

12장

문서 객체 모델과 이벤트 다루기

12.1 문서 객체 모델 이해하기

12.2 노드 선택하기

12.3 노드 조작하기

12.4 노드 추가/삭제하기

12.5 폼 조작하기

12.6 이벤트 다루기

12.7 이벤트 객체와 this

12.4 노드 추가/삭제하기

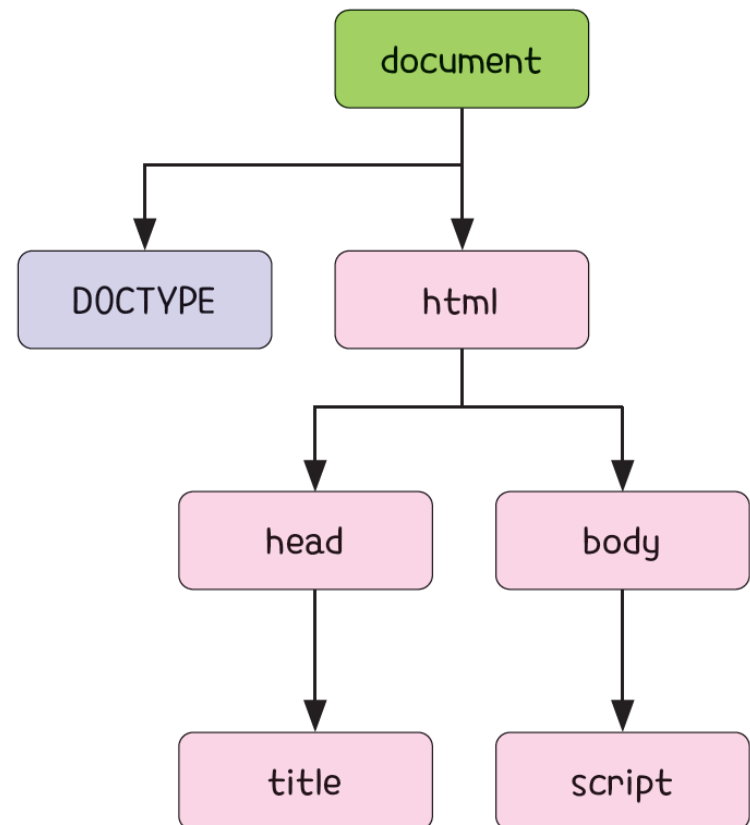
1. 노드 추가하기

- 노드 생성
 - createElement() : 요소 노드를 생성
 - createTextNode() : 텍스트 노드를 생성
 - createAttribute() : 속성 노드를 생성
- 노드 연결
 - <기준 노드>.appendChild(<자식 노드>) : 기준 노드에 자식 노드를 연결
 - <기준 노드>.setAttributeNode(<속성 노드>) : 기준 노드에 속성 노드를 연결

12.4 노드 추가/삭제하기

```
<!DOCTYPE html>
<html>
<head>
| <title>Create Node</title>
</head>
<body>
| <script></script>
</body>
</html>
```

그림 12-15 예제 코드의 DOM 트리

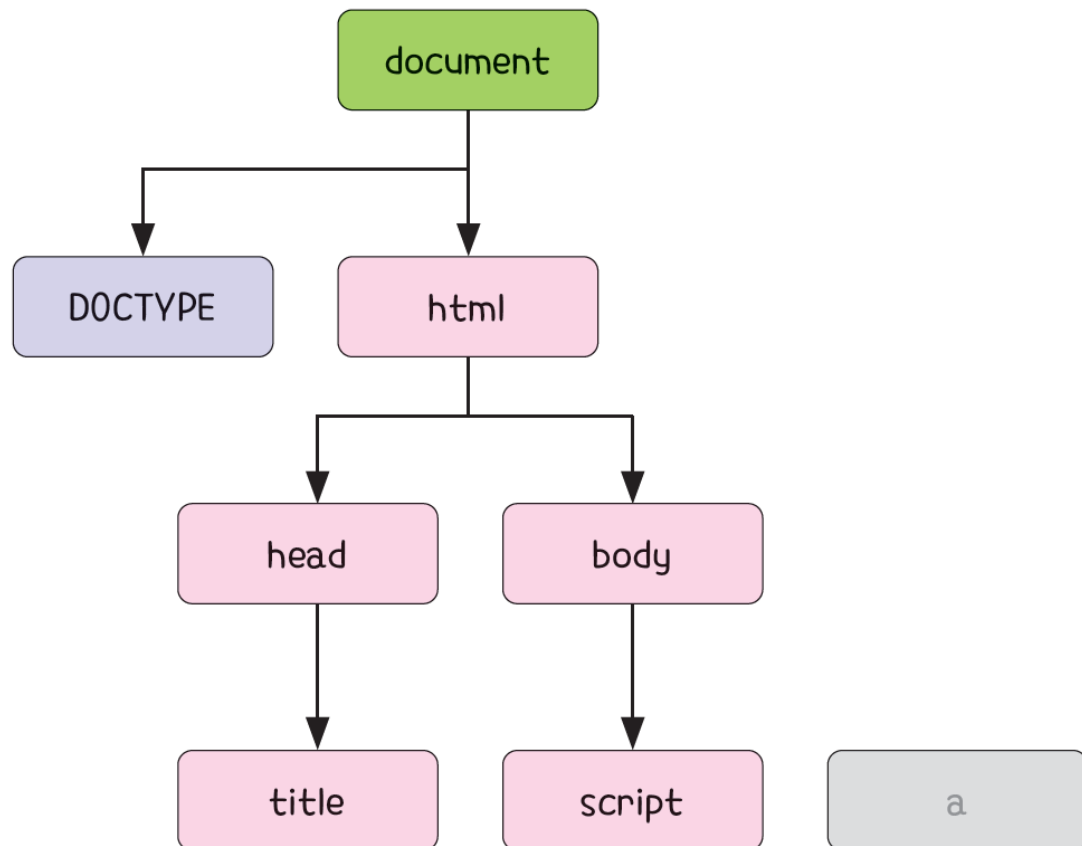


12.4 노드 추가/삭제하기

그림 12-16 a 요소 노드 생성

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

<script>
  // 요소 노드 생성하기
  const aEl = document.createElement("a");
```

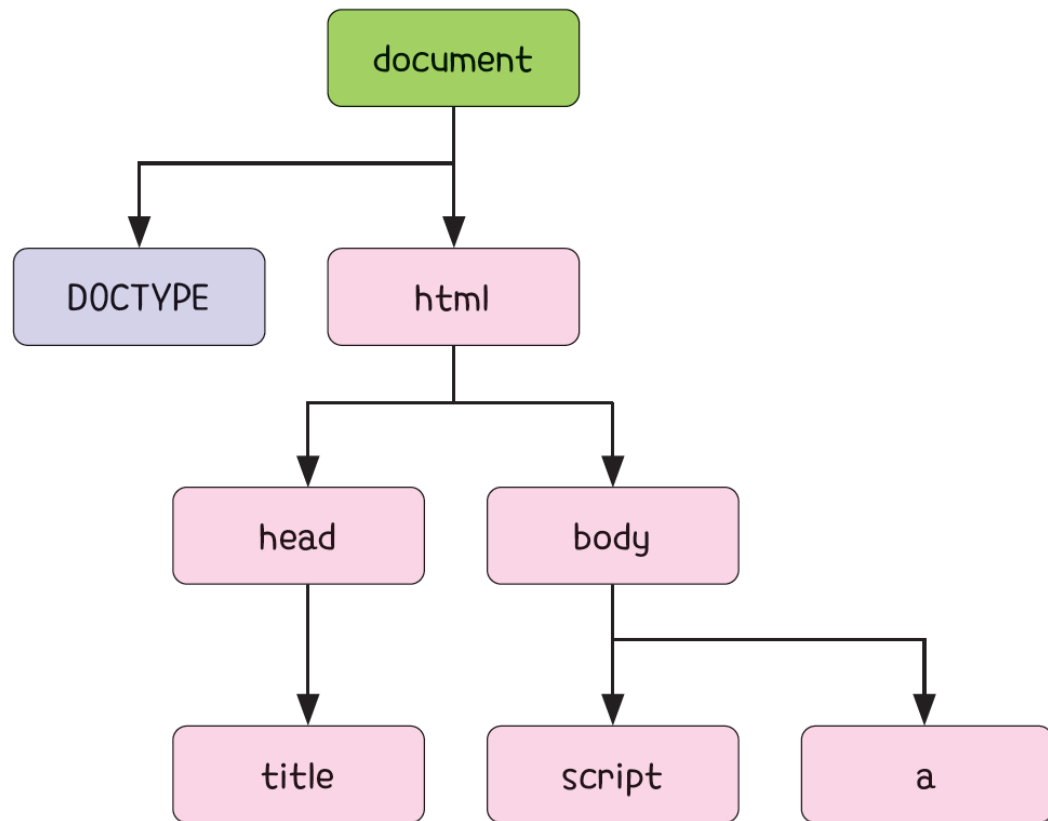


12.4 노드 추가/삭제하기

그림 12-17 a 요소 노드 연결

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

<script>
  // 요소 노드 생성하기
  const aEl = document.createElement("a");
  document.body.appendChild(aEl);
```



12.4 노드 추가/삭제하기

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

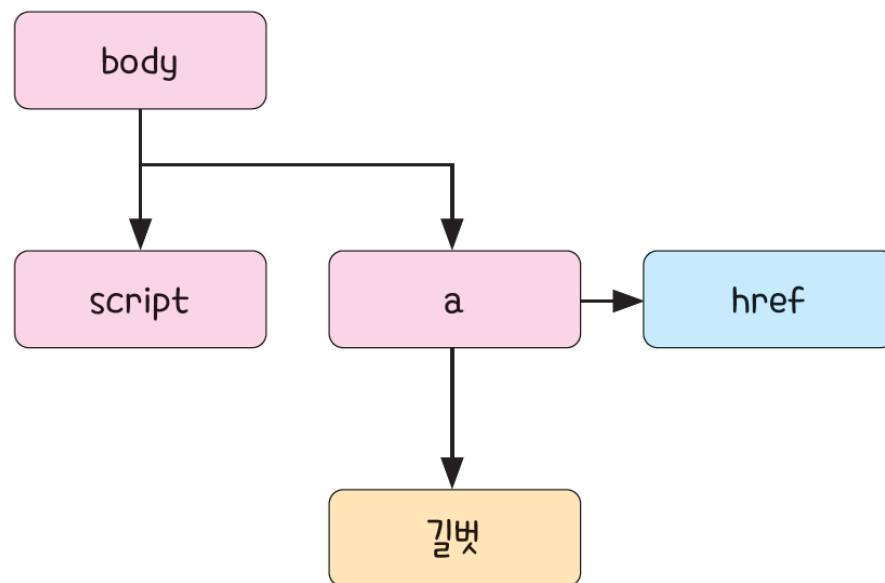
  <script>
    // 요소 노드 생성하기
    const aEl = document.createElement("a");
    document.body.appendChild(aEl);

    // 텍스트 노드 추가하기
    const txtEl = document.createTextNode("길벗");
    document.querySelector("a").appendChild(txtEl);

    // href 속성 추가하기
    const hrefAttr = document.createAttribute("href");
    hrefAttr.value = 'https://www.gilbut.co.kr';
    document.querySelector("a").setAttributeNode(hrefAttr);

  </script>
</body>
</html>
```

그림 12-21 속성 노드 연결



12.4 노드 추가/삭제하기

2. 노드 삭제하기

- 부모 노드에 연결된 자식 노드를 삭제

형식 <부모 노드>.removeChild(<자식 노드>)

```
<!DOCTYPE html>
<html lang="ko">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>remove Child</title>
</head>
<body>
  <p>text 1</p>
  <a href="https://www.gilbut.co.kr">길벗</a>
  <a href="https://www.sucoding.kr">수코딩</a>
  <script>
    const pEl = document.querySelector("p");
    pEl.parentNode.removeChild(pEl);
  </script>
</body>
</html>
```

12.4 노드 추가/삭제하기

```
<!DOCTYPE html>
<html lang="ko">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>remove Child</title>
</head>
<body>
  <p>text 1</p>
  <a href="https://www.gilbut.co.kr">길벗</a>
  <a href="https://www.sucoding.kr">수코딩</a>
  <script>
    const childNodes = document.body.childNodes;
    childNodes.forEach((node)=>{
      if(node.nodeName === "A")
        node.parentNode.removeChild(node);
    })
  </script>
</body>
</html>
```


12.5 폼 조작하기

1. form 태그 선택하기

- forms 속성 사용하기

- 모든 form 태그를 참조하는 HTMLCollection 객체가 반환
- 반환된 HTMLCollection 객체를 사용해 화면에 있는 form 요소 노드를 선택

- name 속성 사용하기

- form 태그의 name 속성값으로 form 요소 노드를 선택

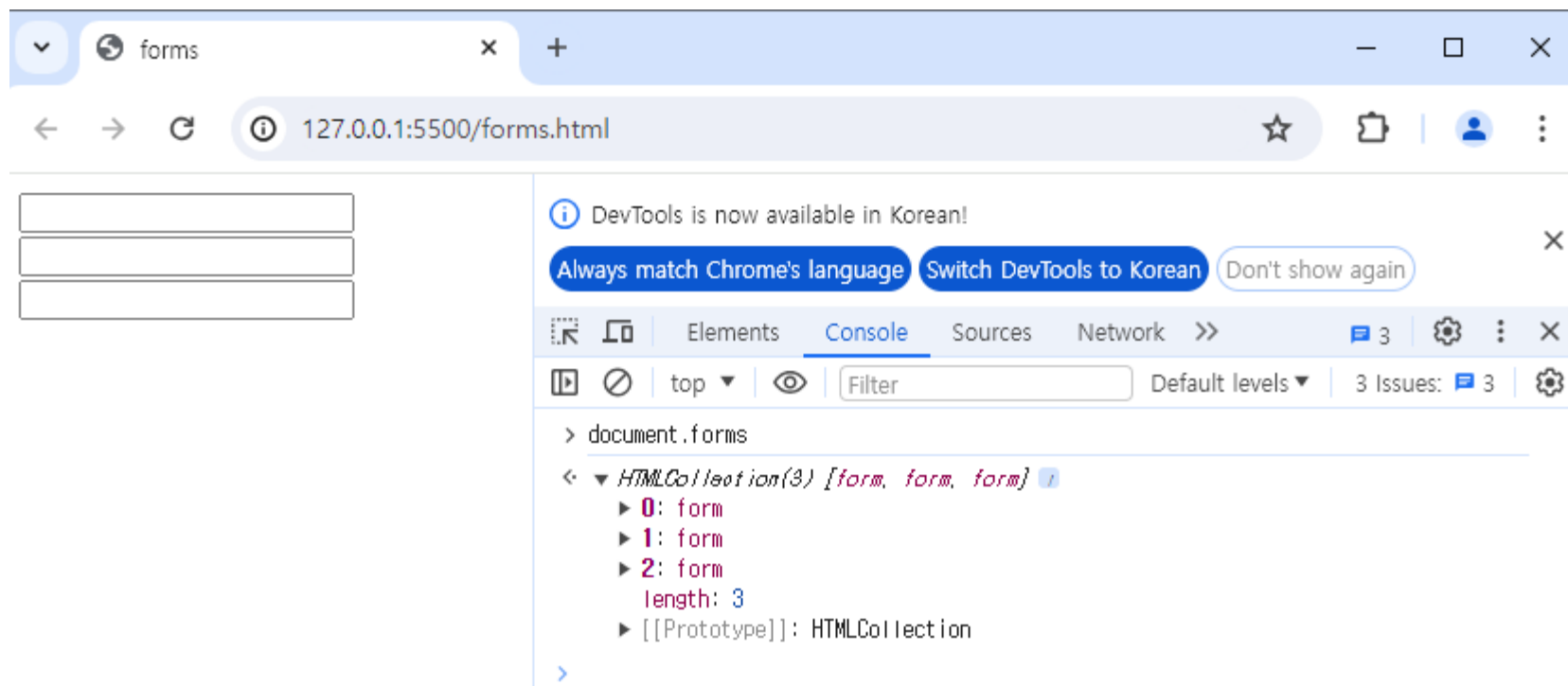
12.5 폼 조작하기

1. form 태그 선택하기

- forms 속성 사용하기 예제

```
<!DOCTYPE html>
<html lang="ko">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>forms</title>
</head>
<body>
  <form>
    <input type="text">
  </form>
  <form>
    <input type="text">
  </form>
  <form>
    <input type="text">
  </form>
</body>
</html>
```

12.5 폼 조작하기



The screenshot shows a web browser window with a single tab titled "forms". The address bar displays the URL "127.0.0.1:5500/forms.html". On the left side of the browser, there are three empty text input fields stacked vertically. On the right side, the Chrome DevTools interface is open, with the "Console" tab selected. A notification at the top of the console states "DevTools is now available in Korean!" with buttons for "Always match Chrome's language", "Switch DevTools to Korean", and "Don't show again". Below the notification, the console shows the following log:

```
> document.forms
< HTMLCollection(3) [form, form, form]
  ▶ 0: form
  ▶ 1: form
  ▶ 2: form
  length: 3
  ▶ [[Prototype]]: HTMLCollection
```

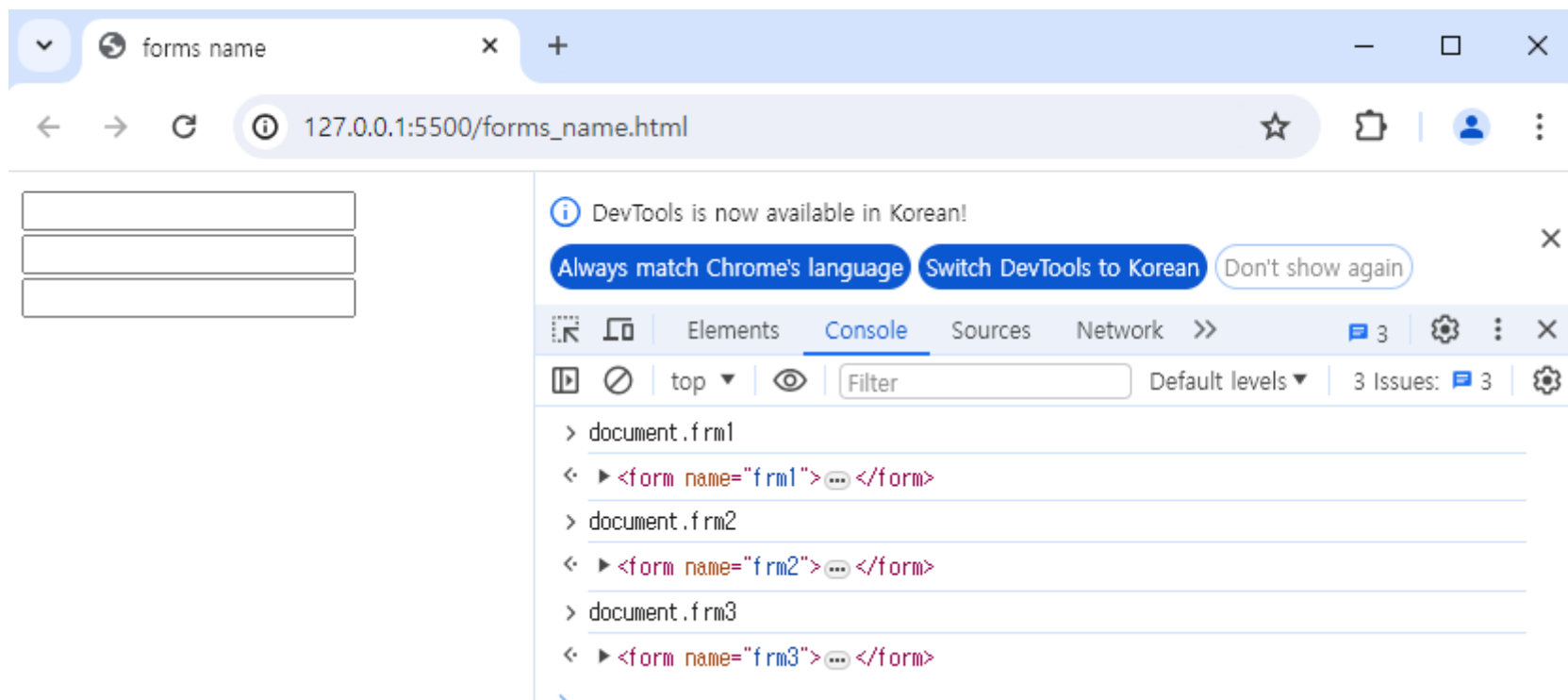
12.5 폼 조작하기

1. form 태그 선택하기

- name 속성 사용하기 예제

```
<!DOCTYPE html>
<html lang="ko">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>forms name</title>
</head>
<body>
  <form name="frm1">
    <input type="text">
  </form>
  <form name="frm2">
    <input type="text">
  </form>
  <form name="frm3">
    <input type="text">
  </form>
</body>
</html>
```

12.5 폼 조작하기



The screenshot shows a web browser window with a single tab titled "forms name". The address bar displays the URL "127.0.0.1:5500/forms_name.html". The page content consists of three empty text input fields stacked vertically.

The Chrome DevTools Console is open, showing the "Elements" panel. The DOM tree is expanded to show the form elements:

- `document.frm1`
 - `<form name="frm1">...</form>`
- `document.frm2`
 - `<form name="frm2">...</form>`
- `document.frm3`
 - `<form name="frm3">...</form>`

12.5 폼 조작하기

2. 폼 요소 선택하기(input, select, button태그 등)

- Elements 속성을 사용하는 방법
 - elements 속성 : form 요소 노드의 하위 노드 중 폼 요소를 반환
 - 반환된 객체를 사용해 개별 폼 요소 노드에 인덱스로 접근
- name 속성을 사용하는 방법
 - 폼 요소의 name 속성값으로 폼 요소 노드에 바로 접근

12.5 폼 조작하기

2. 폼 요소 선택하기(input, select, button태그 등)

- Elements 속성을 사용하는 방법 예제

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>forms_element</title>
</head>
<body>
  <form name="frm1">
    <label for="uname">이름</label>
    <input type="text" id="uname" name="uname">
    <label for="age">나이</label>
    <input type="text" id="age" name="age">
    <label for="gender">성별</label>
    <select id="gender" name="gender">
      <option value="male">male</option>
      <option value="female">female</option>
    </select>
    <button type="submit">전송</button>
  </form>
</body>
</html>
```

12.5 폼 조작하기

The screenshot shows a web browser window with a single tab titled 'forms_element'. The address bar displays '127.0.0.1:5500/forms_element.html'. The web page contains a form with the following elements:

- A text input field labeled '이름' (Name).
- A text input field labeled '나이' (Age).
- A dropdown menu labeled '성별' (Gender) with 'male' selected.
- A button labeled '전송' (Submit).

The Chrome DevTools Console is open, showing the following log:

```
> document.forms.elements
< HTMLFormControlsCollection(4) [input#uname, input#age, select#gender, button, ...]
  ▶ 0: input#uname
  ▶ 1: input#age
  ▶ 2: select#gender
  ▶ 3: button
  ▶ age: input#age
  ▶ gender: select#gender
  ▶ uname: input#uname
  ▶ length: 4
  ▶ [[Prototype]]: HTMLFormControlsCollection
```


12.5 폼 조작하기

2. 폼 요소 선택하기(input, select, button태그 등)

- name 속성을 사용하는 방법 예제

The screenshot shows a web browser window with a single tab titled 'forms_element'. The address bar displays '127.0.0.1:5500/forms_element.html'. The web page contains a form with the following elements:

- A text input field labeled '이름' (Name).
- A text input field labeled '나이' (Age).
- A dropdown menu labeled '성별' (Gender) with 'male' selected.
- A button labeled '전송' (Submit).

The Chrome DevTools Console is open, showing the following JavaScript code and its output:

```
> document.frm1.elements[0]
< <input type="text" id="uname" name="uname">
> document.frm1.elements['uname']
< <input type="text" id="uname" name="uname">
> |
```

12.5 폼 조작하기

The screenshot shows a web browser window with a single tab titled 'forms_element'. The address bar displays the URL '127.0.0.1:5500/forms_element.html'. The webpage content includes a form with the following elements:

- A label '이름' (Name) followed by a text input field.
- A label '나이' (Age) followed by a text input field.
- A label '성별' (Gender) followed by a dropdown menu currently showing 'male'.
- A '전송' (Submit) button.

The Chrome DevTools Console is open, showing the following log entries:

```
> document.frm1.uname
< <input type="text" id="uname" name="uname">
> document.frm1.age
< <input type="text" id="age" name="age">
> document.forms[0].gender
< ▶ <select id="gender" name="gender">... </select>
```

12.5 폼 조작하기

3. 폼 요소의 입력값 다루기

- 한 줄 입력 요소 / 여러 줄 입력 요소
 - 한줄입력 요소는 input태그의 type속성값을 text, password, number, url, search, email 등으로 지정했을때 표시되는 폼요소
 - 여러줄 입력 요소는 textarea태그를 사용하면 표시되는 폼요소
 - value 속성 참조 : 입력한 값을 가져오기
 - value 속성에 값 할당 : 새로운 값을 설정
- 체크박스 요소 / 라디오버튼 요소
 - checked 속성 : 체크 또는 선택 상태를 확인
 - checked 속성에 true 할당 : 체크 또는 선택 상태를 기본으로 설정

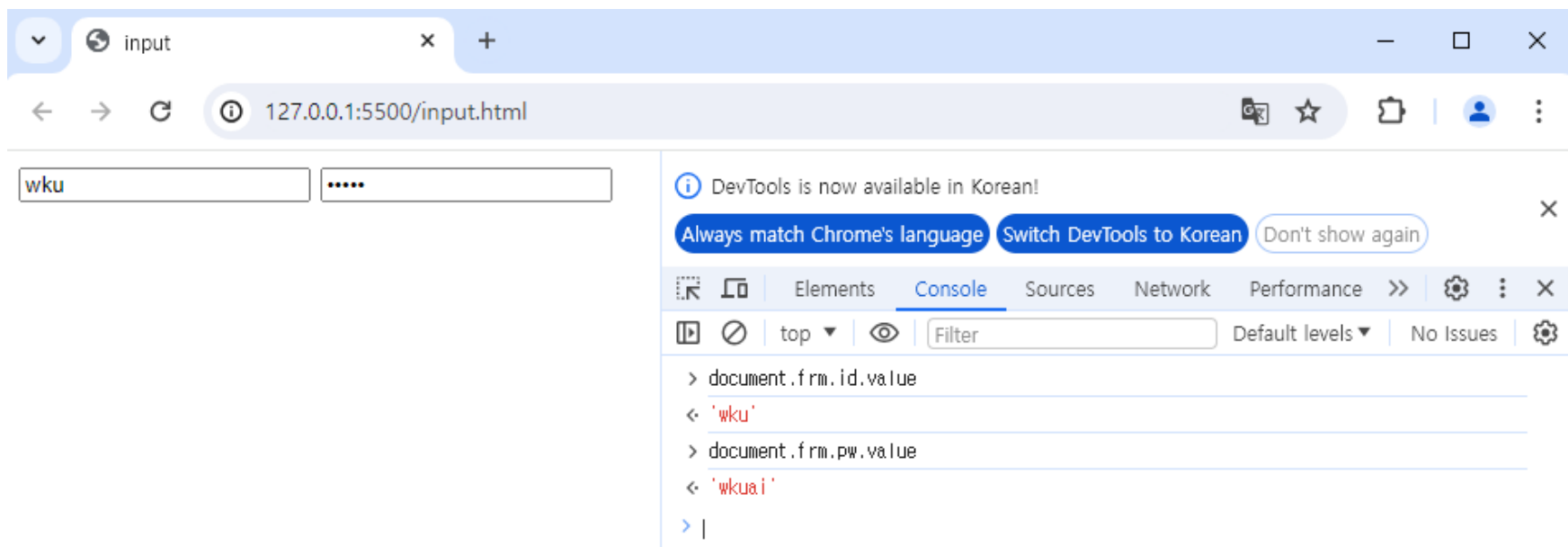
12.5 폼 조작하기

3. 폼 요소의 입력값 다루기

- 한 줄 입력 요소 예제

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>input</title>
</head>
<body>
  <form name="frm">
    <input type="text" name="id">
    <input type="password" name="pw">
  </form>
</body>
</html>
```

12.5 폼 조작하기

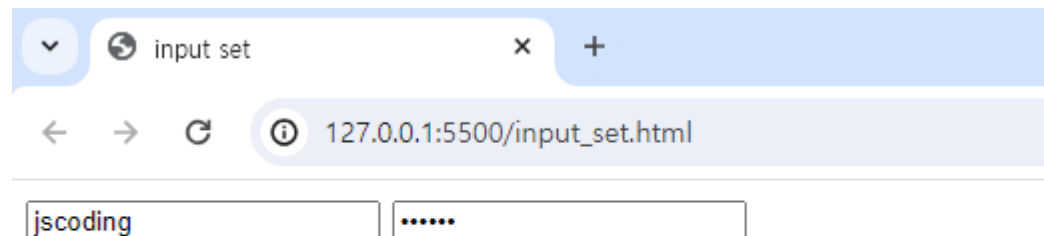


12.5 폼 조작하기

3. 폼 요소의 입력값 다루기

- 한 줄 입력 요소 예제(value속성에 값을 자바스크립트로 직접 삽입)

```
<!DOCTYPE html>
<html lang="ko">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>input set</title>
</head>
<body>
  <form name="frm">
    <input type="text" name="id">
    <input type="password" name="pw">
  </form>
  <script>
    document.frm.id.value = 'jscoding';
    document.frm.pw.value = 'aaaccc';
  </script>
</body>
</html>
```

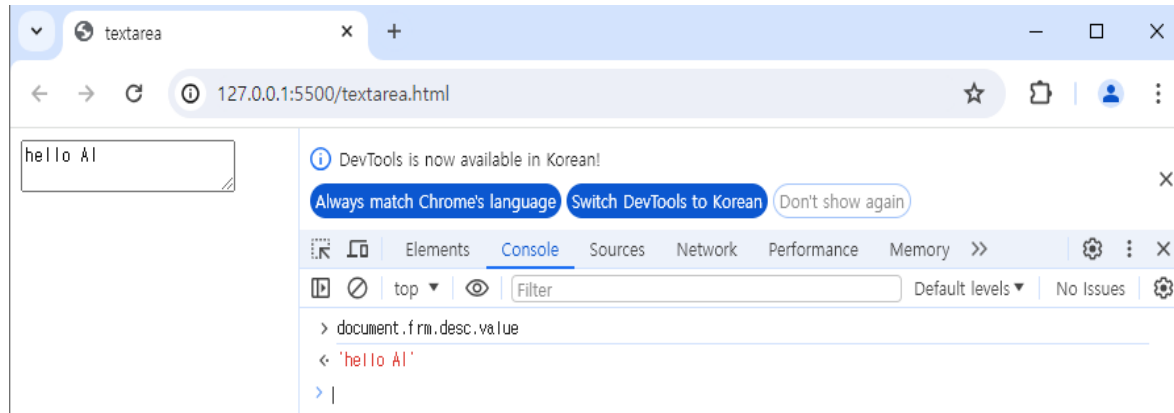


12.5 폼 조작하기

3. 폼 요소의 입력값 다루기

- 여러 줄 입력 요소 예제(textarea)

```
<!DOCTYPE html>
<html lang="ko">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>textarea</title>
</head>
<body>
  <form name="frm">
    <textarea name="desc"></textarea>
  </form>
</body>
</html>
```



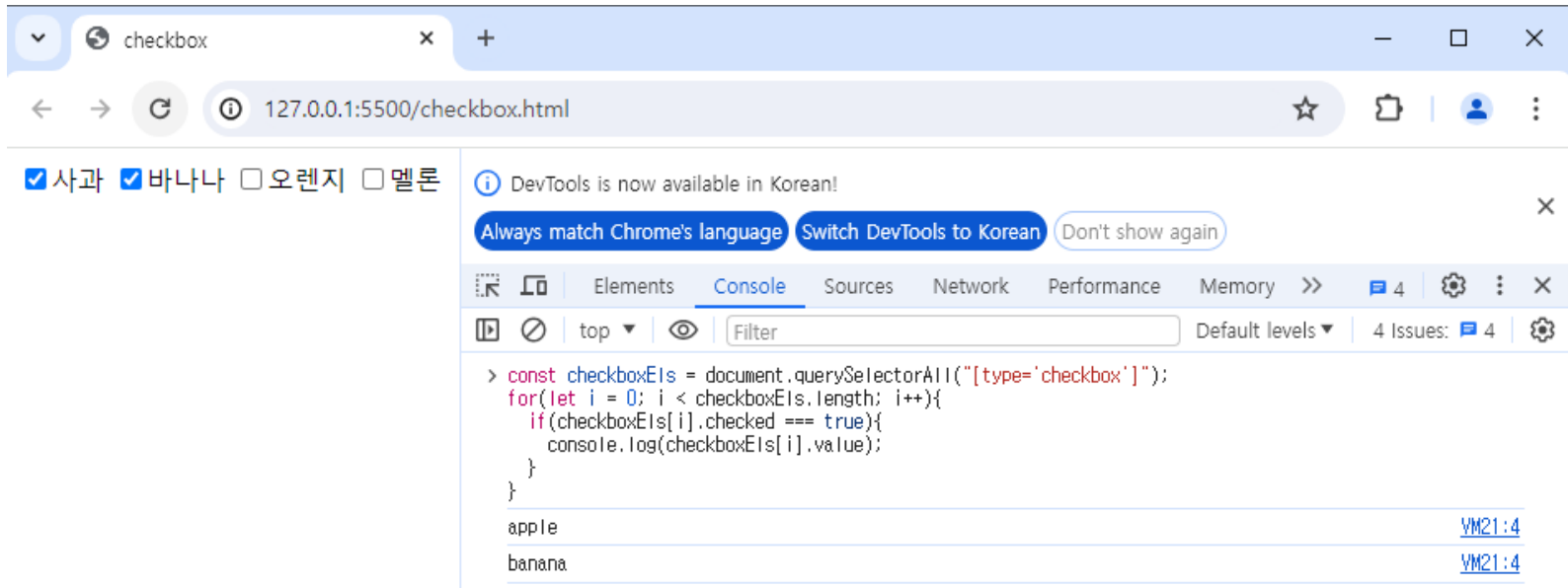
12.5 폼 조작하기

3. 폼 요소의 입력값 다루기

- 체크박스 요소 예제

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>checkbox</title>
</head>
<body>
  <form>
    <label><input type="checkbox" value="apple">사과</label>
    <label><input type="checkbox" value="banana">바나나</label>
    <label><input type="checkbox" value="orange">오렌지</label>
    <label><input type="checkbox" value="melon">멜론</label>
  </form>
</body>
</html>
```


12.5 폼 조작하기

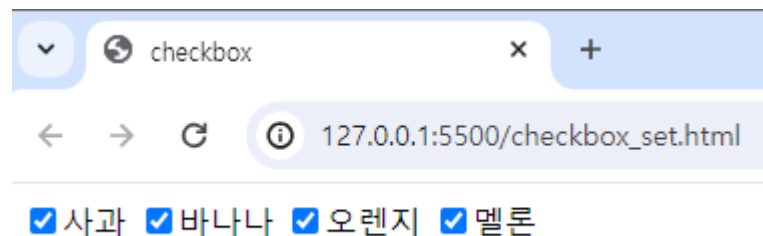


```
const checkboxEls =
document.querySelectorAll("[type='checkbox']");
for(let i = 0; i < checkboxEls.length; i++){
  if(checkboxEls[i].checked === true){
    console.log(checkboxEls[i].value);
  }
}
```

12.5 폼 조작하기

- 체크박스 요소 응용 예제

```
<!DOCTYPE html>
<html lang="ko">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>checkbox</title>
</head>
<body>
  <form>
    <label><input type="checkbox" value="apple">사과</label>
    <label><input type="checkbox" value="banana">바나나</label>
    <label><input type="checkbox" value="orange">오렌지</label>
    <label><input type="checkbox" value="melon">멜론</label>
  </form>
  <script>
    const checkboxEls = document.querySelectorAll("[type='checkbox']");
    for(let i = 0; i < checkboxEls.length; i++){
      checkboxEls[i].checked = true;
    }
  </script>
</body>
</html>
```



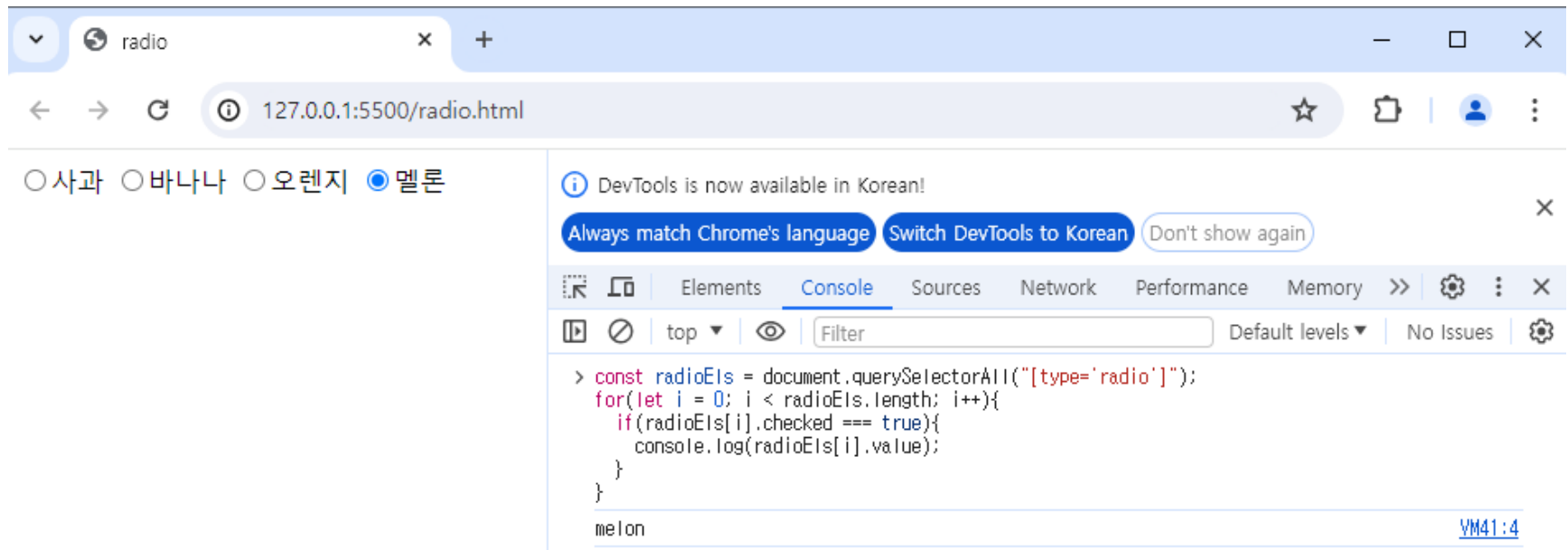
12.5 폼 조작하기

3. 폼 요소의 입력값 다루기

- 라디오버튼 요소 예제

```
<!DOCTYPE html>
<html lang="ko">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>radio</title>
</head>
<body>
  <form>
    <label><input type="radio" name="fruits" value="apple">사과</label>
    <label><input type="radio" name="fruits" value="banana">바나나</label>
    <label><input type="radio" name="fruits" value="orange">오렌지</label>
    <label><input type="radio" name="fruits" value="melon">멜론</label>
  </form>
</body>
</html>
```

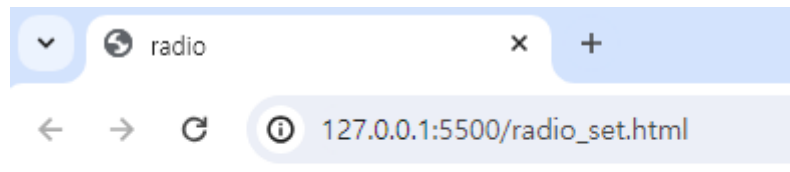
12.5 폼 조작하기



```
const radioEls = document.querySelectorAll("[type='radio']");
for(let i = 0; i < radioEls.length; i++){
  if(radioEls[i].checked === true){
    console.log(radioEls[i].value);
  }
}
```

12.5 폼 조작하기

- 라디오버튼 요소 응용 예제



☐ 사과 ☒ 바나나 ☐ 오렌지 ☐ 멜론

```
<!DOCTYPE html>
<html lang="ko">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>radio</title>
</head>
<body>
  <form>
    <label><input type="radio" name="fruits" value="apple">사과</label>
    <label><input type="radio" name="fruits" value="banana">바나나</label>
    <label><input type="radio" name="fruits" value="orange">오렌지</label>
    <label><input type="radio" name="fruits" value="melon">멜론</label>
  </form>
  <script>
    const radioEls = document.querySelectorAll("[type='radio']");
    for(let i = 0; i < radioEls.length; i++){
      if(radioEls[i].value === 'banana'){
        radioEls[i].checked = true;
      }
    }
  </script>
</body>
</html>
```

12.5 폼 조작하기

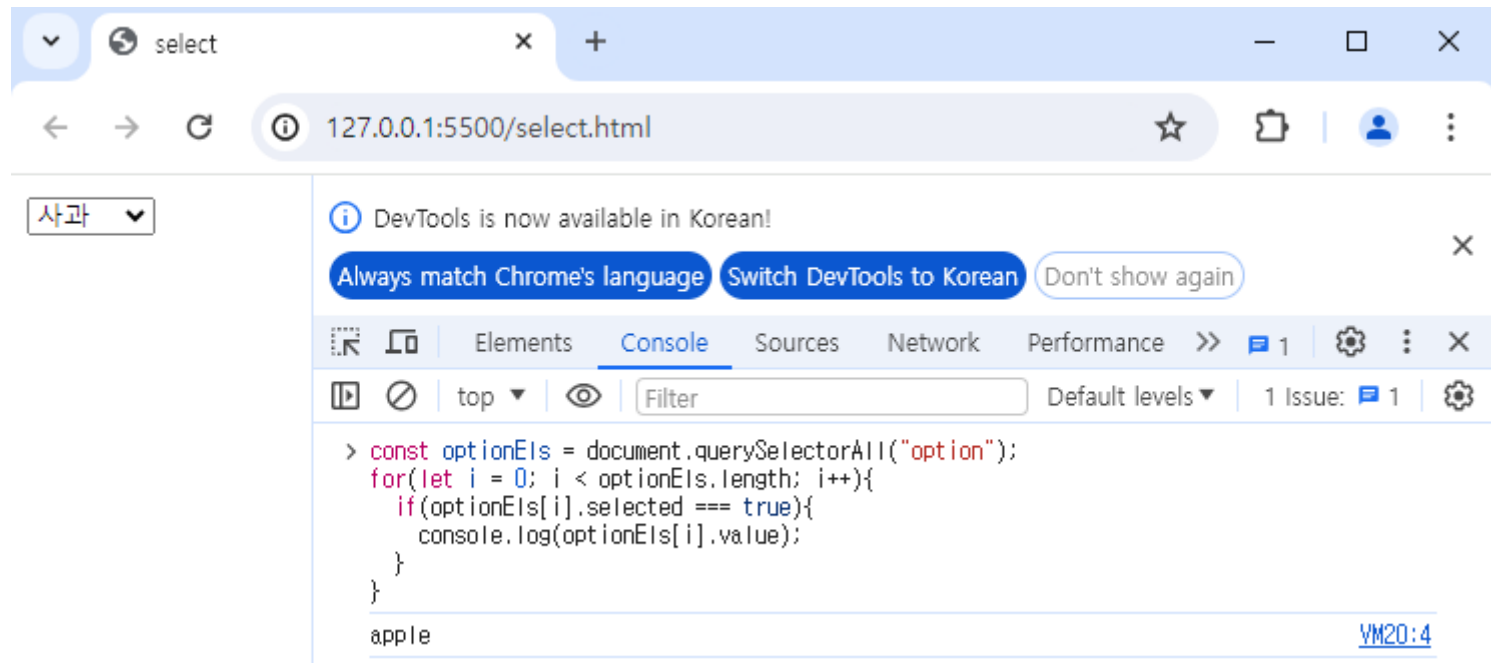
- 콤보박스 요소
 - selected 속성 : 항목의 선택 상태를 확인
 - selected 속성에 true 할당 : 항목을 선택 상태로 설정
- 파일 업로드 요소
 - FileList 객체 : 요소의 값 가져오기
- 폼 요소 관련 기타 메서드
 - submit() : 폼 요소의 값 전송(submit)
 - focus() : 폼 요소에 포커스(커서) 이동

12.5 폼 조작하기

- 콤보박스 요소 예제

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>select</title>
</head>
<body>
  <form>
    <select>
      <option value="apple">사과</option>
      <option value="banana">바나나</option>
      <option value="orange">오렌지</option>
      <option value="melon">멜론</option>
    </select>
  </form>
</body>
</html>
```

12.5 폼 조작하기

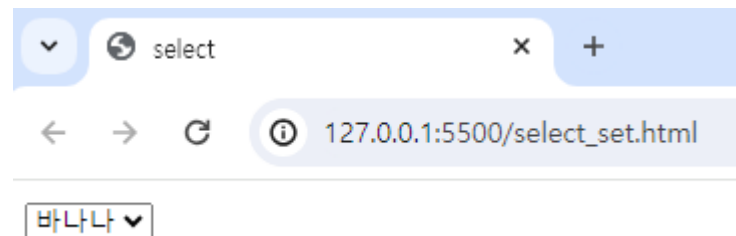


```
const optionEls = document.querySelectorAll("option");
for(let i = 0; i < optionEls.length; i++){
  if(optionEls[i].selected === true){
    console.log(optionEls[i].value);
  }
}
```


12.5 폼 조작하기

- 콤보박스 요소 응용 예제

```
<!DOCTYPE html>
<html lang="ko">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>select</title>
</head>
<body>
  <form>
    <select name="opt">
      <option value="apple">사과</option>
      <option value="banana">바나나</option>
      <option value="orange">오렌지</option>
      <option value="melon">멜론</option>
    </select>
  </form>
  <script>
    const optionEls = document.querySelectorAll("option")
    for(let i = 0; i < optionEls.length; i++){
      if(optionEls[i].value === 'banana'){
        optionEls[i].selected = true;
      }
    }
  </script>
</body>
</html>
```

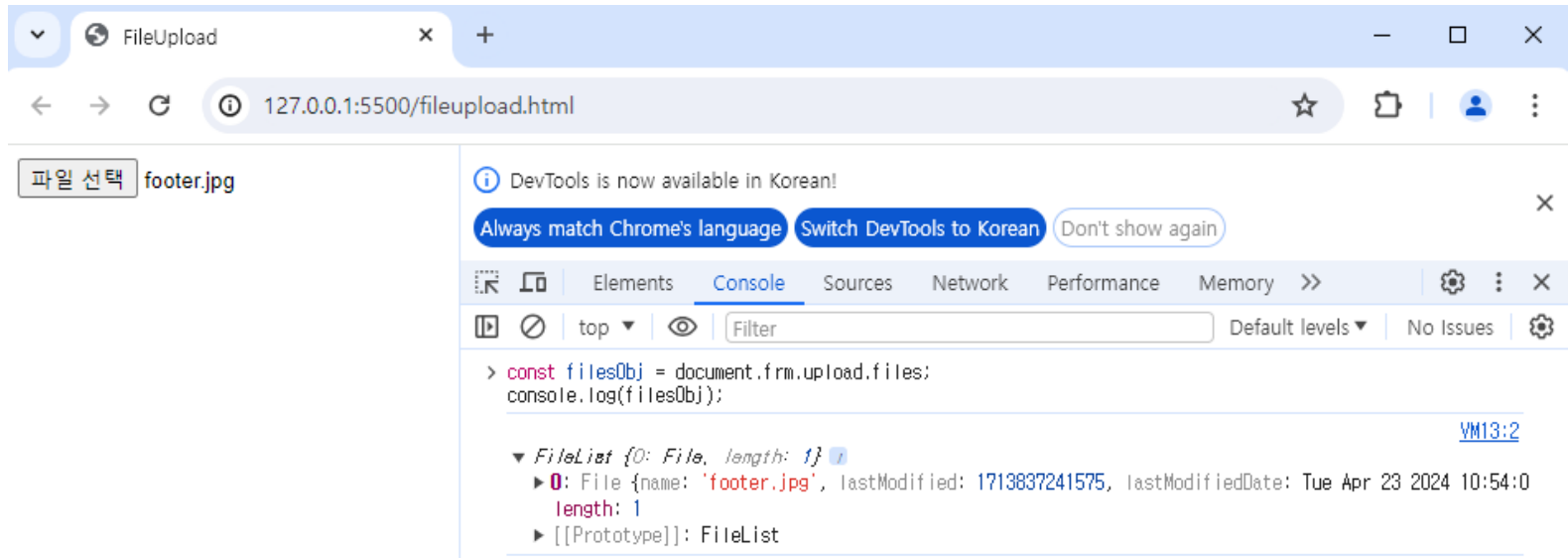


12.5 폼 조작하기 과제

- 파일 업로드 요소 예제

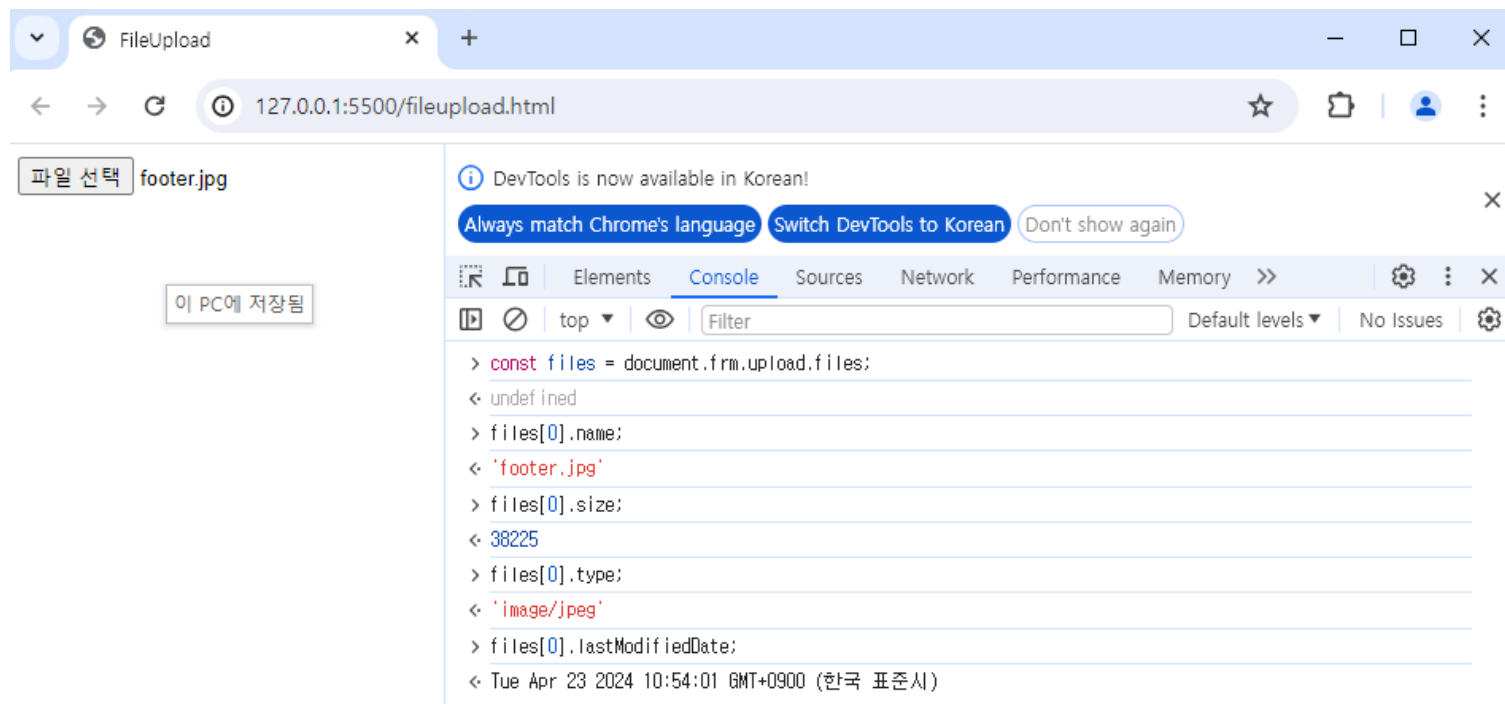
```
<!DOCTYPE html>
<html lang="ko">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>FileUpload</title>
</head>
<body>
  <form name="frm">
    <input type="file" name="upload">
  </form>
</body>
</html>
```

12.5 폼 조작하기 과제



```
const filesObj = document.frm.upload.files;
console.log(filesObj);
```

12.5 폼 조작하기 과제(제출과제)



```
const files = document.frm.upload.files;
files[0].name; // 파일 이름
files[0].size; // 파일 크기
files[0].type; // 파일 타입
files[0].lastModifiedDate; // 파일 마지막 수정일
```