

hw03

정보통신공학과

2017112207

이건우

p1

The screenshot shows a Mac desktop environment. On the left, a Finder window displays a file named 'downlo'. On the right, a Code editor window titled 'practice_1.html - codes' is open, showing the following code:

```
<!DOCTYPE html>
<html>
<head>
<title>User-defined objects</title>
<script type="text/javascript">
function addPrice(amount) {
    this.price = amount;
}

function book(title, author) {
    this.title = title;
    this.author = author;
    this.addPrice = addPrice;
}
</script>
</head>
<body>
<script type="text/javascript">
var myBook = new book("trotzdem Ja zum Leben sagen", "Viktor Frankl");
myBook.addPrice(15000);

document.write("Book title is: " + myBook.title + "<br>");
document.write("Book author is: " + myBook.author + "<br>");
document.write("Book price is: " + myBook.price + "<br>");
</script>
</body>
</html>
```

The browser window on the left shows the output of the code: "Book title is: trotzdem Ja zum Leben sagen", "Book author is: Viktor Frankl", and "Book price is: 15000".

p2

The screenshot shows the Xcode interface with the file `practice_2.html` open. The code demonstrates the `toString()` method for numbers in different bases:

```
<!DOCTYPE html>
<html>
<body>
<p> The toString() method can output numbers as base 16 (hex),  

<p id="demo"> </p>
<button onclick="myFunction()"> Try it </button>
<script>
function myFunction() {
    var myNumber = 128;
    document.getElementById("demo").innerHTML = "128 = " +
    + myNumber + " Decimal, <br>" +
    + myNumber.toString(16) + " Hexadecimal, <br>" +
    + myNumber.toString(8) + " Octal, <br>" +
    + myNumber.toString(2) + " Binary"
}
</script>
</body>
</html>
```

초기 화면

The screenshot shows the Xcode interface with the file `practice_2.html` open. After clicking the "Try it" button, the browser window displays the output of the script:

128 = 128
128 Decimal,
80 Hexadecimal,
200 Octal,
10000000 Binary

버튼 클릭

p3

A screenshot of a Mac OS X desktop. At the top, there is a menu bar with Apple, Code, File, Edit, Selection, View, Go, Run, Terminal, Window, and Help. Below the menu bar is a toolbar with icons for back, forward, search, and file operations. The main window shows a Finder sidebar on the left with items like 'User-defined objects', 'practice_2.html', and 'practice_3.html'. The main pane displays the code for 'practice_3.html'.

```
<!DOCTYPE html>
<html>
<body>
<p id="demo"></p>
<script>
var family = new Array("Kim", "Lee", "Park", "Choi", "Yoon");
document.getElementById("demo").innerHTML = family;
document.write("slice method: " + family.slice(0, 2));
</script>
</body>
</html>
```

p4

A screenshot of a Mac OS X desktop. At the top, there is a menu bar with Apple, Safari, File, Edit, View, History, Bookmarks, Window, and Help. Below the menu bar is a toolbar with icons for back, forward, search, and file operations. The main window shows a Finder sidebar on the left with items like 'User-defined objects', 'practice_2.html', 'practice_3.html', and 'practice_4.html'. The main pane displays the code for 'practice_4.html'.

```
<!DOCTYPE html>
<html>
<body>
<p> Please input a number between 5 and 10: </p>
<input id="demo" type="text">
<button type="button" onclick="myFunction()"> Test input </button>
<p id="message"></p>
<script>
function myFunction() {
    var message, x;
    message = document.getElementById("message");
    message.innerHTML = "";
    x = document.getElementById("demo").value;

    try {
        if (x == "") throw "is empty";
        if (isNaN(x)) throw "is not a number";
        x = Number(x);
        if (x > 10) throw "is too high";
        if (x < 5) throw "is too low";
    }
    catch (err) {
        message.innerHTML = "input " + err;
    }

    finally {
        document.getElementById("demo").value = "";
    }
}
</script>
</body>
</html>
```

초기 화면

Please input a number between 5 and 10:

input is empty

```
<!DOCTYPE html>
<html>
<body>
<p> Please input a number between 5 and 10: </p>
<input id="demo" type="text">
<button type="button" onclick="myFunction()"> Test input </button>
<p id="message"></p>

<script>
function myFunction() {
    var message, x;
    message = document.getElementById("message");
    message.innerHTML = "";
    x = document.getElementById("demo").value;

    try {
        if (x == "") throw "is empty";
        if (isNaN(x)) throw "is not a number";
        x = Number(x);
        if (x > 10) throw "is too high";
        if (x < 5) throw "is too low";
    }
    catch (err) {
        message.innerHTML = "input " + err;
    }

    finally {
        document.getElementById("demo").value = "";
    }
}
</script>
</body>
</html>
```

아무것도 입력하지 않고 버튼 클릭

Please input a number between 5 and 10:

input is not a number

```
<!DOCTYPE html>
<html>
<body>
<p> Please input a number between 5 and 10: </p>
<input id="demo" type="text">
<button type="button" onclick="myFunction()"> Test input </button>
<p id="message"></p>

<script>
function myFunction() {
    var message, x;
    message = document.getElementById("message");
    message.innerHTML = "";
    x = document.getElementById("demo").value;

    try {
        if (x == "") throw "is empty";
        if (isNaN(x)) throw "is not a number";
        x = Number(x);
        if (x > 10) throw "is too high";
        if (x < 5) throw "is too low";
    }
    catch (err) {
        message.innerHTML = "input " + err;
    }

    finally {
        document.getElementById("demo").value = "";
    }
}
</script>
</body>
</html>
```

문자 입력

The screenshot shows a Mac desktop with a Safari browser window and a Sublime Text code editor. The browser window displays a simple form with a text input field and a button labeled "Test input". The message in the browser says: "Please input a number between 5 and 10:". The code editor shows the following code:

```
<!DOCTYPE html>
<html>
<body>
    <p> Please input a number between 5 and 10: </p>
    <input id="demo" type="text">
    <button type="button" onclick="myFunction()"> Test input </button>
    <p id="message"></p>
<script>
    function myFunction() {
        var message, x;
        message = document.getElementById("message");
        message.innerHTML = "";
        x = document.getElementById("demo").value;

        try {
            if (x == "") throw "is empty";
            if (isNaN(x)) throw "is not a number";
            x = Number(x);
            if (x > 10) throw "is too high";
            if (x < 5) throw "is too low";
        }
        catch (err) {
            message.innerHTML = "input " + err;
        }

        finally {
            document.getElementById("demo").value = "";
        }
    }
</script>
</body>
</html>
```

5보다 작은 수 입력

The screenshot shows a Mac desktop with a Safari browser window and a Sublime Text code editor. The browser window displays the same form as before, but the input field now contains the value "11". The message in the browser says: "input 11". The code editor shows the same code as the previous screenshot.

10보다 큰 수 입력

The screenshot shows a Mac desktop environment. On the left, a Safari browser window is open with the URL `http://127.0.0.1:5503/hw03/codes/practice_4.html`. The page content is:

```
Please input a number between 5 and 10:  
7  
Test input
```

On the right, a Sublime Text editor window is open with the file `practice_4.html`. The code is:

```
<!DOCTYPE html>
<html>
<body>


Please input a number between 5 and 10: </p>


<input id="demo" type="text">
<button type="button" onclick="myFunction()"> Test input </button>

<p id="message"></p>

<script>
function myFunction() {
    var message, x;
    message = document.getElementById("message");
    message.innerHTML = "";
    x = document.getElementById("demo").value;

    try {
        if (x == "") throw "is empty";
        if (isNaN(x)) throw "is not a number";
        x = Number(x);
        if (x > 10) throw "is too high";
        if (x < 5) throw "is too low";
    }
    catch (err) {
        message.innerHTML = "input " + err;
    }

    finally {
        document.getElementById("demo").value = "";
    }
}
</script>
</body>
</html>
```

The status bar at the bottom of the Sublime Text window indicates: Ln 35, Col 8 Spaces: 4 UTF-8 LF HTML Go Live.

5와 10 사이의 수 입력

p5

```
<!DOCTYPE html>
<html>
<head>
<script type="text/javascript">
window.onerror = function(msg, url, line) {
    alert("Message: " + msg);
    alert("url: " + url);
    alert("Line number: " + line);
}
</script>
</head>
<body>
<p> Click the following to see the result: </p>
<form>
<input type="button" value="Click Me" onclick="myFunc();"/>
</form>
</body>
</html>
```

초기 화면

Message: Script error.

Close

```
<!DOCTYPE html>
<html>
<head>
<script type="text/javascript">
window.onerror = function(msg, url, line) {
    alert("Message: " + msg);
    alert("url: " + url);
    alert("Line number: " + line);
}
</script>
</head>
<body>
<p> Click the following to see the result: </p>
<form>
<input type="button" value="Click Me" onclick="myFunc();"/>
</form>
</body>
</html>
```

에러 메시지 출력

```
<!DOCTYPE html>
<html>
<head>
<script type="text/javascript">
window.onerror = function(msg, url, line) {
    alert("Message: " + msg);
    alert("url: " + url);
    alert("Line number: " + line);
}
</script>
</head>
<body>
<p>Click the following to see the result:</p>
<form>
<input type="button" value="Click Me" onclick="myFunc();"/>
</form>
</body>
</html>
```

에러 url 출력

```
<!DOCTYPE html>
<html>
<head>
<script type="text/javascript">
window.onerror = function(msg, url, line) {
    alert("Message: " + msg);
    alert("url: " + url);
    alert("Line number: " + line);
}
</script>
</head>
<body>
<p>Click the following to see the result:</p>
<form>
<input type="button" value="Click Me" onclick="myFunc();"/>
</form>
</body>
</html>
```

에러 줄 번호 출력

p6

A screenshot of a Mac OS X desktop. On the left is a photo of a tabby cat sitting on a wooden floor. Below the photo is a caption: "Click the buttons below to handle animation" followed by two buttons: "Start" and "Stop". On the right is a code editor window titled "practice_6.html - codes". The code is a simple JavaScript animation script:

```
<!DOCTYPE html>
<html>
<head>
<title>JavaScript Animation</title>
<script type="text/javascript">
var imgObj = null;
var animate;

function init() {
    imgObj = document.getElementById('myImage'); imgObj.style.left = '0px';
}

function moveRight() {
    imgObj.style.left = parseInt(imgObj.style.left) + 10 + 'px';
    animate = setTimeout(moveRight, 20); // call moveRight in 20ms
}

function stop() {
    clearTimeout(animate); imgObj.style.left = '0px';
}
window.onload = init;

</script>
</head>

<body>
<form> 
<input type="button" value="Stop" onclick="stop();;" />
</form>
</body>
</html>
```

The status bar at the bottom shows "Ln 35, Col 8 Spaces: 4 UTF-8 LF HTML ⌂ Go Live ⌂ ⌂".

초기 화면

A screenshot of a Mac OS X desktop. On the left is a photo of a tabby cat sitting on a wooden floor. Below the photo is a caption: "Click the buttons below to handle animation" followed by two buttons: "Start" and "Stop". On the right is a code editor window titled "practice_6.html - codes". The code is identical to the one in the previous screenshot:

```
<!DOCTYPE html>
<html>
<head>
<title>JavaScript Animation</title>
<script type="text/javascript">
var imgObj = null;
var animate;

function init() {
    imgObj = document.getElementById('myImage'); imgObj.style.left = '0px';
}

function moveRight() {
    imgObj.style.left = parseInt(imgObj.style.left) + 10 + 'px';
    animate = setTimeout(moveRight, 20); // call moveRight in 20ms
}

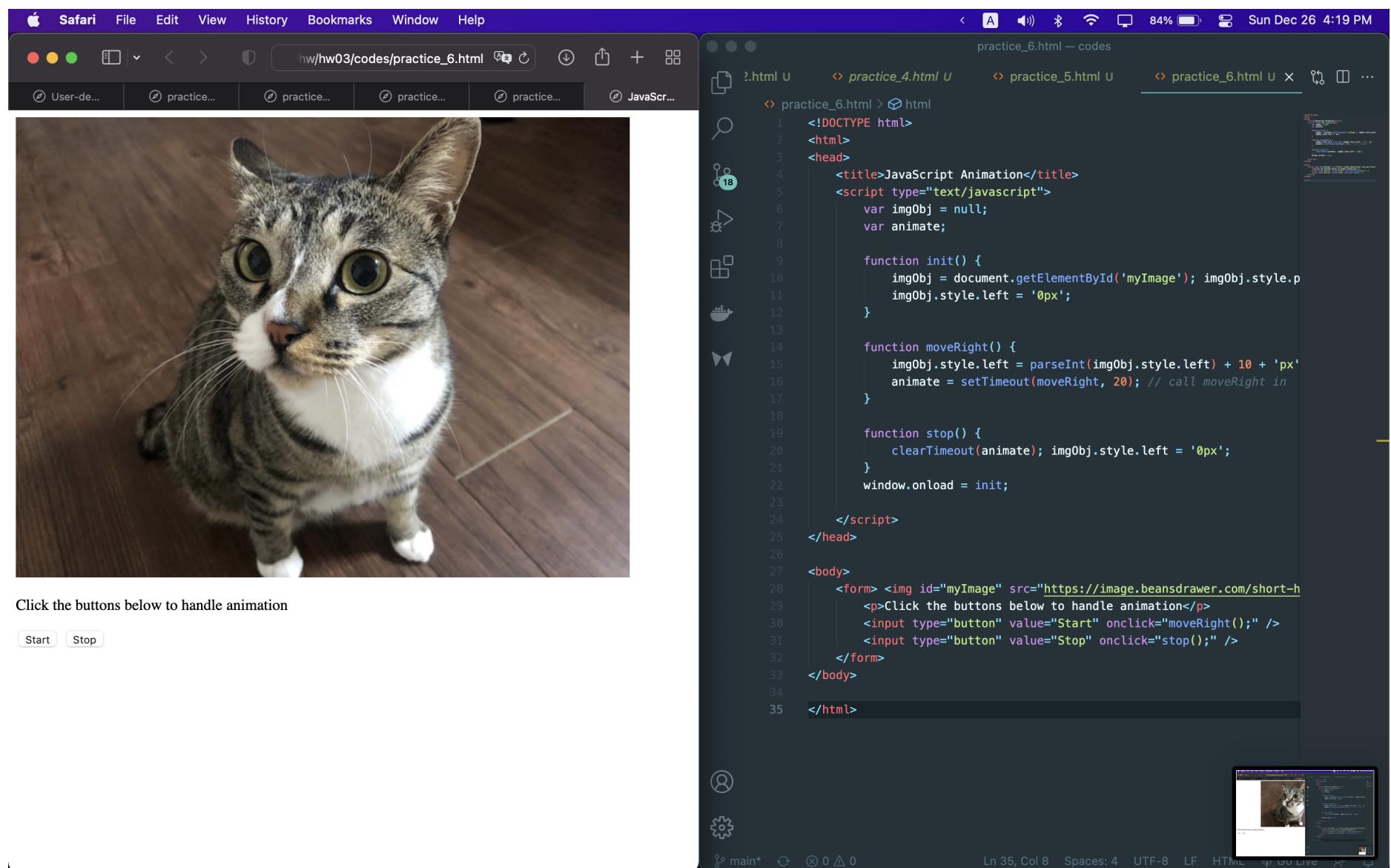
function stop() {
    clearTimeout(animate); imgObj.style.left = '0px';
}
window.onload = init;

</script>
</head>

<body>
<form> 
<input type="button" value="Stop" onclick="stop();;" />
</form>
</body>
</html>
```

The status bar at the bottom shows "Ln 35, Col 8 Spaces: 4 UTF-8 LF HTML ⌂ Go Live ⌂ ⌂". A small thumbnail of the cat image is visible in the bottom right corner of the screen.

start를 누르면 사진 오른쪽으로 이동



stop을 누르면 사진 처음 위치로 이동