 제목처럼 한 줄 단위의 정보를 표시하기 위한 레이아웃
- class="ui-bar" 클래스 속성값에 'ui-bar-a'부터 'ui-bar-e'까지의 테마 속성값을 추가
- 툴바와 비슷한 양식 생성

- 블록 레이아웃(block layout)

- 콘텐츠 영역 안에 사각형 모양의 박스 영역을 생성
- 내용들을 규칙적으로 배치하거나 특정 내용을 강조하기 위해 많이 사용되는 레이아웃
- class="ui-body" 클래스 속성값에 'ui-body-a'부터 'ui-body-e'까지의 속성값을 추가
- 독립된 내용 공간 생성

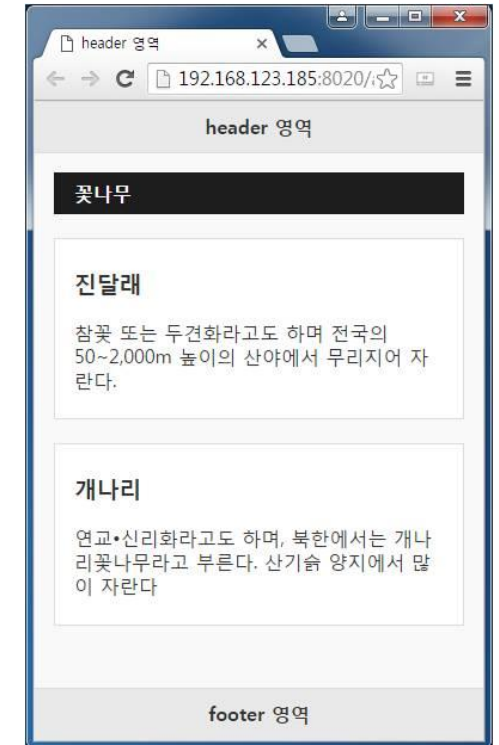
- 그리드(격자) 레이아웃(grid layout)

- 문자열이나 이미지를 행이나 열에 맞추어 일정한 간격으로 배치하고자 할 때 사용
- 화면을 균등하게 여러 행 또는 여러 열로 분할하는 다중 분할 레이아웃
- 버튼이나 네비게이션바, 아이콘 집합 등 작은 요소들을 나란히 나열할 경우에 적합



```
<div data-role="page" id="page1" >
  <div data-role="header" >
    <h1>header 영역</h1>
  </div>
  <div data-role="content" >
    <div class="ui-bar ui-bar-b">꽃나무</div> <br/>
    <div class="ui-body ui-body-a">
      <h3>진달래</h3>
      <p>참꽃 또는 두견화라고도 하며 전국의 50~2,000m
높이의 산야에서 무리지어 자란다.</p>
    </div> <br/>
    <div class="ui-body ui-body-a">
      <h3>개나리</h3>
      <p>연교•신리화라고도 하며, 북한에서는 개나리꽃나
무라고 부른다. 산기슭 양지에서 많이 자란다</p>
    </div> <br/>
  </div>
  <div data-role="footer" data-position="fixed">
    <h4>footer 영역</h4>
  </div>
</div>
```

bar-body.html



2.2 그리드 레이아웃 형식

- 그리드 컨테이너(grid container)
 - 그리드 레이아웃은 둘 이상의 블록들을 갖는 그리드 컨테이너를 만듦으로써 생성됨
 - 컨테이너 내부에 포함하는 열의 개수에 따라 적절한 ui-grid-* 클래스 속성값을 명세
 - 최대 5개의 열을 생성
- [표 8-2] 그리드 컨테이너 속성값

ui-grid-* 속성값	그리드 열 개수	포함되는 그리드 블록
ui-grid-a	2	ui-block-a, ui-block-b
ui-grid-b	3	ui-block-a, ui-block-b, ui-block-c
ui-grid-c	4	ui-block-a, ui-block-b, ui-block-c, ui-block-d
ui-grid-d	5	ui-block-a, ui-block-b, ui-block-c, ui-block-d, ui-block-e

- 예) <div> 태그에 class="ui-grid-a" 클래스 속성을 설정하면 2개의 열로 분할되는 그리드 컨테이너가 생성

```
<div class="ui-grid-a"> </div>
```



그리드 블록

- 그리드 블록(grid block)

- 그리드 안에 생성되는 열의 개수만큼 블록이 그리드 컨테이너 내부에 생성
- 해당 엘리먼트가 몇 번째 블록에 해당하는지를 지정하기 위해 'a'~'e' 값을 사용
- 각 블록들을 식별하기 위하여 ui-block-* 속성값을 지정

- [표 8-3] 그리드 블록 속성값

ui-block-* 속성값	그리드 안 블록의 위치
ui-block-a	왼쪽 첫 번째 블록
ui-block-b	왼쪽 두 번째 블록
ui-block-c	왼쪽 세 번째 블록
ui-block-d	왼쪽 네 번째 블록
ui-block-e	왼쪽 다섯 번째 블록

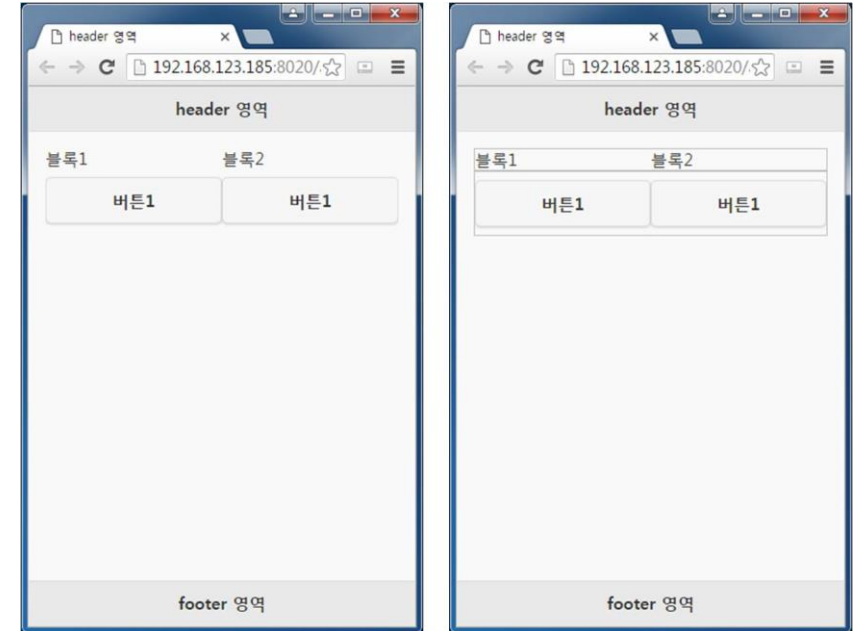
- 예) 그리드 컨테이너 안의 자식 엘리먼트에 class="ui-block-a", class="ui-block-b" 속성을 순서대로 설정하면 2개의 열에 블록이 생성

```
<div class="ui-grid-a">                                <!-- 그리드 컨테이너 -->
  <div class="ui-block-a"> </div>                        <!-- 그리드 블록 -->
  <div class="ui-block-b"> </div>
</div>
```



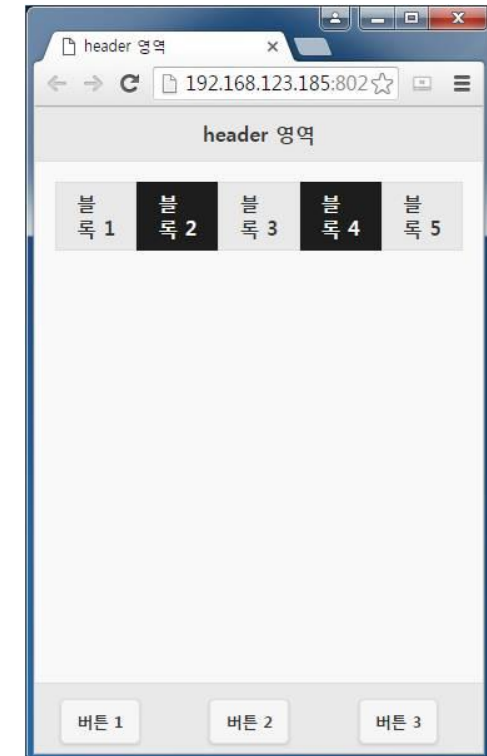
```
<div data-role="page" id="page1" >
  <div data-role="header" >
    <h1>header 영역</h1>
  </div>
  <div data-role="content" >
    <div class="ui-grid-a">
      <div class="ui-block-a">블록1</div>
      <div class="ui-block-b">블록2</div>
    </div>
    <div class="ui-grid-a">
      <div class="ui-block-a">
        <button>버튼1</button>
      </div>
      <div class="ui-block-b">
        <button>버튼1</button>
      </div>
    </div>
  </div>
  <div data-role="footer" data-position="fixed">
    <h4>footer 영역</h4>
  </div>
</div>
```

multicolumn-grid.html



```
<div data-role="page" id="page1" >
  <div data-role="header" >
    <h1>header 영역</h1>
  </div>
  <div data-role="content" >
    <div class="ui-grid-d">
      <div class="ui-block-a"><div class="ui-bar ui-bar-a">블록 1</div></div>
      <div class="ui-block-b"><div class="ui-bar ui-bar-b">블록 2</div></div>
      <div class="ui-block-c"><div class="ui-bar ui-bar-a">블록 3</div></div>
      <div class="ui-block-d"><div class="ui-bar ui-bar-b">블록 4</div></div>
      <div class="ui-block-e"><div class="ui-bar ui-bar-a">블록 5</div></div>
    </div>
  </div>
  <div data-role="footer" class="ui-bar ui-grid-b" data-position="fixed">
    <div class="ui-block-a"><a href="#" data-role="button">버튼 1</a></div>
    <div class="ui-block-b"><a href="#" data-role="button">버튼 2</a></div>
    <div class="ui-block-c"><a href="#" data-role="button">버튼 3</a></div>
  </div>
</div>
```

grid-theme.html





```
<div data-role="page" id="page1" >
  <div data-role="header" >
    <h1>header 영역</h1>
  </div>
  <div data-role="content" >
    <p>2행4열 그리드</p>
    <div class="ui-grid-c ">
      <div class="ui-block-a">1행1열</div>
      <div class="ui-block-b">1행2열</div>
      <div class="ui-block-c">1행3열</div>
      <div class="ui-block-d">1행4열</div>
      <div class="ui-block-a">2행1열</div>
      <div class="ui-block-b">2행2열</div>
      <div class="ui-block-c">2행3열</div>
      <div class="ui-block-d">2행4열</div>
    </div>
    <p>2행3열 그리드</p>
    <div class="ui-grid-b">
      <div class="ui-block-a"></div>
      <div class="ui-block-b"></div>
      <div class="ui-block-c"></div>
      <div class="ui-block-a"></div>
      <div class="ui-block-b"></div>
      <div class="ui-block-c"></div>
    </div>
  </div>
  <div data-role="footer" data-position="fixed">
    <h4>footer 영역</h4>
  </div>
</div>
```

3.1 기본 접이식 패널

- 접이식 패널 레이아웃

- 콘텐츠의 특정 영역을 접혔다 펼치기를 반복하며 내용을 숨기거나 표시
- 스마트폰의 좁은 화면 영역을 효과적으로 사용할 수 있는 방법
- 콘텐츠 영역에 접이식 패널을 여러 개 배치 가능

- 기본 접이식 패널(basic collapsible panel)

- 접고 펼칠 수 있는 패널 블록이 하나만 존재
- 접이식 패널 컨테이너 역할을 하는 <div> 태그에 data-role="collapsible" 속성을 설정

```
<div data-role="collapsible"> </div>    <!-- 접이식 패널 컨테이너 -->
```

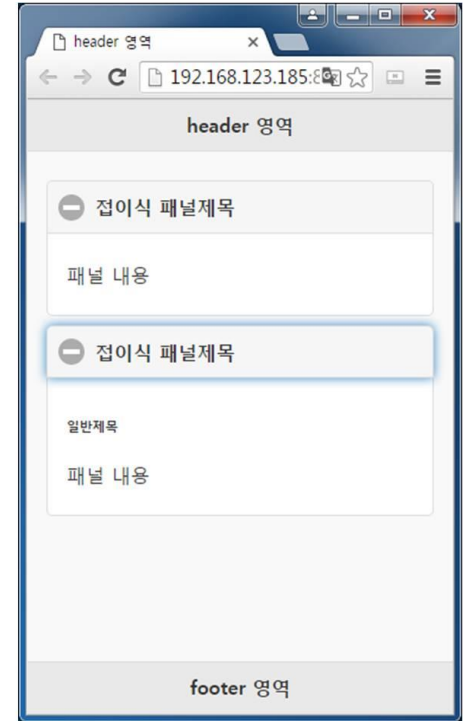
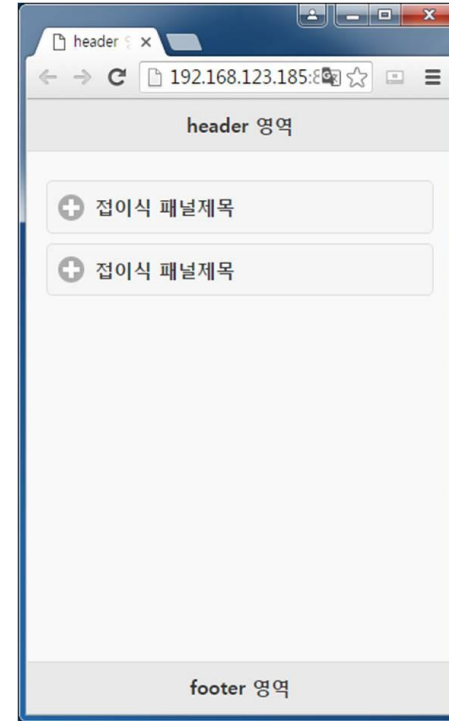
- 패널 제목은 <h1>~<h6> 태그로, 패널 내용은 <p> 태그를 이용하여 명세
- 첫 번째 <h> 태그만을 원래 위치가 아닌 패널 제목으로 표시
- 접이식 패널 안에 <h> 태그가 하나도 없다면 패널은 닫히지 않고 항상 펼쳐짐



기본 접이식 패널 생성하기

```
<div data-role="page" id="page1" data-position="fixed">
  <div data-role="header">
    <h1>header 영역</h1>
  </div>
  <div data-role="content">
    <div data-role="collapsible">
      <h3>접이식 패널제목</h3>
      <p>패널 내용</p>
    </div>
    <div data-role="collapsible">
      <h3>접이식 패널제목</h3>
      <h6>일반제목</h6>
      <p>패널 내용</p>
    </div>
  </div>
  <div data-role="footer" data-position="fixed">
    <h4>footer 영역</h4>
  </div>
</div>
```

basic-panel.html



3.2 중첩 접이식 패널

- 중첩 접이식 패널(nested collapsible panel)
 - 접이식 패널 안에 또다른 접이식 패널을 포함
 - data-role="collapsible" 속성을 설정한 <div> 태그 안에 또다시 data-role="collapsible" 속성을 설정한 <div> 태그를 반복해서 명세

```
<div data-role="collapsible">
    <div data-role="collapsible"> </div>
</div>
```

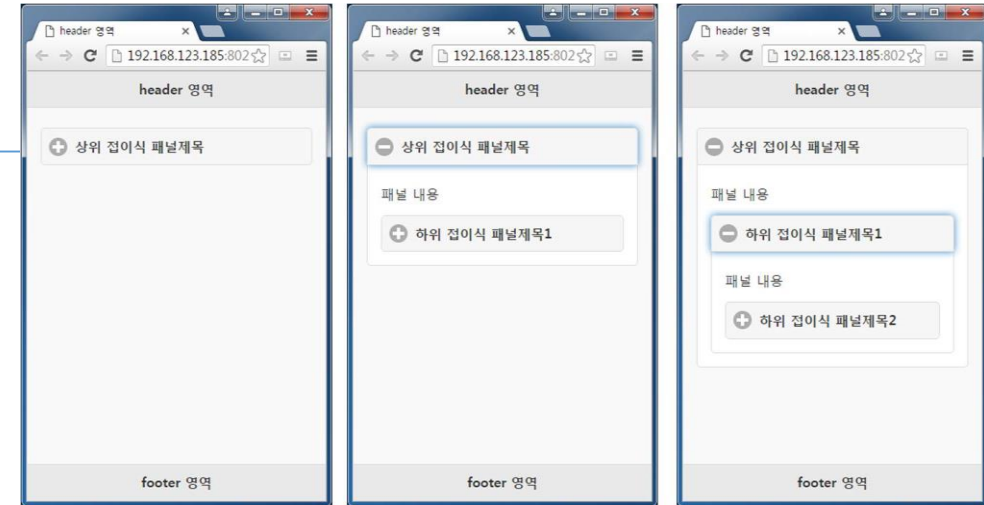
<!-- 상위 접이식 패널 컨테이너 -->
<!-- 하위 접이식 패널 컨테이너 -->

- 중첩 접이식 패널에서 바깥쪽 패널은 안쪽 패널의 컨테이너가 되어 계층적 패널 블록을 구성
- 중첩 단계 2~3 단계 정도가 적당
 - 제한은 없지만 화면 공간 크기와 내부 DOM 트리의 복잡성 등을 고려



nested-panel.html

```
<div data-role="page" id="page1" data-position="fixed">
  <div data-role="header" >
    <h1>header 영역</h1>
  </div>
  <div data-role="content" >
    <div data-role="collapsible">
      <h3>상위 접이식 패널제목</h3>
      <p>패널 내용</p>
      <div data-role="collapsible">
        <h3>하위 접이식 패널제목1</h3>
        <p>패널 내용</p>
        <div data-role="collapsible">
          <h3>하위 접이식 패널제목2</h3>
          <p>패널 내용</p>
        </div>
      </div>
    </div>
  </div>
  <div data-role="footer" data-position="fixed">
    <h4>footer 영역</h4>
  </div>
</div>
```



3.3 그룹 접이식 패널

- 그룹 접이식 패널(group collapsible panel)

- 여러 접이식 패널들을 하나의 그룹으로 묶은 것으로 하나의 연결된 요소로 동작
- 한 번에 하나의 접이식 패널만 선택하여 펼칠 수 있음
- data-role="collapsible-set" 속성을 갖는 <div> 태그를 설정

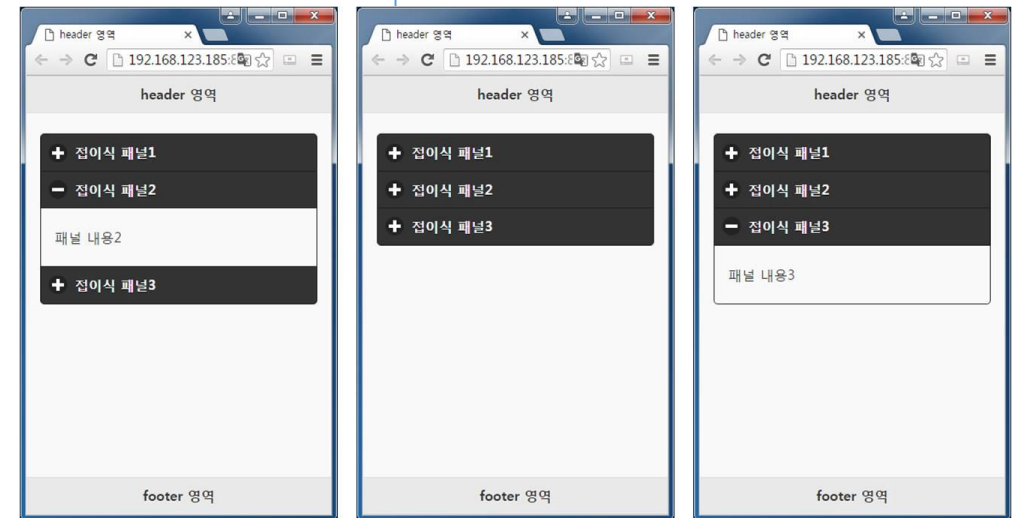
```
<div data-role="collapsible-set">                                <!-- 그룹 접이식 패널 컨테이너 -->
  <div data-role="collapsible"> </div>
  <div data-role="collapsible"> </div>
</div>
```

- '아코디언 패널'
 - 새로운 접이식 패널을 펼치게 되면 이전에 펼쳐졌던 패널은 자동으로 접힘
 - 마치 아코디언(accordion) 동작과 유사
- 특정 패널에 data-collapsed="false" 속성을 설정하면 처음부터 펼쳐진 상태로 표시



group-panel.html

```
<div data-role="page" id="page1">
  <div data-role="header">
    <h1>header 영역</h1>
  </div>
  <div data-role="content">
    <div data-role="collapsible-set" data-theme="b" data-content-theme="e">
      <div data-role="collapsible">
        <h3>접이식 패널1</h3>
        <p>패널 내용1</p>
      </div>
      <div data-role="collapsible" data-collapsed="false">
        <h3>접이식 패널2</h3>
        <p>패널 내용2</p>
      </div>
      <div data-role="collapsible">
        <h3>접이식 패널3</h3>
        <p>패널 내용3</p>
      </div>
    </div>
  </div>
  <div data-role="footer" data-position="fixed">
    <h4>footer 영역</h4>
  </div>
</div>
```



4.1 HTML5 표준 폼 태그(1)

● 폼(form)

- 사용자와의 상호 작용을 위해 필수적인 요소
- 표준 HTML5 태그 형식의 네이티브 웹폼 컨트롤을 자동으로 모바일폼 컨트롤로 렌더링
- <form> 태그
 - 입력 컨트롤과 관련 정보를 묶어 하나의 입력폼으로 그룹화하고 전송 방식을 설정
 - 입력폼 영역의 시작과 끝을 지정하지만 웹 브라우저에 표시되지는 않음
 - 하위에 <input>, <select>, <textarea> 등 입력 컨트롤을 생성하여 폼을 완성

● [표 8-4] 폼 태그

속성	속성값	기능
action	URL 주소	사용자 입력 정보를 전송할 위치(처리 프로그램의 주소)를 지정
method	GET POST	사용자 입력 정보의 전송 방식을 지정 GET방식 : URL 주소에 입력 정보(변수명+변수값)를 추가해서 문자열로 보내는 방식 POST방식 : URL 주소가 아닌 내부적(트랜잭션)으로 보내는 방식 보안성이 높고 전송량에 제한이 없음
name	문자열	폼의 이름



HTML5 표준 폼 태그(2)

- 입력(<input>) 태그
 - type 속성값에 따라 다양한 유형의 입력 컨트롤을 생성
 - 폼 안에서 사용자로부터 정보를 입력 받는 가장 기본적인 방식
- [표 8-5] 기본 입력 유형

속성	속성값	기능
type	text password hidden checkbox radio button submit reset image file	문자열 입력양식 비밀번호 입력양식(●로 표시) 글자 숨김 입력양식 체크박스 버튼(복수 선택) 라디오 버튼(단일 선택) 일반 버튼 전송 버튼(데이터를 서버에 전송) 초기화 버튼(입력 데이터를 모두 지움) 이미지 전송 버튼 파일 업로드 입력양식
name	문자열	입력폼 안에서 명명하기 위한 입력양식 이름 데이터 전송 시 변수명으로 사용
id	문자열	입력폼 안에서 식별하기 위한 고유한 입력양식 식별자
value	문자열 수치	컨트롤의 표시 제목이나 내부 기본값(기본 문자열)을 지정 데이터 전송시 전달될 변수값을 지정
max	수치	입력 최대값
min	수치	입력 최소값
step	수치	입력 증분값



HTML5 표준 폼 태그(3)

● [표 8-6] 고급 입력 유형

속성	속성값	기능
type	search	키워드 검색 입력양식
	tel	전화번호 입력양식
	email	이메일 입력양식
	url	URL 웹주소 입력양식
	number	숫자 입력양식(스핀버튼 ▲ ▼ 선택)
	range	숫자범위 입력양식(슬라이드 막대 이동)
	month	월 입력양식
	week	주 입력양식
	date	날짜 입력양식
	time	시간 입력양식
	datetime	날짜시간 입력양식
	datetime-local	지역날짜시간 입력양식
	color	색상 입력양식



HTML5 표준 폼 태그(4)

- 라벨(<label>) 태그
 - 입력 컨트롤에 대해 의미나 설명을 표시
 - 라벨 문자열과 입력 컨트롤을 서로 연결
 - <input> 태그의 id 속성값을 <label> 태그의 for 속성값으로 설정
 - <label> 태그의 영역을 선택하면 연결된 입력 컨트롤이 자동으로 선택된다.
- 복수문자열 입력(<textarea>) 태그
 - 여러 줄에 걸쳐 긴 문자열을 입력하기 위해 사용
- 선택(<select>) 태그
 - 선택 입력 컨트롤을 생성하기 위해 사용
 - 사용자는 나열된 목록 항목 중 하나를 선택함으로써 입력 정보를 결정한다. 하위 태그로 <optgroup>이나 <option> 태그를 사용할 수 있다. 여러 항목을 동시에 선택하고 싶다면 multiple='multiple'을 설정하면 가능하다.



HTML5 표준 폼 태그(5)

- 옵션 그룹화(<optgroup>) 및 옵션(<option>) 태그
 - <option> 태그
 - <select> 태그나 <optgroup> 태그의 하위 태그로 사용
 - 선택할 수 있는 항목들을 표시한다.
 - <optgroup> 태그
 - <select> 태그의 하위 태그
 - <option> 태그의 선택 항목들을 하나로 묶어 그룹화
 - label 속성은 선택 옵션 그룹에 대한 이름이며 반드시 설정해야 하는 필수 속성
- 입력양식 그룹화(<fieldset>) 및 그룹 설명(<legend>) 태그
 - <fieldset> 태그
 - 입력폼 안의 서로 연관된 입력 컨트롤들을 하나로 묶어 그룹화
 - 폼 안의 입력 컨트롤들을 화면 공간에서 분류함으로써 이해하기 쉽도록 함
 - <legend> 태그
 - <fieldset> 태그로 묶은 입력 양식 그룹에 대한 설명을 표시
 - <legend> 태그는 <fieldset> 태그 하위에 사용



4.2 네이티브 폼 컨트롤

- 네이티브 폼컨트롤 생성하기
 - 네이티브 스타일(native style)
 - 제이쿼리 모바일을 적용하기 이전의 HTML5 태그의 원래 표준 양식
 - 제이쿼리 모바일이 폼 컨트롤을 자동 변환시키는 것을 원하지 않는 경우
 - 폼 태그 안에 data-role="none" 속성을 추가
 - [그림 8-12] native-form.html의 실행 결과(예제8-11)

제이쿼리 모바일 프레임워크 적용 전

제이쿼리 모바일 프레임워크 적용 후



```
<div data-role="page" id="page1" >
  <div data-role="header" >
    <h1>header 영역</h1>
  </div>
  <div data-role="content" >
    <form method="get" action="form.jsp">
      <h1>동아리 회원 가입</h1>
      <fieldset>
        <legend>필수 정보</legend>
        <table>
          <tr>
            <td><label for="s_id">학번</label></td>
            <td><input type="text" name="s_id" id="s_id" disabled data-role="none"/></td>
          </tr>
          <tr>
            <td><label for="s_name">성명</label></td>
            <td><input name="s_name" id="s_name" disabled="disabled" data-role="none"/></td>
          </tr>
        </table>
      </fieldset> <br/>
    </form>
  </div>
</div>
```



```
<fieldset>
  <legend>부가 정보</legend>
  전화번호 : <input type="text" name="s_tel" placeholder="999-999-9999" data-role="none"/> <br/>
  가입날짜 : <input type="date" name="s_date" value="2014-11-11" data-role="none"/> <br/>
  성별 : <select data-role="none">
    <option value="male">남자</option>
    <option value="female">여자</option>
  </select> <br/>
  학년 : <input type="radio" name="syear" value="1" data-role="none">1학년
        <input type="radio" name="syear" value="2" data-role="none">2학년
        <input type="radio" name="syear" value="3" checked data-role="none">3학년
        <input type="radio" name="syear" value="4" data-role="none">4학년 <br/>
  나이 : <input type="number" name="s_year" min="1" max="100" value="20" data-role="none"/> <br/>
</fieldset> <br/>
<input type="submit" value="확인" data-role="none"/>
<input type="reset" value="취소" data-role="none"/>
</form>
</div>
<div data-role="footer" data-position="fixed">
  <h4>footer 영역</h4>
</div>
</div>
```



4.3 모바일 폼 컨트롤 : 입력 상자

- 입력 상자(input box)
 - 문자나 숫자값을 입력 받는 가장 기본적인 폼 컨트롤
 - type 속성값에 따라 다양한 폼 컨트롤 유형이 결정
- 모바일폼 요소 그룹화
 - 폼 컨트롤은 사용자에게 입력 정보 안내를 위해 <label> 태그를 함께 쌍으로 묶어 사용
 - 둘 이상의 모바일 폼 컨트롤을 하나의 그룹처럼 표시
 - 필드 컨테이너(field container)
 - data-role="fieldcontain" 속성을 갖는 <div> 태그로 감싸면 된다.
 - 레이블과 폼 컨트롤들을 정렬시키고 다른 컨트롤과 경계선으로 구분 짓는 래퍼

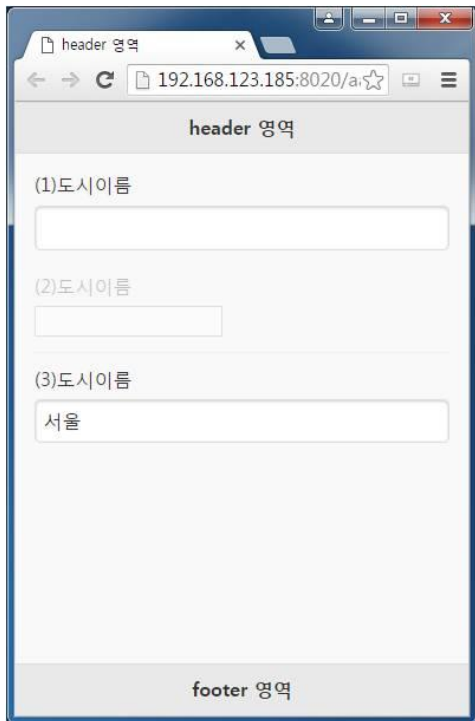
```
<div data-role="fieldcontain">
  <label for="아이디"> 라벨명 </label>
  <input type="입력 유형" name="변수명" id="아이디">
</div>
```

- 단일행 입력 상자
 - 가장 간단한 폼 입력 유형
 - type 속성값으로 'text'를 설정



단일행 입력상자 생성하기

basic-inputbox.html



```
<div data-role="page" id="page1">
  <div data-role="header">
    <h1>header 영역</h1>
  </div>
  <div data-role="content">
    <label>(1)도시이름
      <input type="text" name="return" id="txt1">
    </label>
    <div data-role="fieldcontain" class="ui-disabled">
      <label for="txt2">(2)도시이름</label>
      <input type="text" name="return" id="txt2" data-role="none">
    </div>
    <div data-role="fieldcontain">
      <label for="txt3">(3)도시이름</label>
      <input type="text" name="return" id="txt3" value="서울">
    </div>
  </div>
  <div data-role="footer" data-position="fixed">
    <h4>footer 영역</h4>
  </div>
</div>
```



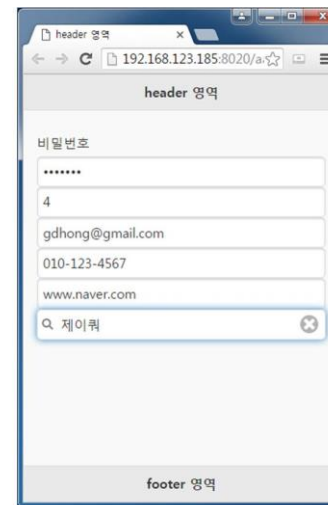
다양한 입력상자 생성하기

multi-inputbox.html

```
<div data-role="page" id="page1" >
  <div data-role="header" >
    <h1>header 영역</h1>
  </div>
  <div data-role="content" >
    <div data-role="fieldcontain">
      <label for="input1">비밀번호</label>
      <input type="password" name="input1" id="input1" placeholder="패스워드를 입력하세요"/>
      <input type="number" name="input2" id="input2" placeholder="숫자를 입력하세요"/>
      <input type="email" name="input3" id="input3" placeholder="이메일을 입력하세요"/>
      <input type="tel" name="input4" id="input4" placeholder="전화번호를 입력하세요"/>
      <input type="url" name="input5" id="input5" placeholder="URL을 입력하세요"/>
      <label for="input6" class="ui-hidden-accessible">검색</label>
      <input type="search" name="input6" id="input6" placeholder="키워드를 입력하세요">
    </div>
  </div>
  <div data-role="footer" data-position="fixed">
    <h4>footer 영역</h4>
  </div>
</div>
```



입력 전



입력 후



다중 행 입력상자 생성하기

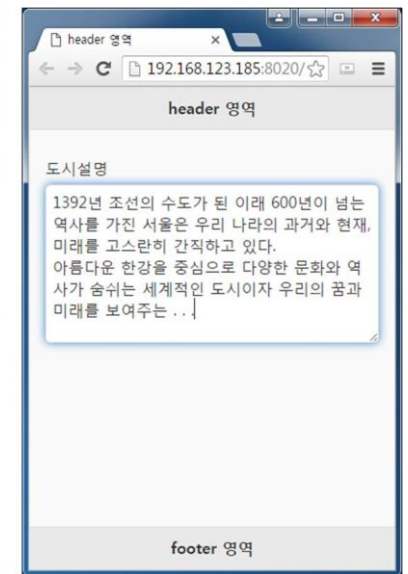
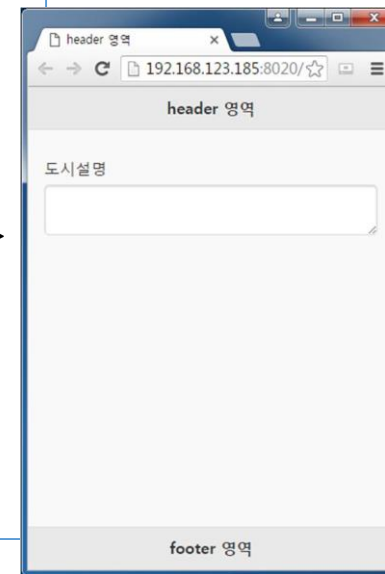
- 다중 행 입력 상자

- 한 줄이 아닌 여러 줄에 걸쳐 문자열을 입력하고자 할 경우, <textarea> 태그 사용

```
<textarea cols="열개수" rows="행개수" id="아이디"> </textarea>
```

textarea.html

```
<div data-role="page" id="page1" >
  <div data-role="header" >
    <h1>header 영역</h1>
  </div>
  <div data-role="content" >
    <div data-role="fieldcontain">
      <label for="txtarea1">도시설명</label>
      <textarea cols="25" rows="5" id="txtarea1"> </textarea>
    </div>
  </div>
  <div data-role="footer" data-position="fixed">
    <h4>footer 영역</h4>
  </div>
</div>
```



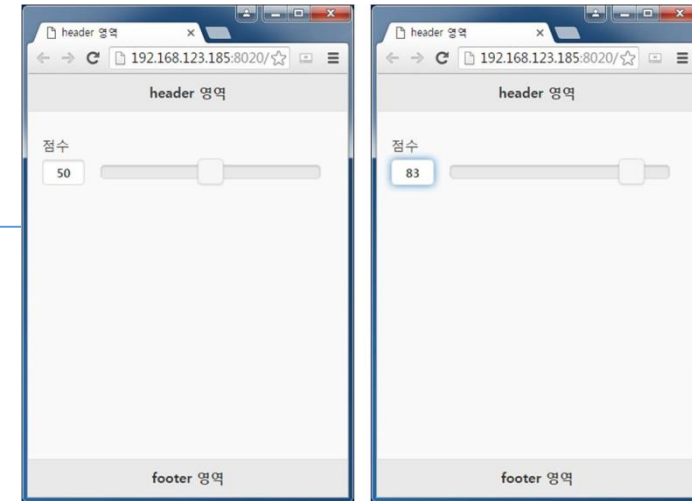
슬라이더 생성하기

● 슬라이더(slider)

- 이동 바를 좌우로 움직여 최소값과 최대값 사이의 값을 선택하는 모바일폼 컨트롤
- <input> 태그에 type="range" 속성값을 지정
- value 속성은 슬라이더의 초기값, min 속성은 최소값, max 속성은 최대값을 의미

range-input.html

```
<div data-role="page" id="page1" >
  <div data-role="header" >
    <h1>header 영역</h1>
  </div>
  <div data-role="content" >
    <div data-role="fieldcontain">
      <label for="score1">점수</label>
      <input type="range" name="score" id="score1" value="50" min="0" max="100"/>
    </div>
  </div>
  <div data-role="footer" data-position="fixed">
    <h4>footer 영역</h4>
  </div>
</div>
```



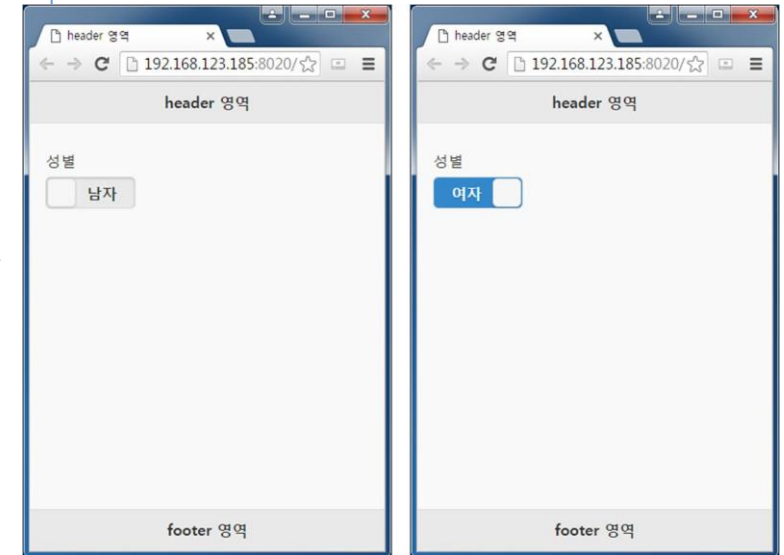
스위치 생성하기

● 스위치(switch)

- 서로 대비되는 두 가지 선택 항목 중에서 하나만을 선택할 수 있는 모바일 폼 컨트롤
- 선택 메뉴와 비슷, <select> 태그 안에 data-role="slider" 속성을 설정

switch-input.html

```
<div data-role="page" id="page1" >
  <div data-role="header" >
    <h1>header 영역</h1>
  </div>
  <div data-role="content" >
    <div data-role="fieldcontain">
      <label for="gender1">성별</label>
      <select name="gender" id="gender1" data-role="slider">
        <option value="male">남자</option>
        <option value="female">여자</option>
      </select>
    </div>
  </div>
  <div data-role="footer" data-position="fixed">
    <h4>footer 영역</h4>
  </div>
</div>
```



라디오 선택 버튼

- 라디오 선택 버튼

- 선택 가능 항목들 중에서 오직 하나만을 선택해야 할 때 사용하는 모바일폼 컨트롤
- <input> 태그 안에 type="radio" 속성을 설정
- name과 id 속성은 반드시 설정
- type="radio" 속성을 갖는 모든 <input> 태그들은 name 속성값이 같아야 함

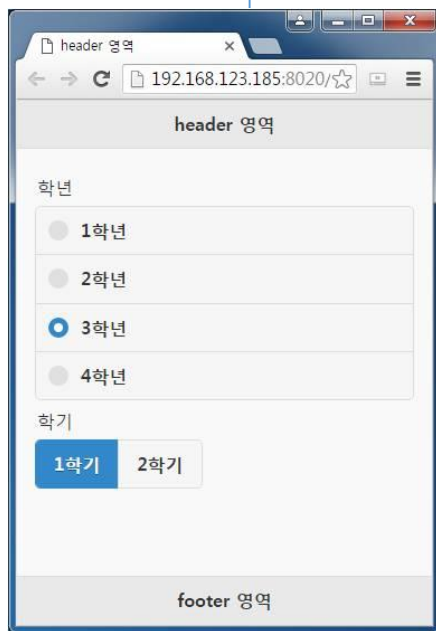
```
<input type="radio" name="변수명" id="아이디" value="변수설정값"/>
```

- <input> 태그들의 개수만큼 라디오 버튼이 생성
- 둘 이상의 버튼들을 그룹으로 묶기 위해 <fieldset> 태그로 감싸고 그 안에 data-role="controlgroup" 속성을 추가

```
<fieldset data-role="controlgroup">  
    <legend>그룹제목</legend>  
    ...  
</fieldset>
```

- <label> 태그는 각 모바일폼 컨트롤에 관한 설명을 위해, <legend> 태그는 그룹 전체에 관한 설명을 위해 사용





```

<div data-role="page" id="page1" >
  <div data-role="header" >
    <h1>header 영역</h1>

  </div>
  <div data-role="content" >
    <fieldset data-role="controlgroup">
      <legend>학년</legend>
      <input type="radio" name="syear" id="syear1" value="1" />
      <label for="syear1">1학년</label>
      <input type="radio" name="syear" id="syear2" value="2"/>
      <label for="syear2">2학년</label>
      <input type="radio" name="syear" id="syear3" value="3" checked="checked"/>
      <label for="syear3">3학년</label>
      <input type="radio" name="syear" id="syear4" value="4"/>
      <label for="syear4">4학년</label>
    </fieldset>
    <fieldset data-role="controlgroup" data-type="horizontal" >
      <legend>학기</legend>
      <input type="radio" name="semester" id="semester1" value="1" checked="checked"/>

      <label for="semester1">1학기</label>
      <input type="radio" name="semester" id="semester2" value="2"/>
      <label for="semester2">2학기</label>
    </fieldset>
  </div>
  <div data-role="footer" data-position="fixed">
    <h4>footer 영역</h4>
  </div>
</div>

```



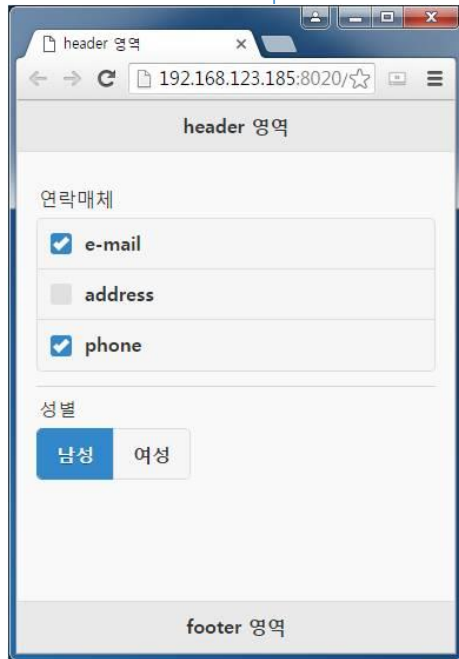
체크 상자 버튼 생성하기

- 체크 상자 버튼

- 대상 목록 중에서 여러 개를 동시에 선택하도록 할 때 사용
- 라디오 선택 버튼과 생성 방법이 비슷, <input> 태그에 type="checkbox" 속성을 설정

```
<input type="checkbox" name="변수명" id="아이디" value="변수설정값" />
```





```

<div data-role="page" id="page1">
  <div data-role="header" >
    <h1>header 영역</h1>
  </div>
  <div data-role="content">
    <div data-role="fieldcontain">
      <fieldset data-role="controlgroup">
        <legend>연락매체</legend>
        <input type="checkbox" name="type1" id="type1" value="1" checked="checked"/>
        <label for="type1">e-mail</label>
        <input type="checkbox" name="type2" id="type2" value="2" />
        <label for="type2">address</label>
        <input type="checkbox" name="type3" id="type3" value="3" checked="checked"/>
        <label for="type3">phone</label>
      </fieldset>
    </div>
    <fieldset data-role="controlgroup" data-type="horizontal">
      <legend>성별</legend>
      <input type="checkbox" name="gtype1" id="gtype1" value="m" checked="checked"/>
      <label for="gtype1">남성</label>
      <input type="checkbox" name="gtype 2" id="gtype2" value="f" />
      <label for="gtype2">여성</label>
    </fieldset>
  </div>
  <div data-role="footer" data-position="fixed">
    <h4>footer 영역</h4>
  </div>
</div>

```



4.4 모바일 폼 컨트롤 : 선택 메뉴(1)

- 기본 선택 메뉴(select menu)

- 여러 항목 중에서 하나를 선택하기 위한 폼 컨트롤이다.
- 표시 공간이 작아 효율적

```
<select name="변수명" id="아이디">  
    <option>안내글</option>  
    <option value="변수설정값">옵션</option>  
    ... 생략 ...  
</select>
```

- 선택 대상 제외(비활성화)

- <option> 태그의 value 속성이 없는 항목 -> 안내글 역할
- <option> 태그의 data-placeholder="true" 속성값이 추가된 항목
- <option> 태그 안에 disabled='disabled' 속성값을 추가한 항목



선택 메뉴(2)

- 그룹 선택 메뉴
 - 선택 메뉴의 항목들을 그룹핑하는 효과
 - <optgroup> 태그 안에 label 속성값을 설정
 - 각 그룹의 이름을 선택 항목들 사이에 표시, 구분하기 위해 사용

```
<optgroup label="그룹명">  
  <option value="변수설정값">옵션</option>  
  ... 생략 ...  
</optgroup>
```

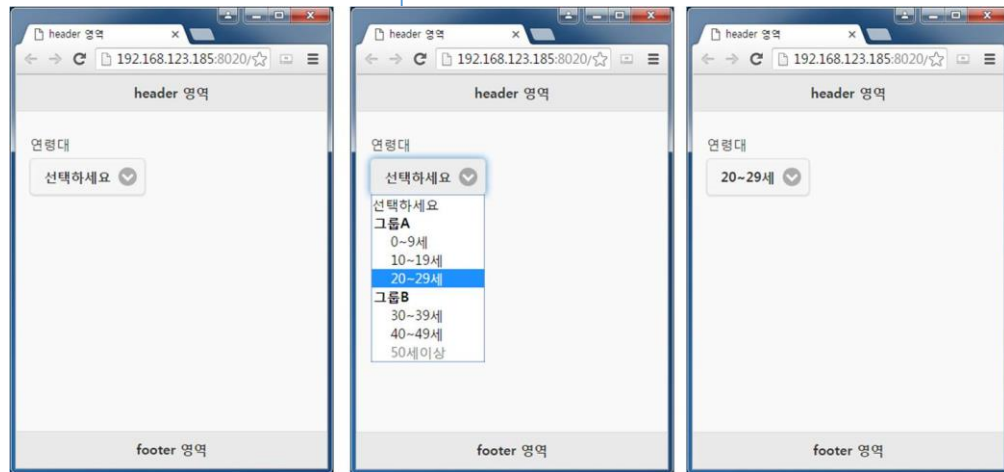


```

<div data-role="page" id="page1" >
  <div data-role="header" >
    <h1>header 영역</h1>

  </div>
  <div data-role="content" >
    <div data-role="fieldcontain">
      <label for="age1">연령대</label>
      <select name="age1" id="age1" data-inline="true">
        <option value="">선택하세요</option>
        <optgroup label="그룹A">
          <option value="num1">0~9세</option>
          <option value="num2">10~19세</option>
          <option value="num3">20~29세</option>
        </optgroup>
        <optgroup label="그룹B">
          <option value="num4">30~39세</option>
          <option value="num5">40~49세</option>
          <option value="num6" disabled="disabled">50세 이상</option>
        </optgroup>
      </select>
    </div>
  </div>
  <div data-role="footer" data-position="fixed">
    <h4>footer 영역</h4>
  </div>
</div>

```



선택 메뉴(3)

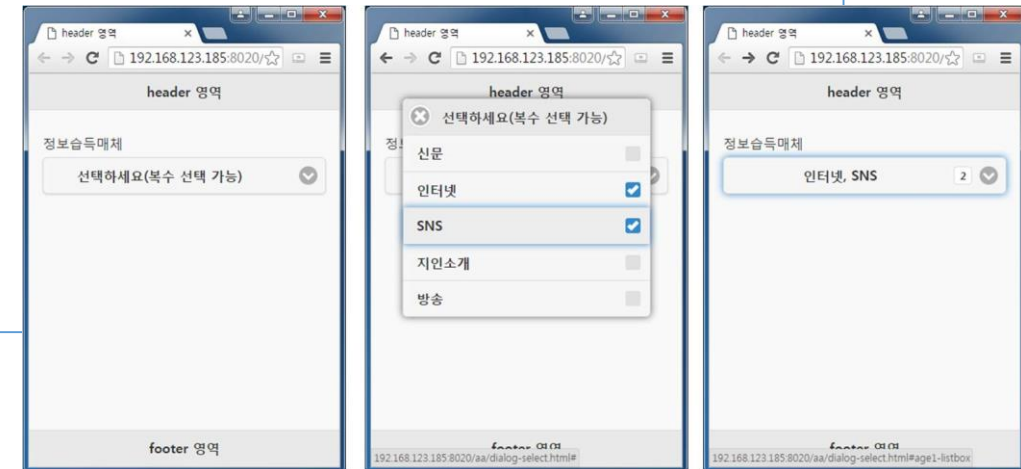
- 변형 선택 메뉴

- <select> 태그 안에 data-native-menu="false" 속성을 지정
- 선택 메뉴가 대화 상자와 비슷한 모양으로 표시(조금 더 깔끔한 모양)
- 'multiple' 속성을 설정 : 항목을 동시에 여러 개를 선택 가능
- 여러 항목을 선택하고 선택 메뉴를 닫으면 선택 항목들과 선택된 개수 추가로 표시

```
<select name="변수명" id="아이디" data-native-menu="false">  
    ... 생략 ...  
</select>
```



```
<div data-role="page" id="page1" >
  <div data-role="header" >
    <h1>header 영역 </h1>
  </div>
  <div data-role="content" >
    <div data-role="fieldcontain">
      <label for="age1">정보습득매체 </label>
      <select name="age1" id="age1" data-native-menu="false" multiple="multiple">
        <option value=" " data-placeholder="true">선택하세요(복수 선택 가능)</option>
        <option value="num1">신문</option>
        <option value="num2">인터넷</option>
        <option value="num3">SNS</option>
        <option value="num4">지인소개 </option>
        <option value="num5">방송</option>
      </select>
    </div>
  </div>
  <div data-role="footer" data-position="fixed">
    <h4>footer 영역</h4>
  </div>
</div>
```



- 변형 선택 메뉴

- <select> 태그 안에 data-native-menu="false" 속성을 지정
- 선택 메뉴가 대화 상자와 비슷한 모양으로 표시(조금 더 깔끔한 모양)
- 'multiple' 속성을 설정 : 항목을 동시에 여러 개를 선택 가능
- 여러 항목을 선택하고 선택 메뉴를 닫으면 선택 항목들과 선택된 개수 추가로 표시

```
<select name="변수명" id="아이디" data-native-menu="false">  
    ... 생략 ...  
</select>
```



5.1 독도 소개 웹앱 : dokdoApp



Reference

- ✓ 모바일 웹 웹 앱 하이브리드 앱 입문, 박성진, 생능출판
- ✓ 명품 웹 프로그래밍, 황기태, 생능출판

