

연습문제

문제 1: 여러 `

` 요소에서 첫 번째와 두 번째 요소 추출하기

상황

HTML 문서에 여러 개의 `

` 요소가 있습니다.

요구사항

1. `document.querySelectorAll('div')`를 사용해 모든 `

` 요소를 선택합니다.
2. 배열 구조 분해 할당을 사용하여 첫 번째 요소와 두 번째 요소를 각각 `firstDiv`와 `secondDiv` 변수에 할당합니다.
3. `firstDiv`의 텍스트 콘텐츠를 `"Hello"`로 변경합니다.

예시 코드

```
```js
const divElements = document.querySelectorAll('div');
const [firstDiv, secondDiv] = divElements;
firstDiv.textContent = "Hello";
...

```

### 문제 2: 단일 요소의 속성을 객체 구조 분해로 추출하기

### 상황

HTML 문서에 다음과 같이 정의된 `

` 요소가 있습니다.

```
```html
<p id="info" class="text-info">Some text</p>
...
```
```

## 요구사항

1. `document.getElementById("info")`로 해당 `

` 요소를 선택합니다.
2. 객체 구조 분해 할당을 사용하여 요소의 `textContent`와 `className`을 각각 `content`와 `cls` 변수에 할당합니다.
3. 콘솔에 두 변수를 출력합니다.

## 예시 코드

```
```js
const infoElement = document.getElementById("info");
const { textContent: content, className: cls } = infoElement;
console.log(content, cls);
...
---
```

문제 3: 이벤트 객체에서 대상(target) 추출하기

상황

HTML 문서에 다음과 같이 버튼이 있습니다.

```
```html
<button id="btn">Click me</button>
...

```

## 요구사항

1. 버튼에 `click` 이벤트 리스너를 추가합니다.
2. 이벤트 핸들러 함수의 매개변수로 전달받은 `event` 객체에서 구조 분해 할당을 사용하여 `target`을 추출합니다.
3. 추출한 `target`의 `id`를 콘솔에 출력합니다.

예시 코드

```
```js
const button = document.getElementById("btn");

button.addEventListener("click", function(event) {

    const { target } = event;

    console.log(target.id);

});
```

```

### 문제 4: NodeList에서 특정 요소 추출하기

상황

HTML 문서에 5개의 `

` 요소가 있다고 가정합니다.

요구사항

1. `document.querySelectorAll('div')`를 사용해 모든 `

` 요소를 가져옵니다.
2. 배열 구조 분해 할당을 사용하여 첫 번째 요소 (`firstDiv`)와 세 번째 요소 (`thirdDiv`)를 추출합니다. (두 번째 요소는 건너뜁니다.)
3. `firstDiv`의 `backgroundColor`를 빨간색('red')으로, `thirdDiv`의 `backgroundColor`를 파란색('blue')으로 변경합니다.

예시 코드

```
```js
const divElements = document.querySelectorAll('div');

const [firstDiv, , thirdDiv] = divElements;

firstDiv.style.backgroundColor = 'red';
```

```
thirdDiv.style.backgroundColor = 'blue';
```

```
...
```

```
---
```

문제 5: HTMLCollection 배열 변환 후 첫 번째와 마지막 요소 추출하기

상황

HTML 문서에 `

` 요소 내부에 여러 `- ` 요소들이 있습니다.

요구사항

1. `document.getElementsByTagName('li')`를 사용해 HTMLCollection을 가져옵니다.
2. `Array.from()`을 사용해 HTMLCollection을 배열로 변환합니다.
3. 배열 구조 분해 할당을 사용하여 배열의 첫 번째 요소를 `firstItem`에 할당하고, 나머지 요소들 중 마지막 요소를 `lastItem`에 할당합니다.
4. 두 요소의 `textContent`를 콘솔에 출력합니다.

예시 코드

```
```js
const liCollection = document.getElementsByTagName('li');
const liArray = Array.from(liCollection);
const [firstItem, ...rest] = liArray;
const lastItem = rest[rest.length - 1];
console.log(firstItem.textContent, lastItem.textContent);
...

```

```

```

### ### 문제 6: 객체 속성 추출과 기본값 사용하기

## 상황

아래 객체는 HTML 요소의 속성 정보를 나타냅니다.

```
``js
const elementAttributes = { id: 'demo', className: 'sample-class', 'data-info': 'example' };
``
```

## 요구사항

1. 객체 구조 분해 할당을 사용하여 `id`는 그대로, `className`은 별칭 `cls`로, `data-info`는 `dataInfo`라는 변수에 할당합니다.
2. 만약 `data-info` 속성이 없다면, `dataInfo`에 기본값 `"none"`을 할당합니다.
3. 콘솔에 각 변수를 출력합니다.

## 예시 코드

```
``js
const { id, className: cls, 'data-info': dataInfo = "none" } = elementAttributes;

console.log(id, cls, dataInfo);
``
```

---

## ### 문제 7: 폼 요소의 입력값 추출하기

## 상황

HTML 문서에 다음과 같이 폼과 `` 요소들이 있습니다.

```
``html
<form id="myForm">

 <input name="username" value="JohnDoe">

 <input name="email" value="john@example.com">
``
```

</form>

...

## 요구사항

1. `document.getElementById("myForm")`를 사용해 폼 요소를 선택합니다.
2. 폼의 `elements` 프로퍼티를 사용하여 입력 요소들의 객체를 가져옵니다.
3. 객체 구조 분해 할당을 사용하여 그 객체에서 `username`과 `email` 속성을 추출합니다.
4. 각 입력 요소의 `value`를 콘솔에 출력합니다.

## 예시 코드

```
```js
const form = document.getElementById("myForm");
const { username, email } = form.elements;
console.log(username.value, email.value);
...
---
```

문제 8: 인라인 스타일 객체에서 속성 추출하기

상황

HTML 문서에 아래와 같이 인라인 스타일이 적용된 요소가 있습니다.

```
```html
<div id="styled" style="color: red; background-color: blue; font-size: 16px;"> </div>
...

```

### 요구사항

1. `document.getElementById("styled").style`로 스타일 객체를 가져옵니다.

2. 객체 구조 분해 할당을 사용하여 `color`, `backgroundColor`, `fontSize` 프로퍼티를 추출합니다.
3. 각 값을 콘솔에 출력합니다.

예시 코드

```
```js
const styleObj = document.getElementById("styled").style;
const { color, backgroundColor, fontSize } = styleObj;
console.log(color, backgroundColor, fontSize);
```
```

---

### ### 문제 9: 자식 요소 배열에서 일부와 나머지 추출하기

상황

HTML 문서에 `

요구사항

1. `document.getElementById("parent").children`으로 HTMLCollection을 가져옵니다.
2. `Array.from()`을 사용하여 HTMLCollection을 배열로 변환합니다.
3. 배열 구조 분해 할당을 사용해 배열의 첫 번째 자식은 `firstChild`, 두 번째 자식은 `secondChild`에 할당하고, 나머지 요소들은 배열 `others`에 저장합니다.
4. 콘솔에 각 변수의 값을 출력합니다.

예시 코드

```
```js
const childrenArray = Array.from(document.getElementById("parent").children);
const [firstChild, secondChild, ...others] = childrenArray;
console.log(firstChild, secondChild, others);
```
```

...

---

### ### 문제 10: 데이터 속성을 추출하여 이벤트에서 활용하기

#### 상황

HTML 문서에 다음과 같이 데이터 속성이 포함된 버튼이 있습니다.

```
```html
```

```
<button id="actionBtn" data-user-id="123" data-action="delete">Action</button>
```

```
```
```

#### 요구사항

1. 버튼에 `click` 이벤트 리스너를 추가합니다.
2. 이벤트 핸들러의 매개변수인 `event` 객체에서 구조 분해 할당을 사용해 `target`을 추출합니다.
3. `target.dataset` 객체에서 구조 분해 할당을 통해 `userId`와 `action` 속성을 추출합니다.
4. 이 두 값을 콘솔에 출력합니다.

#### 예시 코드

```
```js
```

```
const actionButton = document.getElementById("actionBtn");
```

```
actionButton.addEventListener("click", function(event) {
```

```
    const { target } = event;
```

```
    const { userId, action } = target.dataset;
```

```
    console.log(userId, action);
```

```
});
```


정답

문제 1

```
<!DOCTYPE html>
<html lang="ko">

<head>
  <meta charset="UTF-8">
  <title>문제 1 정답</title>
</head>

<body>
  <!-- 두 개의 div 요소 -->
  <div>Div 1</div>
  <div>Div 2</div>

  <script>
    const divElements = document.querySelectorAll('div');
    const [firstDiv, secondDiv] = divElements;
    firstDiv.textContent = "Hello";
  </script>
</body>

</html>
```

문제 2

```
<!DOCTYPE html>
<html lang="ko">

<head>
  <meta charset="UTF-8">
  <title>문제 2 정답</title>
</head>

<body>
  <p id="info" class="text-info">Some text</p>

  <script>
    const infoElement = document.getElementById("info");
    const { textContent: content, className: cls } = infoElement;
    console.log("문제 2:", content, cls);
  </script>
</body>

</html>
```

문제 3

```
<!DOCTYPE html>
<html lang="ko">
```

```

<head>
  <meta charset="UTF-8">
  <title>문제 3 정답</title>
</head>

<body>
  <button id="btn">Click me</button>

  <script>
    const button = document.getElementById("btn");
    button.addEventListener("click", function (event) {
      const { target } = event;
      console.log("문제 3:", target.id);
    });
  </script>
</body>

</html>

```

문제 4

```

<!DOCTYPE html>
<html lang="ko">

<head>
  <meta charset="UTF-8">
  <title>문제 4 정답</title>
  <style>
    /* div 요소들이 잘 보이도록 약간의 스타일 적용 */
    div {
      padding: 5px;
      margin: 3px;
    }
  </style>
</head>

<body>
  <!-- 5 개의 div 요소 -->
  <div>Div 1</div>
  <div>Div 2</div>
  <div>Div 3</div>
  <div>Div 4</div>
  <div>Div 5</div>

  <script>
    const divElements = document.querySelectorAll('div');
    const [firstDiv, , thirdDiv] = divElements;
    firstDiv.style.backgroundColor = 'red';
    thirdDiv.style.backgroundColor = 'blue';
  </script>
</body>

```

```
</html>
```

문제 5

```
<!DOCTYPE html>
<html lang="ko">

<head>
  <meta charset="UTF-8">
  <title>문제 5 정답</title>
</head>

<body>
  <ul>
    <li>Item 1</li>
    <li>Item 2</li>
    <li>Item 3</li>
  </ul>

  <script>
    const liCollection = document.getElementsByTagName('li');
    const liArray = Array.from(liCollection);
    const [firstItem, ...rest] = liArray;
    const lastItem = rest[rest.length - 1];
    console.log("문제 5:", firstItem.textContent, lastItem.textContent);
  </script>
</body>

</html>
```

문제 6

```
<!DOCTYPE html>
<html lang="ko">

<head>
  <meta charset="UTF-8">
  <title>문제 6 정답</title>
</head>

<body>
  <script>
    const elementAttributes = { id: 'demo', className: 'sample-class', 'data-info':
'example' };
    const { id, className: cls, 'data-info': dataInfo = "none" } = elementAttributes;
    console.log("문제 6:", id, cls, dataInfo);
  </script>
</body>

</html>
```

문제 7

```
<!DOCTYPE html>
<html lang="ko">

<head>
  <meta charset="UTF-8">
  <title>문제 7 정답</title>
</head>

<body>
  <form id="myForm">
    <input name="username" value="JohnDoe">
    <input name="email" value="john@example.com">
  </form>

  <script>
    const form = document.getElementById("myForm");
    const { username, email } = form.elements;
    console.log("문제 7:", username.value, email.value);
  </script>
</body>

</html>
```

문제 8

```
<!DOCTYPE html>
<html lang="ko">

<head>
  <meta charset="UTF-8">
  <title>문제 8 정답</title>
</head>

<body>
  <div id="styled" style="color: red; background-color: blue; font-size: 16px;">
    Styled Div
  </div>

  <script>
    const styleObj = document.getElementById("styled").style;
    const { color, backgroundColor, fontSize } = styleObj;
    console.log("문제 8:", color, backgroundColor, fontSize);
  </script>
</body>

</html>
```

문제 9

```
<!DOCTYPE html>
```

```

<html lang="ko">

<head>
  <meta charset="UTF-8">
  <title>문제 9 정답</title>
</head>

<body>
  <div id="parent">
    <div>Child 1</div>
    <div>Child 2</div>
    <div>Child 3</div>
    <div>Child 4</div>
  </div>

  <script>
    const parent = document.getElementById("parent");
    const childrenArray = Array.from(parent.children);
    const [firstChild, secondChild, ...others] = childrenArray;
    console.log("문제 9:", firstChild.textContent, secondChild.textContent,
others.map(child => child.textContent));
  </script>
</body>

</html>

```

문제 10

```

<!DOCTYPE html>
<html lang="ko">

<head>
  <meta charset="UTF-8">
  <title>문제 10 정답</title>
</head>

<body>
  <button id="actionBtn" data-user-id="123" data-action="delete">Action</button>

  <script>
    const actionButton = document.getElementById("actionBtn");
    actionButton.addEventListener("click", function (event) {
      const { target } = event;
      const { userId, action } = target.dataset;
      console.log("문제 10:", userId, action);
    });
  </script>
</body>

</html>

```