

콜백함수 & 배열

```
/* 실습5 */
var coding = ["HTML", "CSS", "javaScrip", "jQuery", "JAVA", "JSP", "Oracle", "React"];
for(var i=0; i<coding.length; i++){
    document.write(coding[i]+"<br>");
}
```

배열 for 문 돌려 i 초기값 부여하고 i보다 배열갯수만큼 i 증감 하여 도큐먼트 출력

콜백 함수(callback function)의 정의

- *콜백 함수(callback function)**란,

➡ 다른 함수의 인자로 전달되어, 특정 시점이나 조건이 만족될 때 실행되는 함수를 말한다.

즉, 내가 직접 호출하지 않고, "나중에 실행해!"라고 맡겨두는 함수이다.

◆ 기본 예시

```
function greeting(name) {
    console.log("안녕하세요, " + name);
}

function processUserInput(callback) {
    var name = "김현수";
    callback(name); // 여기서 callback 함수 실행
}
```

```
processUserInput(greeting);
```

- `greeting` → 콜백 함수 (나중에 실행될 함수)
- `processUserInput` → 콜백 함수를 받아서 실행하는 함수

👉 실행 결과:

안녕하세요, 김현수

◆ 콜백 함수가 중요한 이유

1. 비동기 처리 (프론트/백엔드 공통)

- 서버 응답, 파일 읽기, 타이머 같은 작업은 즉시 끝나지 않음.
- 콜백을 등록해두면 작업이 끝났을 때 자동 실행됨.

2. 재사용성 / 유연성

- 함수 동작 중 "어떻게 처리할지"를 바꾸고 싶을 때 콜백을 넘겨주면 됨.

```
<script>
  /* 1. 호출한 곳에 결괏값을 던져줌 */
  var num = 0; // 2. 10을 받음
  function plusNum(num) {
    return num + 10; //3. 10+ 10 해서 다시 결괏값 20 던짐 리턴이없으면 호출되지
    않음
  }
  // alert(plusNum(10)); // 1. 10을 던져서

  /* 2. return false; = 여기까지 실행 */
  var int = 10;
  var sum = 0;
  var mul = 0;
  function cal(int1) {
```

```

    sum = int1 + int;
    mul = int1 * int;
    return mul;
    return sum;
}
// 호출
document.write(cal(100));
</script>

```

◆ 실무 활용 예시

1) 프론트엔드 (브라우저)

```

document.getElementById("btn").addEventListener("click", function() {
    alert("버튼 클릭됨!");
});

```

`function(){...}` 이 콜백. 버튼 클릭 이벤트가 발생할 때 실행.

2) 백엔드 (Node.js)

```

const fs = require("fs");

fs.readFile("test.txt", "utf8", function(err, data) {
    if (err) {
        console.error("파일 읽기 실패", err);
        return;
    }
    console.log("파일 내용:", data);
});

```

`function(err, data){...}` 가 콜백.

파일 읽기가 끝난 뒤(성공/실패 여부에 따라) 실행됨.

◆ 정리

- 콜백 함수 = 다른 함수에 전달되어, 나중에 실행되는 함수
- 프론트엔드 → 이벤트 처리, 애니메이션, Ajax 통신
- 백엔드 → DB 쿼리, 파일 읽기/쓰기, 네트워크 요청

자바스크립트에서 `return` 은 *****함수의 실행 결과를 호출한 곳으로 되돌려주는 것*****을 말합니다.

◆ 1. 기본 개념

```
function add(a, b) {  
  return a + b; // 함수 실행 결과를 돌려줌  
}  
  
var result = add(3, 5);  
console.log(result); // 8
```

- `return a + b;` → `add(3, 5)` 를 호출한 곳에 합산한 `8` 을 돌려준다.
- `return` 이 없으면, 함수는 기본적으로 `undefined` 를 반환한다.

◆ 2. `return` 의 특징

1. 함수 종료

`return` 을 만나면, 그 시점에서 함수 실행이 멈추고 밖으로 결과를 내보낸다.

```
function checkAge(age) {  
  if (age < 20) {  
    return "미성년자"; // 여기서 함수 종료  
  }  
}
```

```
    return "성인";  
}  
console.log(checkAge(15)); // "미성년자"
```

2. 값을 돌려줌

계산 결과, 문자열, 객체, 배열, 심지어 또 다른 함수도 리턴할 수 있음.

```
function getUser() {  
    return { name: "현수", age: 25 };  
}  
console.log(getUser().name); //유저 함수에 이름만 뽑음 = 현수
```

3. 리턴 값 활용 가능

리턴된 값을 변수에 담거나, 다른 함수의 인자로 전달 가능.

```
function square(x) {  
    return x * x;  
}  
console.log(square(4) + square(3)); // 16 + 9 = 25  
// clg 에 square를 두번 호출하고 각 인자값 변경하여 + 하면 결과는  
// 4*4 + 3*3 = 25 된다.
```

◆ 3. return 없는 경우

```
function hello() {  
    console.log("안녕");  
}  
var result = hello();  
console.log(result); // undefined - boolean (false)
```

👉 return 이 없으면 자동으로 undefined 반환.



이전

다음

```
<script>
  let num = 1;
  function gallery(imgNum) {
    // alert(imgNum);
    if (imgNum) { // 다음버튼
      if (num == 7) {
```

```

        return;
    }
    num++;
} else { // 이전버튼
    if (num == 1) {
        return
    }
    num--;
}
// console.log(num);

// let phone = document.querySelector("#phone").src = "./images/img"
+ num + ".jpg";
let phone2 = document.querySelector("#phone").setAttribute("src", "./i
mages/img" + num + ".jpg");
console.log(phone2);
}
</script>
<div id="galleryzone">
    <p>
        
    </p>
    <p>
        <button onclick="gallery(0);">이전</button>
        <button onclick="gallery(1);">다음</button>
    </p>
</div>

```

배열객체(Array) : 여러개의 데이터 저장

◆ 배열(Array)의 정의

➡ 배열은 여러 개의 값을 하나의 변수에 순서대로 저장할 수 있는 자료구조이다.

- 값들은 **인덱스(index)** 로 구분 (0부터 시작)
- 숫자, 문자열, 객체, 함수 등 **모든 타입** 저장 가능
- 객체 = (참조 변수형)

◆ 배열 선언 방법

1) 리터럴 방식 (가장 많이 사용)

```
var fruits = ["국어", "영어", "수학"];
```

2) `new Array()` 생성자 방식

```
var numbers = new Array(10, 20, 30);
```

◆ 배열 요소 접근

```
var bb = ["국어", "영어", "수학"];
```

```
console.log(bb[0]); // "국어"
```

```
console.log(bb[1]); // "영어"
```

```
console.log(bb[2]); // "수학"
```

👉 인덱스(index)로 접근 (0부터 시작).

◆ 배열 주요 특징

1. 길이 속성 (`length`)

```
console.log(bb.length); // 3
```


2. 여러 타입 혼합 가능

```
var mixed = ["문자", 123, true, {name:"현수"}, function(){return "hi"}];
```

3. 동적으로 크기 변경 가능

```
var cc = [];  
cc.push("A");  
cc.push("B");  
console.log(cc); // ["A", "B"]
```

◆ 배열 주요 메서드 예시

- `push()` : 끝에 추가
- `pop()` : 끝에서 제거
- `shift()` : 앞에서 제거
- `unshift()` : 앞에 추가
- `forEach()` : 배열 반복
- `map()` : 배열 요소 변환
- `filter()` : 조건에 맞는 요소만 추출
- `reverse()` : 배열 뒤집기
- `sort()` : 정렬

```
var numbers = [1, 2, 3, 4, 5];  
numbers.reverse(); // [5,4,3,2,1]
```

✓ 정리

자바스크립트 배열은 여러 데이터를 순서대로 담을 수 있는 객체 타입 자료구조이다.

- 인덱스(순서)는 0부터 시작
- 다양한 메서드로 데이터 조작 가능

김현수

배열의 길이:5

=====

배열의 길이:3개 중 과목은 영어

=====

배열의 길이:5

=====

```
<!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>배열객체(Array) : 같은 변수명에 여러개의 데이터 저장</title>
  <script>
    /*1. 배열선언방법 */
    //1-1. 배열선언
    var aa = new Array(); //참조변수 aa 에 대입
    //배열 데이터 할당 (메모리 번짓수 0 부터 index 시작)
    aa[0] = "가";
    aa[1] = 3;
    aa[2] = "김현수";
    aa[3] = "대한";
    aa[4] = "민국";
```

```

// console.log(aa);

//1-2. 배열선언 + 배열에 데이터 할당
var bb = new Array("국어","영어","수학"); //생성과 동시에 데이터 할당
// console.log(bb);

//1-3. 배열선언 + 생성 + 배열에 데이터 할당;

var cc = [1,2,3,4,5];
// console.log(cc);

/*2. 배열데이터 호출 */
document.write(aa[2]+"<br>");
document.write("배열의 길이:" + aa.length + "<br>");
document.write("=====<br>");
//bb에서
document.write("배열의 길이:" + bb.length + "개 중 과목은 " + bb[1] + "<br>");
document.write("=====<br>");
//cc에서
document.write("배열의 길이:" + cc.length + "<br>");
document.write("=====<br>");

</script>
</head>

<body>

</body>

</html>

```

1. 배열 실습

HTML
CSS
javaScript

```
/*실습1. 웹퍼블리싱 과목 3개(HTML,CSS,javaScript)를 배열로 정의 | 배열명=subject  
(for문)(브라우저)과목 3개 모두 출력*/  
  
var subject = ['HTML', 'CSS', 'javaScript'];  
  
for (let i = 0; i < subject.length; i++) {  
    document.write(subject[i] + "<br>");  
};
```

2. 날짜 객체 생성 (new Date)



GMT(Greenwich Mean Time)는 **영국의 런던 그리니치 천문대를 기준으로 하는 세계 표준 시간**으로

현재는 협정 세계시(UTC)가 공식적으로 사용되고 있다.

(GMT+09:00 그리니치 시계랑 한국과 9시간 빠르다는 의미)

날짜 객체 생성 = `new Date;`

1. 날짜 객체 생성 Date | `today = 참조 변수명 => 콘솔 출력 ;`

```
var today = new Date;  
console.log(today);
```

Mon Sep 15 2025 11:49:07 GMT+0900 (한국 표준시)

```
<style>
  strong {
    color:red;
  }
</style>
<script>

  /* 1. 날짜 객체 생성 Date | today = 참조 변수명 => 콘솔 출력 ; */
  var today = new Date;
  console.log(today);

  var thisYear = today.getFullYear(); //년
  var thisMonth = today.getMonth(); //월

  document.write("해당년은: <strong>" + thisYear + "</strong>년 입니다.<br>")
  document.write("해당 월은: <strong>" + thisMonth + "</strong>월 입니다.<br>")

</script>
```

해당년은: **2025**년 입니다.
해당 월은: **8**월 입니다.

왜 8월로 찍히는가??? 지금은 9월인데.

```
<script>
  var thisMonth = today.getMonth()+1; //월
</script>
```

return type 이 index로 시작하며 0부터 11이라서 (+1로 따로 지정해줘야한다.)

해당년은: **2025**년 입니다.

해당 월은: **9**월 입니다.

해당 일은: **15**월 입니다.

해당 요일은: **1**일 입니다.

```
<style>
  strong {
    color:red;
  }
</style>
<script>
```

```
/* 1. 날짜 객체 생성 Date | today = 참조 변수명 => 콘솔 출력 ; */
var today = new Date;
console.log(today);
```

```
var thisYear = today.getFullYear(); //년
var thisMonth = today.getMonth()+1; //월
var thisDate = today.getDate(); //날짜
var thisDay = today.getDay(); //일 일요일(0) ~ 토요일(6)
```

```

    document.write("해당년은: <strong>" + thisYear + "</strong>년 입니다.<br>");
    document.write("해당 월은: <strong>" + thisMonth + "</strong>월 입니다.<br>");
    document.write("해당 일은: <strong>" + thisDate + "</strong>월 입니다.<br>");
    document.write("해당 요일은: <strong>" + thisDay + "</strong>월 입니다.<br>");

</script>

```

스위치 케이스문을 써서 해당 요일나오게 하는 방법은??

```

switch (thisDay) {
    case 0:
        dayName = "일요일";
        break;
    case 1:
        dayName = "월요일";
        break;
    case 2:
        dayName = "화요일";
        break;
    case 3:
        dayName = "수요일";
        break;
    case 4:
        dayName = "목요일";
        break;
    case 5:
        dayName = "금요일";
        break;
}

```



```

case 6:
    dayName = "토요일";
    break;
}
document.write("해당 요일은: <strong>" + dayName + "</strong> 입니다.<br>");

```

완성된 코드 모습

해당 년은: **2025**년 입니다.
 해당 월은: **9**월 입니다.
 해당 일은: **15**월 입니다.
 해당 요일은: **1**일 입니다.
 해당 요일은: **월요일** 입니다.
 해당 시는: **12** 입니다.
 해당 분은: **35** 입니다.
 해당 초는: **52** 입니다.
 해당 밀리세컨: **944** 입니다.
 크리스마스 이브까지 99일 남았습니다!

```

<style>
    strong {
        color: red;
    }
</style>
<script>

```

```

/* 1. 날짜 객체 생성 Date | today = 참조 변수명 => 콘솔 출력 ; */
var today = new Date;
//console.log(today);
var thisYear = today.getFullYear(); //년
var thisMonth = today.getMonth() + 1; //월
var thisDate = today.getDate(); //날짜
var thisDay = today.getDay(); //일 일요일(0) ~ 토요일(6)
var thisHour = today.getHours(); // 시
var thisMin = today.getMinutes(); // 분
var thisSec = today.getSeconds(); // 초
var thisMilliSec = today.getMilliseconds(); // 밀리세컨

```

```

/* 스위치 케이스문을 써서 해당 요일을 나오게해야한다 */
switch (thisDay) {
  case 0:
    dayName = "일요일";
    break;
  case 1:
    dayName = "월요일";
    break;
  case 2:
    dayName = "화요일";
    break;
  case 3:
    dayName = "수요일";
    break;
  case 4:
    dayName = "목요일";
    break;
  case 5:
    dayName = "금요일";
    break;
  case 6:
    dayName = "토요일";

```

```

        break;
    }

    document.write("해당 년은: <strong>" + thisYear + "</strong>년 입니다.<br>");
    document.write("해당 월은: <strong>" + thisMonth + "</strong>월 입니다.<br>");
    document.write("해당 일은: <strong>" + thisDate + "</strong>월 입니다.<br>");
    document.write("해당 요일은: <strong>" + thisDay + "</strong>일 입니다.<br>");
    document.write("해당 요일은: <strong>" + dayName + "</strong> 입니다.<br>");
    document.write("해당 시는: <strong>" + thisHour + "</strong> 입니다.<br>");
    document.write("해당 분은: <strong>" + thisMin + "</strong> 입니다.<br>");
    document.write("해당 초는: <strong>" + thisSec + "</strong> 입니다.<br>");
    document.write("해당 밀리세컨: <strong>" + thisMilliSec + "</strong> 입니다.<br>");

```

/* 만약 오늘날짜에서 크리스마스날 alert창이 뜨게하려면? */

```

// 오늘 날짜
var start = new Date();
// 목표 날짜 (크리스마스 이브)
var end = new Date("2025-12-24");
// 밀리초 차이
var diff = end.getTime() - start.getTime();
// 변환: 일 단위
var days = Math.floor(diff / (1000 * 60 * 60 * 24));

if (days > 0) {

```

```

    document.write("크리스마스 이브까지 " + days + "일 남았습니다!");
} else if (days === 0) {
    alert("오늘은 크리스마스 이브입니다!");
} else {
    document.write("크리스마스 이브가 지났습니다 ");
}
</script>

```

문자열 객체 메서드 / 속성

기본 데이터형 3총사 : String(문자), Number(숫자), Boolean(불린)

```

k
12
29
luck
Hello Thank me good luck to you
12
lo T
luck
hello thank you good luck to you
HELLO THANK YOU GOOD LUCK TO YOU
32
32

```

◆ 문자열 메서드 정리

```

<script>
    var str = "Hello Thank you good luck to you";

```

```
//인덱스 번호에 해당하는 (문자)를 반환  
document.write(str.charAt(10), "<br>"); //k
```

```
//왼쪽부터 찾을 문자와 제일 먼저 일치하는 문자의 (인덱스번호)를 반환 (암기)  
document.write(str.indexOf("you"), "<br>"); //12
```

```
//오른쪽부터 찾을 문자와 제일 먼저 일치하는 문자의 (인덱스번호)를 반환  
document.write(str.lastIndexOf("you"), "<br>"); //29 | 찾는 값이 없을 경우 -  
1 반환
```

```
//왼쪽부터 찾을 문자와 일치하는 문자를 찾아, 제일 먼저 일치하는 문자를 반환 (암기)  
document.write(str.match("luck"), "<br>"); //luck | 찾는 값이 없을 경우 null  
반환
```

```
//왼쪽부터 바꿀 문자와 일치하는 문자를 찾아, 제일 먼저 일치하는 새문자를 치환 (암기)  
document.write(str.replace("you", "me"), "<br>"); // you→me
```

```
//왼쪽부터 찾을 문자와 제일 먼저 찾는 문자의 (인덱스번호)를 반환  
document.write(str.search("you"), "<br>"); //12
```

```
//문자를 자른후, 남는 문자를 치환 (암기) => 인덱스번호 3부터 7이전까지 추출 (= >3,4,5,6)  
document.write(str.slice(3, 7), "<br>"); //lo T
```

```
//문자 갯수만큼 문자열 반환 (암기) = > 인덱스번호 21부터 ,글자수4개 추출  
document.write(str.substr(21, 4), "<br>"); //luck
```

```
//영문대문자를 모두 소문자로 치환 (암기)  
document.write(str.toLowerCase(), "<br>"); // LowerCase
```

```
//영문소문자를 모두 대문자로 치환 (암기)  
document.write(str.toUpperCase(), "<br>"); // UpperCase
```

```
//갯수반환 (암기)
document.write(str.length, "<br>"); //총 길이 갯수 32
//문자열 앞 또는 뒤에 공백 문자열 삭제
document.write(str.length, "<br>"); //총 길이 갯수 32
</script>
```

1) `charAt(index)`

- 해당 인덱스 번호의 **문자 반환**

```
str.charAt(10); // "k"
// "Hello Thank me good luck to you"
```

2) `indexOf("문자열")`

- 왼쪽(앞)부터 검색 → 제일 먼저 찾은 문자열의 **인덱스 번호 반환**
- 없으면 `1`

```
str.indexOf("you"); // 12
// "Hello Thank me good luck to you"
```

3) `lastIndexOf("문자열")`

- 오른쪽(뒤)부터 검색 → 제일 먼저 찾은 문자열의 **인덱스 번호 반환**
- 없으면 `1`

```
str.lastIndexOf("you"); // 29
// "Hello Thank me good luck to you"
```

4) `match("문자열")`

- 찾는 문자열과 일치하는 값을 **반환**
- 없으면 `null`

```
str.match("luck"); // "luck"  
// "Hello Thank me good luck to you"
```

5) `replace("찾을값","바꿀값")`

- 첫 번째로 찾은 문자열을 다른 문자열로 **치환**

```
str.replace("you","me");  
// "Hello Thank me good luck to you"
```

6) `search("문자열")`

- 문자열에서 처음 등장하는 위치의 **인덱스 번호 반환**

```
str.search("you"); // 12  
// "Hello Thank me good luck to you"
```

7) `slice(start, end)`

- 문자열을 **start** 인덱스부터 **end** 직전까지 추출

```
str.slice(3,7); // "lo T" (3,4,5,6)  
// "Hello Thank me good luck to you"
```

8) `substr(start, length)`

- start 인덱스부터 지정한 **length** 길이만큼 추출

```
str.substr(21,4); // "luck"  
// "Hello Thank me good luck to you"
```

9) `toLowerCase()`

- 영문 전체를 소문자로 변환

```
str.toLowerCase();
```

10) toUpperCase()

- 영문 전체를 대문자로 변환

```
str.toUpperCase();
```

11) length

- 문자열의 총 길이(문자 수) 반환

```
str.length; // 32
// "Hello Thank me good luck to you"
```

✓ 최종 정리표

메서드	기능	예시 결과
charAt(10)	10번째 문자 반환	"k"
indexOf("you")	"you" 첫 위치(왼→오)	12
lastIndexOf("you")	"you" 마지막 위치(오→왼)	29
match("luck")	"luck"와 일치 → 반환	"luck"
replace("you", "me")	"you" → "me" 치환	"Hello Thank me good luck to you"
search("you")	"you" 첫 인덱스 번호 반환	12
slice(3,7)	인덱스 3~6 추출	"lo T"
substr(21,4)	인덱스 21부터 4글자 추출	"luck"
toLowerCase()	소문자로 변환	"hello thank you good luck to you"
toUpperCase()	대문자로 변환	"HELLO THANK YOU GOOD LUCK TO YOU"
length	문자열 길이 반환	32



1. 문자열 실습

비밀번호**8**자가 입력되었습니다.

소문자로 받아 대문자로 바꾼 이름은**KHS**입니다

입력받은 이름은 **kimhyunsu** 입니다.

입력받은 전화번호는 **0101234****** 입니다.

```
<style>
  strong {
    color: red;
  }
</style>
<script>
  /*
    실습1.(프롬프트)비밀번호를 입력받아,
    (경고창)8자 미만이면 "8자 이상으로 다시 입력하십시오"
    (브라우저)8자 이상이면 "비밀번호 ???자가 입력되었습니다."
  */

  var str = prompt("비밀번호 입력", 11111111);
  console.log(str.length);

  if (str.length < 8) {
    alert("다시입력해주세요");
    location.reload();
  } else {
    document.write("비밀번호<strong>" + str.length + "</strong>자가 입력되
었습니다.<br>")
  }

  /*
```

```

실습2.
(프롬프트) 이름을 영문소문자로 입력받아, 대문자로 치환
(브라우저)"입력한 이름은 ??? 입니다.
*/

var name = prompt("소문자 입력", "khs");
var upper = name.toUpperCase();
console.log(upper);
document.write("소문자로 받아 대문자로 바꾼 이름은<strong>" + upper + "</strong>입니다<br>");

/*
실습3. (프롬프트)영문 이름 입력받아, 대문자입력시 소문자로 치환
(도큐먼트) 입력받은 이름은 ??? 입니다.
(프롬프트) 전화번호 입력받아, 끝 네자리를 "****"로 치환 = substr(),replace()
(도큐먼트) 전화번호는 0101234**** 입니다.
*/

var name = prompt("영문 이름 입력", "KIMHYUNSU");
var lowerName = name.toLowerCase(); // 대문자를 소문자로
document.write("입력받은 이름은 <strong>" + lowerName + "</strong> 입니다.<br>");

var phone = prompt("전화번호 입력", "01012345678");
var hiddenphone = phone.substr(0, phone.length - 4) + "****"; //받은 번호
뒤에서 4칸

document.write("입력받은 전화번호는 <strong>" + hiddenphone + "</strong> 입니다.<br>");

//Not a Number 숫자가 아니다.
</script>

```

◆ 실행 결과

- 입력:
 - 이름: KIMHYUNSU
 - 전화번호: 01012345678
- 출력:

입력받은 이름은 kimhyunsu 입니다.
입력받은 전화번호는 0101234**** 입니다.

✓ 요약

- `substr(num.length - 4, 4)` → 전화번호의 마지막 4자리 뽑기
- `replace(끝4자리, "****")` → 뽑아낸 부분을 `****` 로 치환

📌 BOM, 브라우저 객체

```
<script>
/* 1. location 객체 = 브라우저 주소창과 연관 */
var infoBrower = "현재 주소는 : " + location.href + " 입니다.<br>";
document.write(infoBrower);
infoBrower = "현재 사용하는 프로토콜 : " + location.protocol + " 입니다.<br>";
document.write(infoBrower);
infoBrower = "현재 사용하는 포트번호 : " + location.host + " 입니다.<br>";
document.write(infoBrower);

/* 2. navigator 객체 = 브라우저 정보 */
var infoBrower2 = "";
infoBrower2 = "브라우저 코드명 : " + navigator.appCodeName + "<br>";
document.write(infoBrower2);
infoBrower2 = "브라우저 이름 : " + navigator.appName + "<br>";
document.write(infoBrower2);
```

```

infoBrower2 = "브라우저 버전 : " + navigator.appVersion + "<br>";
document.write(infoBrower2);
infoBrower2 = "브라우저 사용언어 : " + navigator.language + "<br>";
document.write(infoBrower2);
infoBrower2 = "브라우저 엔진이름 : " + navigator.product + "<br>";
document.write(infoBrower2);
infoBrower2 = "브라우저 환경 : " + navigator.platform + "<br>";
document.write(infoBrower2);

// (알기)
infoBrower2 = "브라우저 환경 : " + navigator.userAgent + "<br>";
document.write(infoBrower2);

/* 3.infoBrowserTotal 변수명 한번만 사용하여 위의 정보를 (도큐먼트)에 나오도록
코딩 */
var infoBrowserTotal = "";
infoBrowserTotal += "브라우저 코드명 : " + navigator.appCodeName + "<br
>";
infoBrowserTotal += "브라우저 이름 : " + navigator.appName + "<br>";
infoBrowserTotal += "브라우저 버전 : " + navigator.appVersion + "<br>";
infoBrowserTotal += "브라우저 사용언어 : " + navigator.language + "<br>";
infoBrowserTotal += "브라우저 엔진이름 : " + navigator.product + "<br>";
infoBrowserTotal += "브라우저 환경 : " + navigator.platform + "<br>";
infoBrowserTotal += "브라우저 환경 : " + navigator.userAgent + "<br>";
document.write(infoBrowserTotal);
</script>

```

◆ 출력 결과 예시 (크롬 기준)

현재 주소는 : http://127.0.0.1:5500/200_JS/16_%EB%B8%8C%EB%9D%BC%EC%9A%B0%EC%A0%80%EA%B0%9D%EC%B2%B4.html 입니다.
현재 사용하는 프로토콜 : http: 입니다.
현재 사용하는 포트번호 : 127.0.0.1:5500 입니다.
브라우저 코드명 : Mozilla
브라우저 이름 : Netscape
브라우저 버전 : 5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/140.0.0.0 Safari/537.36
브라우저 사용언어 : ko-KR
브라우저 엔진이름 : Gecko
브라우저 환경 : Win32
브라우저 환경 : Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/140.0.0.0 Safari/537.36
브라우저 코드명 : Mozilla
브라우저 이름 : Netscape
브라우저 버전 : 5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/140.0.0.0 Safari/537.36
브라우저 사용언어 : ko-KR
브라우저 엔진이름 : Gecko
브라우저 환경 : Win32
브라우저 환경 : Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/140.0.0.0 Safari/537.36

✓ 포인트 정리

- `location` 객체 → 주소창 관련 정보
- `navigator` 객체 → 브라우저, OS, 환경 정보
- 모든 정보를 `infoBrowserTotal` 문자열에 누적(+=)
- 마지막에 `document.write(infoBrowserTotal)` 한 번만 호출

💀 1. BOM, 브라우저 객체 실습

사용자 OS 환경이 윈도우 사용자면 MS Windows OS 사용자(도큐먼트)
아니면, MS Windows OS 이외의 사용자(도큐먼트)



MS Window OS 사용자

```

<script>
    var client = navigator.userAgent.toLowerCase(); //사용자 정보를 소문자로 전
    부변경
    console.log(client); // 콘솔 찍고

    var result = client.indexOf("windows"); // window라는 text를 찾아
    console.log(result); // 콘솔에 있는지 확인

    if (result) { // 만약 있다면
        document.write(' <br
    >')
        document.write('MS Window OS 사용자');
    } else { // 없다면
        document.write(' <br>');
        document.write('MS Window OS 이외의 사용자');
    }
}
</script>

```

Screen 객체 : 사용자의 화면 크기or 정보를 알아내기

UI 영역을 제외한 너비 :1920
 UI 영역을 제외한 높이 :1080
 pc사용자

```

<script>
    var aW = screen.width; // 가로 너비 1920
    document.write("UI 영역을 제외한 너비 : " , aW + "<br>");

```

```

var aH = screen.height; // 세로 높이 1080
document.write("UI 영역을 제외한 높이 :", aH + "<br>");

/* 현재 사용자가 모바일을 사용하고 있으면 (브라우저)모바일사용자
현재 사용자가 PC를 사용하고 있으면 (브라우저)PC사용자 520 */

if(aW <= 520){
    document.write('모바일사용자');
}else{
    document.write('pc사용자');
}

//PC브라우저(1920x1080) , 테블릿(1024x768) , 모바일(520)
</script>

```

문제

다음 조건에 맞는 자바스크립트 코드를 작성하세요.

- 오늘 날짜 객체를 생성하고, **현재 연도와 월**을 화면에 출력한다.
(단, 월은 1~12 형식으로 나오게 할 것)
- 사용자에게 영문 이름을 `prompt()` 로 입력받고, 입력받은 이름을 **전부 대문자로 변환하여 출력**한다.
- 문자열 `"Hello JavaScript"` 에서 `"Script"` 부분만 잘라서 출력한다.
(힌트: `slice()` 또는 `substr()` 사용)
- 현재 브라우저의 **사용 언어(navigator.language)** 와 **현재 주소(location.href)** 를 한 번에 출력한다.

예시 출력 (입력이 kim 일 경우)

현재 연도는 2025년이고, 현재 월은 9월입니다.

입력받은 이름은 KIM 입니다.

문자열에서 잘라낸 부분은 Script 입니다.

브라우저 사용 언어: ko-KR

현재 주소: http://localhost:5500/test.html

현재 연도는2025년이고, 현재 월은 9월입니다.

입력받은 이름은 KIM 입니다.

문자열에서 잘라낸 부분은 Script 입니다.

브라우저 사용언어 : ko-KR 입니다.

현재 주소는 : http://127.0.0.1:5500/200_JS/17_test.html 입니다.

<script>

```
var today = new Date();
var thisYear = today.getFullYear(); //2025
var thisMonth = today.getMonth() + 1; // 9월
var user = prompt("이름을 작성해주세요.", "kim");
var reName = user.toUpperCase();
var str = "Hello JavaScript";
```

```
var lang = navigator.language;
var href = location.href;
```

```
document.write("현재 연도는" + thisYear + "년이고, 현재 월은 " + thisMonth
+ "월입니다.<br>");
document.write("입력받은 이름은 " + reName + " 입니다.<br>");
document.write("문자열에서 잘라낸 부분은 " + str.slice(10, 16) + " 입니다.<br>");
```



```
>");
    document.write("브라우저 사용언어 : " + lang + " 입니다.<br>");
    document.write("현재 주소는 : " + href + " 입니다.<br>");

</script>
```

1. 날짜 객체

```
var today = new Date();
var thisYear = today.getFullYear();
var thisMonth = today.getMonth() + 1;
```

👉 `getMonth()` 는 0부터 시작이라 `+1` 해주신 것도 정확합니다. ✅

1. 영문 이름 대문자로 변환

```
var user = prompt("이름을 작성해주세요.", "kim");
var reName = user.toUpperCase();
```

👉 입력값을 전부 대문자로 변환해서 출력, 조건과 딱 맞습니다. ✅

1. 문자열 추출

```
var str = "Hello JavaScript";
str.slice(10, 16) // "Script"
```

👉 `slice()` 로 정확히 `"Script"` 뽑으셨습니다. (`substr(10,6)` 도 같은 결과) ✅

1. 브라우저 정보

```
var lang = navigator.language;
var href = location.href;
```

👉 언어와 주소 모두 잘 가져오셨고, 출력도 정상입니다. ✅

◆ 개선 팁 (선택사항)

- `slice(10, 16)` 대신 `"JavaScript".slice(4)` 같은 방식도 가능 → 더 간단할 수도 있어요.
 - `document.write()` 대신 `<div id="result"></div>` + `innerHTML` 로 넣으면, 새로고침 시 다른 내용이 사라지지 않아 더 실무적으로 좋아요.
-