

## Javascript Basic

방학특강 – 2024

# 학습 내용

1. 자바스크립트 기초
2. DOM
3. 이벤트와 이벤트 처리
4. UI 종합 실습



# **\_Visual Studio Code Extension**

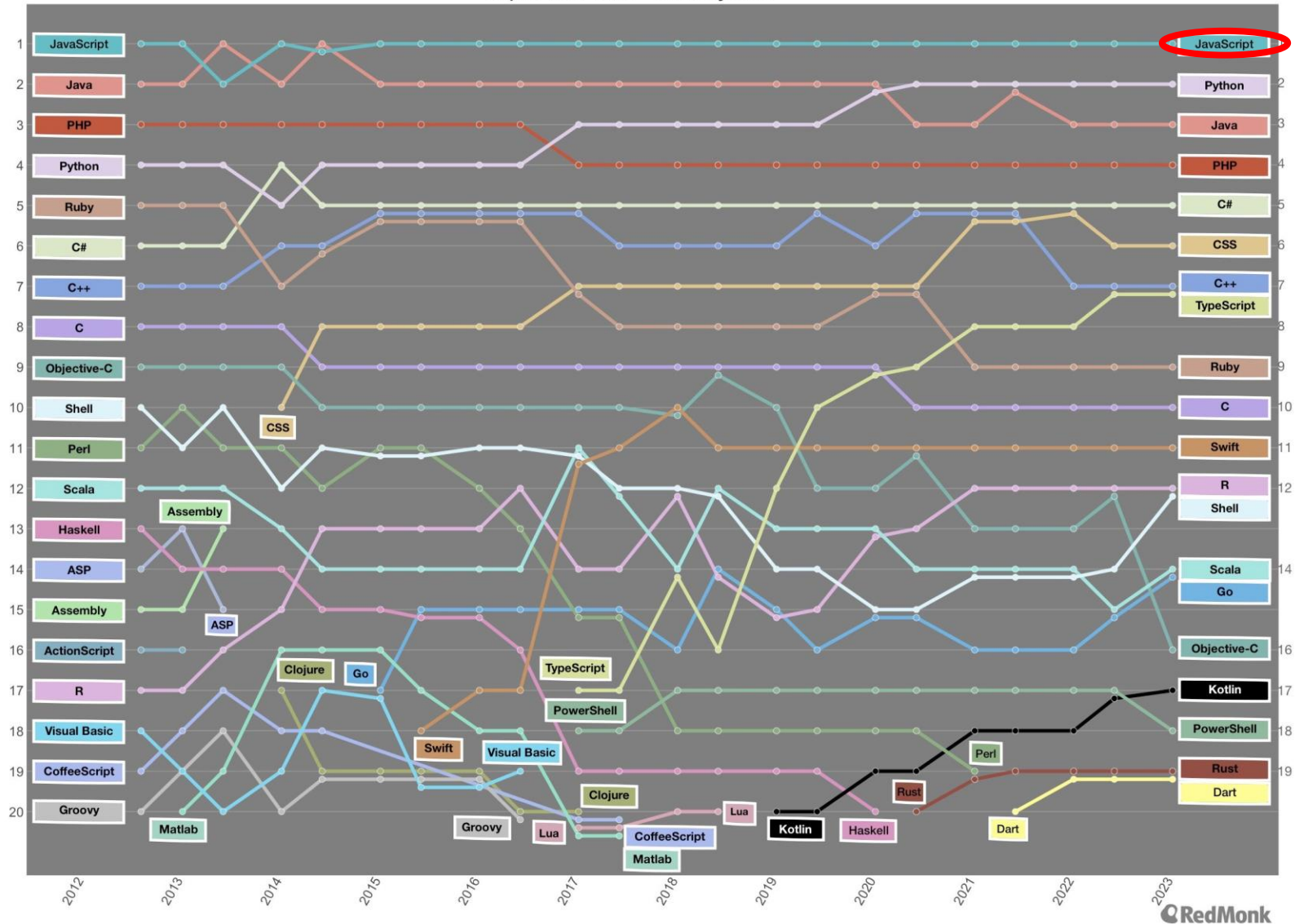
## Visual Studio Code Extension

- Korean Language Pack for Visual Studio Code
- Live Server
- Prettier
- Path Intellisense

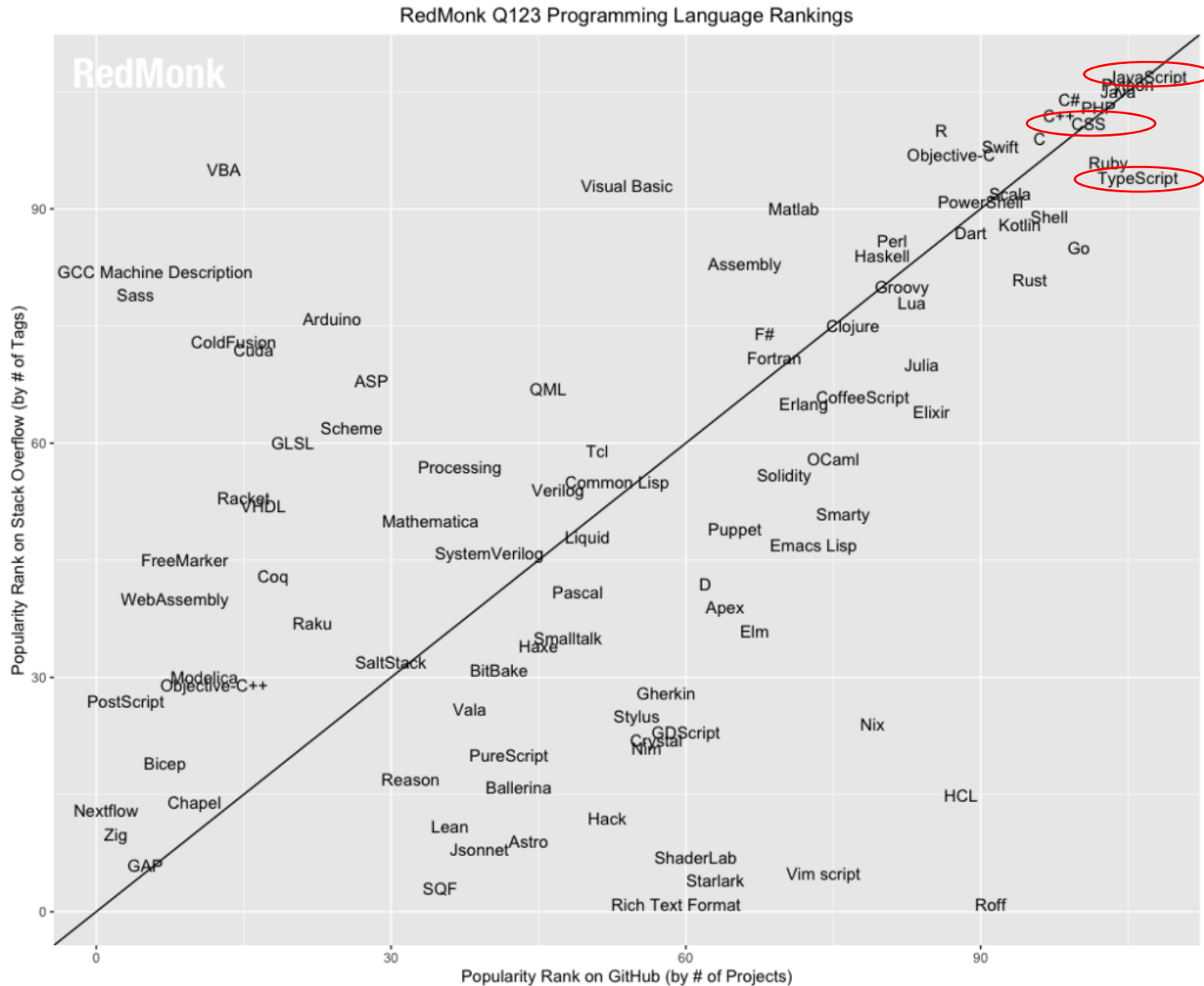
# \_Javascript에 대해

## RedMonk Language Rankings

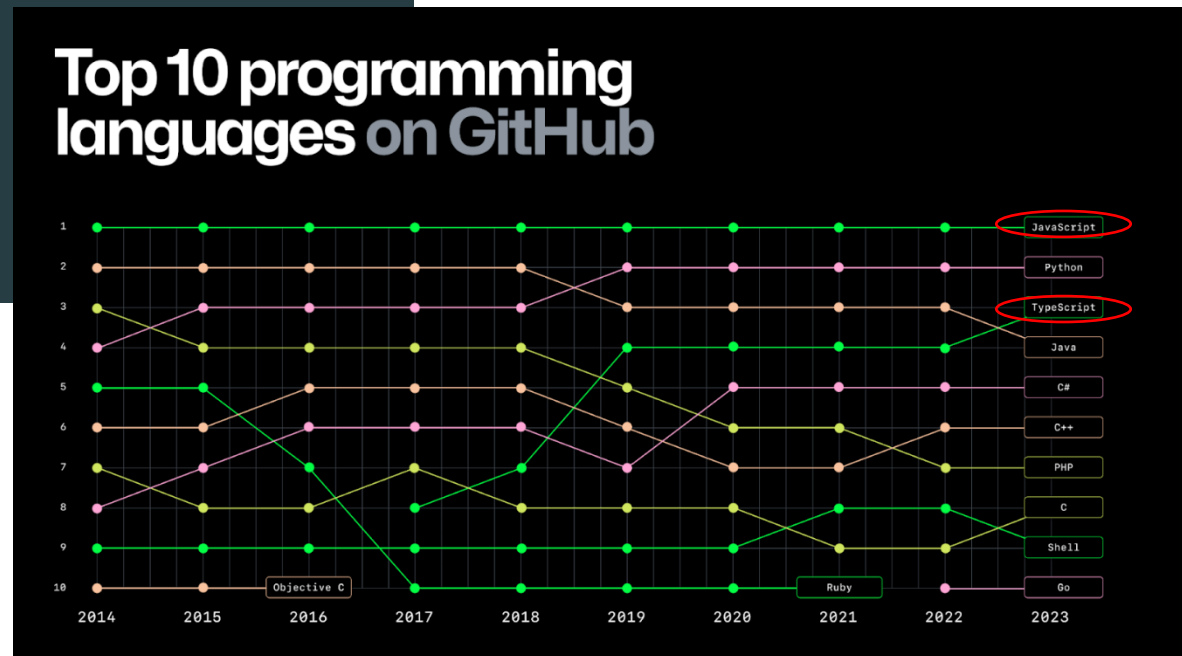
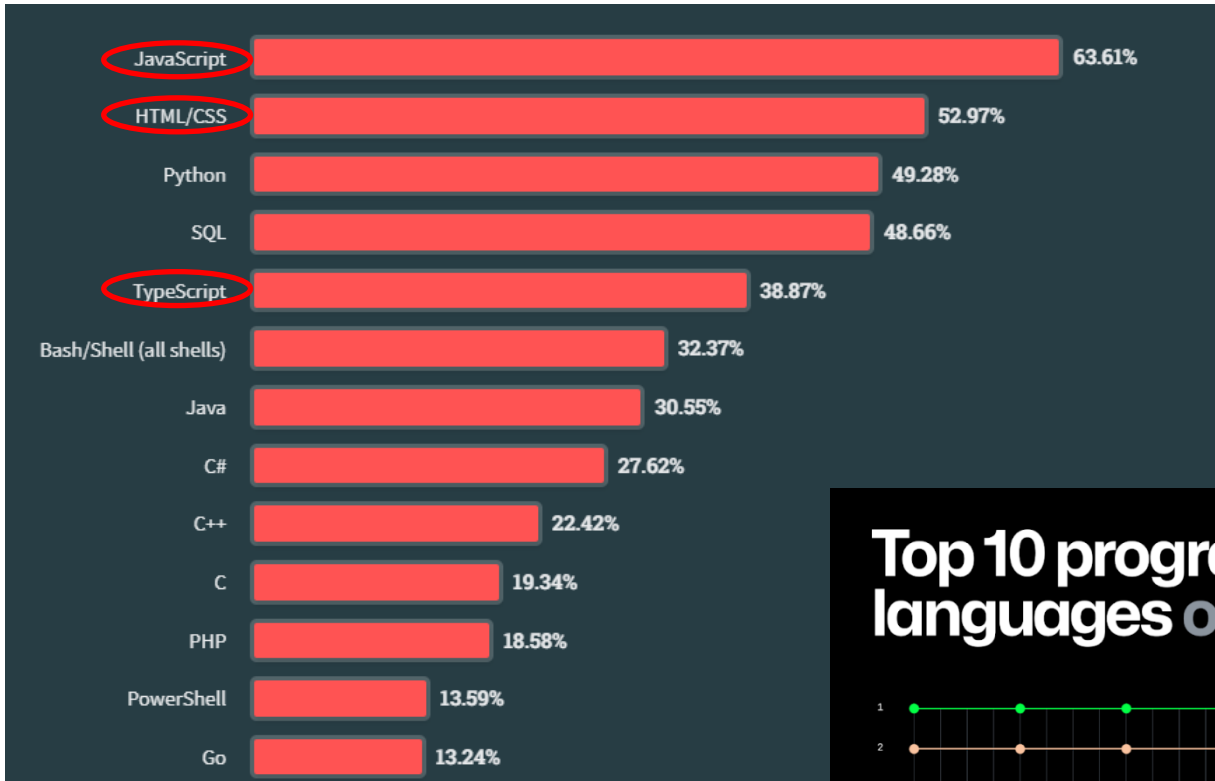
September 2012 - January 2023



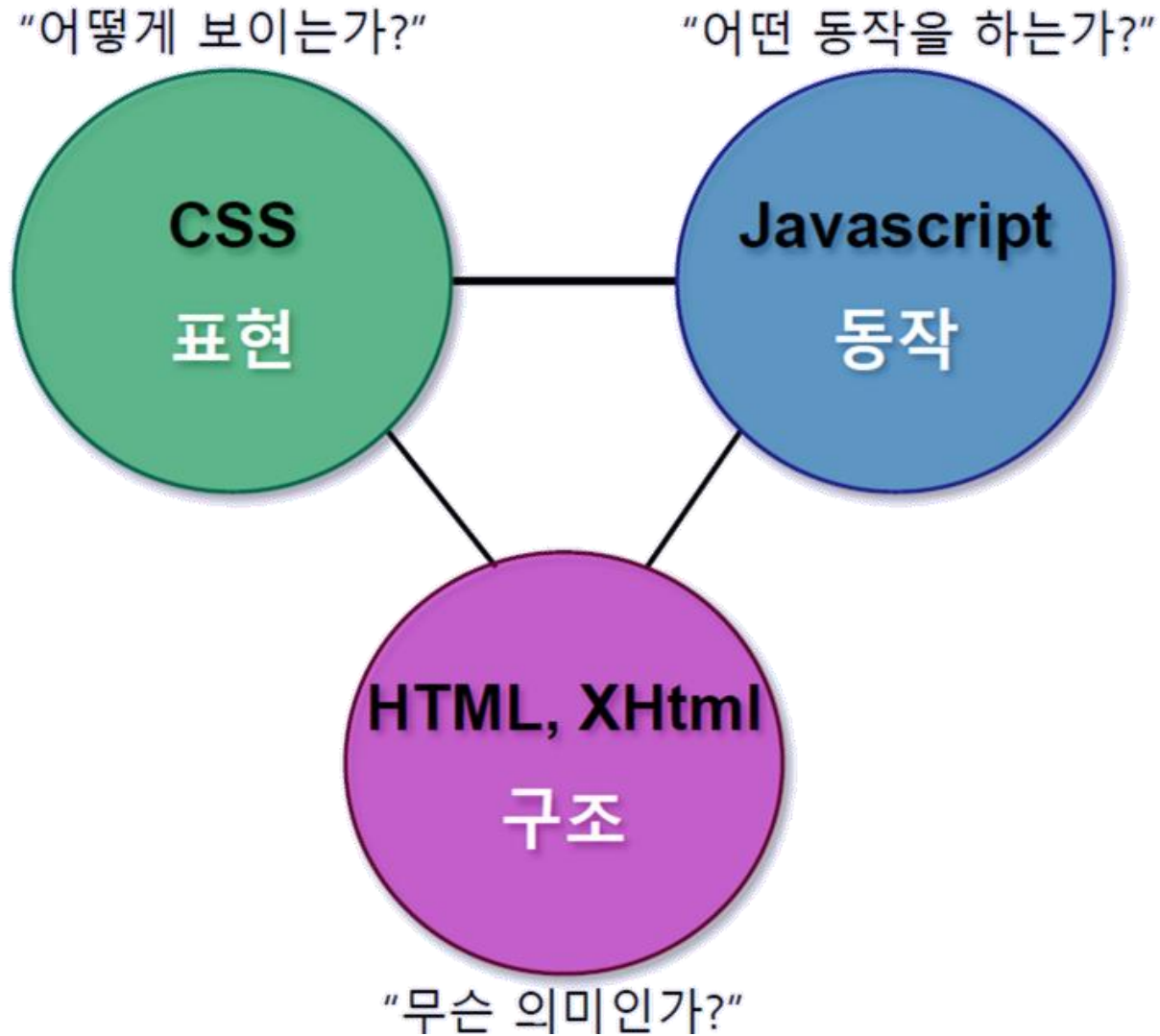
# \_Javascript에 대해



# \_Javascript에 대해



## \_자바스크립트의 역할



# 1\_기본 입력과 출력

1. 기본 출력 : `console.log("출력 내용");`

2. 기본 입력 : `prompt("메시지", 디폴트값) ;`

```
let num = 10;
```

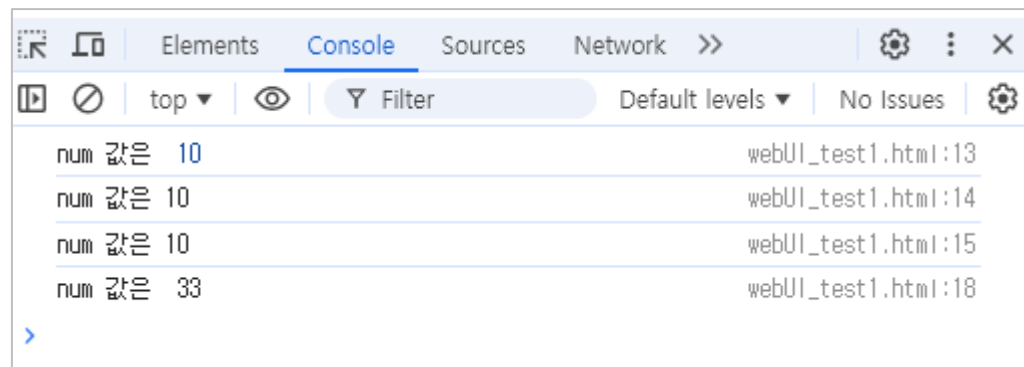
```
console.log("num 값은 ", num);
```

```
console.log("num 값은 " + num);
```

```
console.log(`num 값은 ${num}` );
```

```
num = prompt("새로운 값 입력", 0);
```

```
console.log("num 값은 ", num);
```





# 1\_함수

- 정의형식(선언문) : function [함수명] (가인자) { 함수 로직 }
- 호출형식(표현식) : [함수명](실인자);
- 리턴할 값이 있으면 return 문 뒤에 명시

```
let num = 10;
```

```
function plusNum(val1, val2) {  
  let num = val1 + val2;  
  return num;  
}
```

```
console.log("함수 호출 결과 값은 " + plusNum(num) + "입니다.");
```

# 1\_함수

두 수를 받아서 곱한 결과를 리턴하는 함수를 만드시오.

```
<script>
function plusNum(val1, val2) {
  let num = val1 + val2;
  return num;
}

let result = plusNum(20, 10);

console.log("함수 호출 결과 값은 " + result + "입니다.");
alert ("함수 호출 결과 값은 " + result + "입니다.");

// 곱하기 함수 구현

let result2 = multiNum(20, 10);
console.log("함수 호출 결과 값은 " + result2 + "입니다.");

</script>
```

# 1\_함수

두 수를 받아서 곱한 결과를 리턴하는 함수를 만드시오.

```
<script>
```

```
function plusNum(val1, val2) {  
  let num = val1 + val2;  
  return num;  
}
```

```
let result = plusNum(20, 10);
```

```
console.log("함수 호출 결과 값은 " + result + "입니다.");  
alert ("함수 호출 결과 값은 " + result + "입니다.");
```

```
function multiNum(val1, val2) {  
  let num = val1 * val2;  
  return num;  
}
```

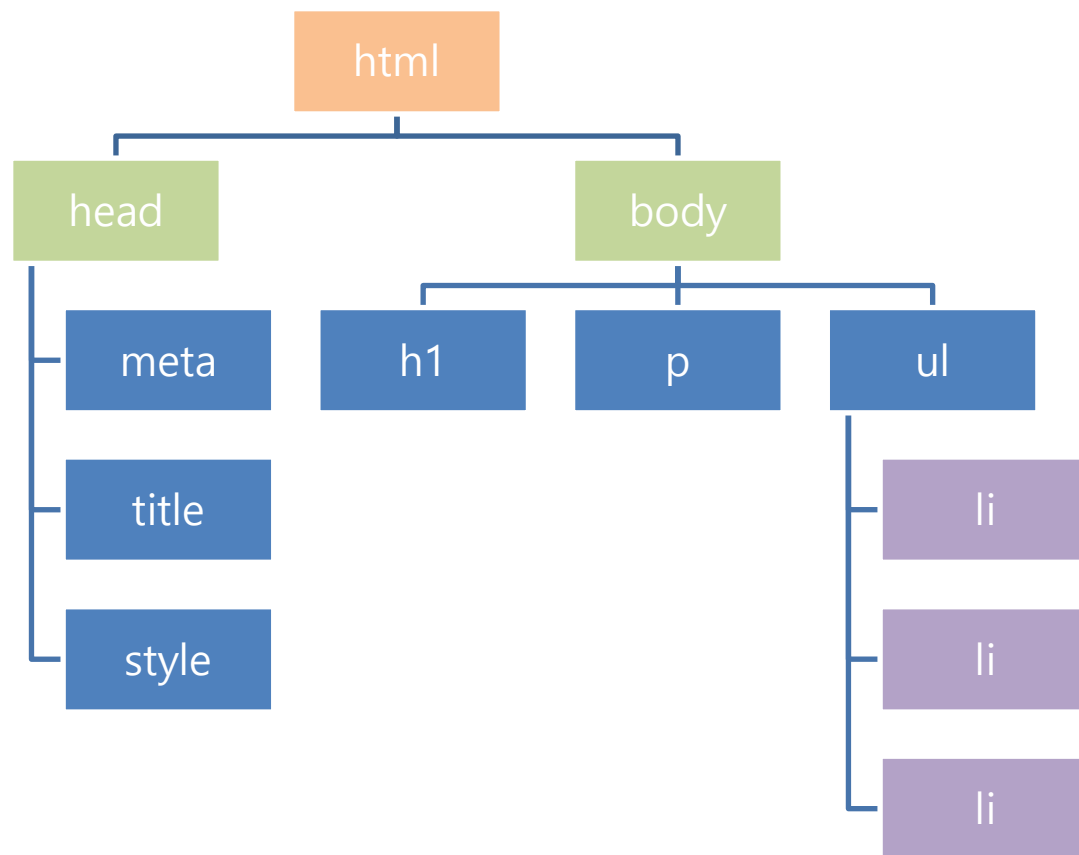
```
let result2 = multiNum(20, 10);  
console.log("함수 호출 결과 값은 " + result2 + "입니다.");
```

```
</script>
```

## 2\_DOM (Document Object Model)

- 문서 객체 모델 : document 객체(문서)와 관련된 객체(태그)의 집합을 나타냄
- HTML 태그를 자바스크립트에서 사용할 수 있는 객체로 만들.  
문서 객체를 조작한다는 말은 태그 (엘리먼트) 를 조작한다는 말

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="EUC-KR">
    <title>DOM</title>
    <style>
      body {
        color:white;
        background-color:black;
      }
    </style>
  </head>
  <body>
    <h1 title="want">원하는 것</h1>
    <p title="book">보고 싶은 책</p>
    <ul id="list">
      <li>멈추면 비로소 보이는 것들</li>
      <li>무지개 원리</li>
      <li>은교</li>
    </ul>
  </body>
</html>
```



## 2\_DOM : 문서객체의 선택

- 엘리먼트 선택 : 엘리먼트를 다루기 위해서는 DOM 객체를 찾는 것에서 부터 시작
- document.querySelector( [선택자] )  
document.querySelectorAll( [선택자] )  
document.getElementById( [선택할 Id ] )

<div class="game" id="myid" year="2022" value="100" > Hello </div>

①                      ②                      ③                      ④

	Selector	설 명
①	.game	클래스 값이 game인 모든 엘리먼트 ex) .game
②	#myid	아이디가 myid인 엘리먼트 ex) #myId
③	div[year="2022"]	어트리뷰트 year의 값이 '2022'인 모든 div 엘리먼트 ex) div[year="2022"] div[display=none]
④	.game[value="100"]	클래스가 'game'인 것 중 value의 값이 '100'인 엘리먼트 ex) .game[value="100"]

## 2\_DOM : 문서객체의 선택 (CSS)

```
<style>
  .sports {
    font-size: 20px;
    font-weight: bold;
    color: blue;
  }

  .rpg {
    font-size: 15px;
    font-weight: bold;
    color: crimson;
    border: 3px solid rgb(7, 72, 194);
  }
</style>
```

```
<div>
```

```
<div id="game1" class="sports" year="2022">NBA 2022</div>
<div id="game2" class="sports" year="2022">FIFA 2022</div>
<div id="game3" class="sports" year="2022">NFL 2022</div>
<br /> <br />
```

```
<div id="game7" year="2022">Marvel Fighter</div>
<br /> <br />
```

```
<div id="game4" class="rpg" year="2008">WoW</div>
<div id="game5" class="rpg" year="1990">Jelda</div>
<div id="game6" class="rpg" year="1998">diablo</div>
```

```
</div>
```

NBA 2022  
FIFA 2022  
NFL 2022

Marvel Fighter

WoW

Jelda

diablo

## 2\_DOM : 문서 객체(엘리먼트)의 조작

`<div class="game" style="width:100px;" year="2022" value="100" > Hello </div>`

④                      ②                      ③                      ⑤                      ①

	대상	표준 Javascript
①	Text (내용)	<code>obj1.innerHTML = "Hello" ;</code> <code>obj1.textContent = "Hello" ;</code>
②	Style (표현)	<code>jObj.style.color = "red";</code> <code>jObj.style.width = "20px";</code>
③	일반 속성 (태그 기능)	<code>srcVal = jObj.getAttribute('src');</code> <code>jObj.setAttribute('src', '100.png');</code>
④	Class (류움 지정)	<code>JObj.classList.add("rpg");</code> <code>JObj.classList.remove("rpg");</code>
⑤	Value (UI 입력값)	<code>let value = JObj.value ;</code> <code>JObj.value = 100 ;</code>

## 2\_DOM : 문서 객체(엘리먼트)의 조작

<div>

```
<div id="game1" class="sports" year="2022">NBA 2022</div>
<div id="game2" class="sports" year="2022">FIFA 2022</div>
<div id="game3" class="sports" year="2022">NFL 2022</div>
<br /> <br />
```

```
<div id="game7" year="2022">Marvel Fighter</div>
<br /> <br />
```

```
<div id="game4" class="rpg" year="2008">WoW</div>
<div id="game5" class="rpg" year="1990">Jelda</div>
<div id="game6" class="rpg" year="1998">diablo</div>
```

</div>

NBA 2022  
FIFA 2022  
NFL 2022

Marvel Fighter

WoW

Jelda

diablo

```
const game1El = document.querySelector("#game1");
game1El.innerHTML = "NBA 2000";
```

```
let yearAttr = game1El.getAttribute("year");
console.log('game1El year attribute = ', yearAttr);
game1El.setAttribute("year", "2000");
yearAttr = game1El.getAttribute("year");
console.log('game1El year attribute = ', yearAttr);
```

```
const game2El = document.querySelector("#game2");
game2El.style.border = "1px solid blue";
game2El.style.padding = "24px";
game2El.style.width = "48px";
```

```
const game7El = document.querySelector("#game7");
game7El.classList.add("rpg");
```

```
const rpgameEls = document.querySelectorAll(".rpg");
rpgameEls.forEach(function (el) {
    console.log(el.innerHTML);
});
```



## 2\_DOM : 문서 객체(엘리먼트)의 조작

1. 객체 내부 (태그 사이) 조작 (속성) : [요소객체].innerHTML    [요소객체].textContent

2. 스타일 조작 (style 속성) : [요소객체].style.[스타일명]

3. 일반 속성 조작 (함수)

- `setAttribute([속성이름], [속성 값]), getAttribute([속성이름])`
- 웹표준 속성은 '[요소객체].[속성명]' 으로 접근 가능    ex) `image.src`

4. 클래스 조작 (함수) : 자바스크립트에서는 클래스명으로 요소를 묶어서 처리하는 일이 많으므로 매우 중요

- 클래스 추가 : `[요소객체].classList.add([추가클래스]);`
- 클래스 삭제 : `[요소객체].classList.remove([삭제클래스]);`

5. UI 값 조작 (속성) : UI에 입력된 값을 대상

- 값 얻기 : `[요소객체].value`
- 값 수정 : `[요소객체].value = [새값];`

## 3\_이벤트와 이벤트 핸들러

### 1. 이벤트 핸들러 (이벤트 발생시 호출되는 함수) 등록

1) 정의 : function을 정의한다. → `function myFunction (..) {…}`

2) 이벤트 핸들러 등록

- 방법1 : 태그 속성에서 등록

`<div onclick="myFunction()"> </div>`

- 방법2 : 자바스크립트에서 등록

엘리먼트객체.addEventListner("click", myFunction);

### 2. 이벤트 종류

Mouse Events	Keyboard Events	Form Events	Document/Window Events
click dblclick mouseenter mouseleave mousedown mouseup hover	keypress keydown keyup	submit change focus blur	load resize scroll unload

### 3\_이벤트와 이벤트 핸들러

```
<div id="divbox" style="width:500px;height:200px; background: yellowgreen;"> </div>
<div id="mouseXYText" style="width:500px;height:50px;border: 1px solid black; " > </div>
<br><br><hr>
입력창 <input id="inputbox" type="text" style="width:400px;height:30px;" />
<div id="keyCodeText" style="width:500px;height:30px;border: 1px solid black; " > </div>

<script>
    document.querySelector("#divbox").addEventListener("click", function (event) {
        document.querySelector("#mouseXYText").innerHTML =
            "mouse >>>> " + event.offsetX + " , " + event.offsetY;

        // alert("divbox를 클릭했습니다.");
    });
    document.querySelector("#inputbox").addEventListener("keyup", function (event) {
        document.querySelector("#keyCodeText").innerHTML =
            "keyCode >>>> " + event.keyCode;

        // alert("inputbox에서 키보드를 눌렀습니다.");
    });
</script>
```



## 4\_UI 종합 실습

### UI Elements Example

#### Input Field and Button

Show Input Value

#### Checkbox and Radio Buttons

☐ Checkbox  
☐ Option 1  
☐ Option 2

#### Dropdown Menu

Option 1

Show Selection Values

#### Toggle Switch

☐ Toggle this switch element

Show Toggle

#### Notification Message

Show Notification

This is a notification message

```
<button class="btn btn-primary" onclick="showInputValue()">  
Show Input Value</button>
```

```
<button class="btn btn-primary mt-2"  
onclick="showSelectionValues()">Show Selection Values</button>
```

```
<button class="btn btn-primary" onclick="showToggle()">  
Show Toggle</button>
```

```
<button class="btn btn-primary" onclick="showNotification()">  
Show Notification</button>
```

## 4\_UI 종합 실습

```
const textArea = document.querySelector('#notification');
const inputField = document.querySelector('#inputField');
let textUIValue = "";
let checkUIValue = "";
let toggleUIValue = "";
```

```
function showInputValue() {
    textUIValue = 'Input Value: ' + inputField.value;
    alert(textUIValue);
    refreshText();
}
```

```
function showToggle() {
    const checkbox = document.querySelector('#toggleSwitch').checked;
    toggleUIValue = `Toggle: ${checkbox}`;
    alert(toggleUIValue);
    refreshText();
}
```

```
function refreshText() {
    textArea.innerHTML = textUIValue + '<br>' + checkUIValue + '<br>' + toggleUIValue;
}
```

## 4\_UI 종합 실습

```
function showSelectionValues() {
    const checkbox = document.querySelector('#checkbox').checked;
    const dropdown = document.querySelector('#dropdown').value;
    const radios = document.getElementsByName('radioGroup');

    let selectedRadio = null;
    for (const radio of radios) {
        if (radio.checked) {
            selectedRadio = radio.value;
            break;
        }
    }
    checkUIValue = `Checkbox: ${checkbox}\nDropdown: ${dropdown}\nSelected Radio: ${selectedRadio}`;
    alert(checkUIValue);
    refreshText();
}

function showNotification() {
    const notification = document.querySelector('#notification');
    if(notification.style.display !== 'none'){
        notification.style.display = 'none';
    }else{
        notification.style.display = 'block';
    }
}
```

---

## 실습 예제

---

## 실습-1

**실습 문제** : 아래와 같이 UI를 만들고 아이디/암호를 입력 받아 처리하는 프로그램 완성.

- 아이디/암호는 user/user1234과 맞으면 로그인 되었다는 메시지 박스가 뜨면 됨  
(틀리면 틀리다는 메시지 박스를 띄운 후 '암호'입력란을 비움)

The image shows a simple login form. On the left, there are two labels: '아이디' (ID) and '비밀번호' (Password). To the right of '아이디' is a text input field. To the right of '비밀번호' is a password input field. To the right of both input fields is a button labeled '로그인' (Login).

```
<div style="width: 550px; height: 350px;">
  <div style="position: absolute; top: 40px; left: 10px;">아이디</div>
  <div style="position: absolute; top: 80px; left: 10px;">비밀번호</div>
  <div style="position: absolute; top: 40px; left: 95px;">
    <input type="text" style="width:215px; height: 25px;" id="user_id" value="">
  </div>
  <div style="position: absolute; top: 80px; left: 95px;">
    <input type="password" style="width:215px; height: 25px;" id="user_password" value="">
  </div>
  <div style="position: absolute; top: 40px; left: 330px;">
    <input type="button" style="width: 100px;height:75px;" id="btnLogin" value="로그인"></input>
  </div>
</div>
```



## 실습-2

### 최종 실습 문제 : 성적 처리 프로그램

1. 화면에서 '입력' 버튼을 누르면 이름, 국어, 영어, 수학 성적 입력 받음
2. 성적이 입력되면 총점과 평균이 자동으로 계산 됨
3. 총점과 평균이 계산되면 석차도 계산 됨
4. 화면에서 '전체 출력' 버튼을 누르면 입력된 모든 학생들이 성적이 출력  
(브라우저 화면에 출력 – 콘솔 아님)
5. 화면에서 '성적 집계' 버튼을 누르면 전체 학생들의 총점에 대한 합계와 평균,  
과목별 총점과 평균이 출력

## 실습-2

```
<body>
  <div>
    <input type="button" id="inputBtn" style="width:100px" value="입력"/>
    <input type="button" id="printAllBtn" style="width:100px" value="전체출력"/>
    <input type="button" id="scoreBtn" style="width:100px" value="성적집계"/>
    <br/> <br/>
    <input type="text" id="inputCount" style="width:300px" />
  </div>
  <br /> <br />
  <div id="resultText"> </div>
  <script>

  </script>
</body>
```

