

if & for문 (control statement)

◆ 제어문(Control Statement)

프로그램의 흐름을 제어하는 문법. 크게 **조건문(분기)**과 반복문으로 나뉜다.

```
/* a 단일 if문
   if(조건식:true,false){
       실행문1;
       실행문2;
       ...
       조건식이 true일 경우 실행할 자바스크립트코드
   }
*/
```

1. 조건문 목차

- 특정 조건이 `true / false` 인지에 따라 코드 실행을 달리함.

✓ if 문

```
if (조건식) {
    실행문;
}
```

✓ if ~ else 문

```
if (조건식) {
    실행문1;
} else {
    실행문2;
}
```

✓ if ~ else if ~ else 문

```
if (조건1) {  
    실행문1;  
} else if (조건2) {  
    실행문2;  
} else {  
    실행문3;  
}
```

✓ switch 문

- 값(변수, 식)의 결과에 따라 분기 처리.

```
switch (값) {  
    case 1:  
        실행문1;  
        break;  
    case 2:  
        실행문2;  
        break;  
    default:  
        실행문기본;  
}
```

◆ 반복문(Loop Statement)

동일한 코드를 여러 번 실행할 때 사용.

1. for문

- 반복 횟수가 정해져 있을 때 주로 사용.

```
for (int i = 0; i < 5; i++) {  
    System.out.println("i = " + i);  
}
```

2. while문

- 조건이 `true` 일 동안 반복.

```
int i = 0;  
while (i < 5) {  
    System.out.println("i = " + i);  
    i++;  
}
```

3. do ~ while문

- 조건을 확인하기 전에 최소 한 번은 실행됨.

```
int i = 0;  
do {  
    System.out.println("i = " + i);  
    i++;  
} while (i < 5);
```

4. 향상된 for문 (for-each)

- 배열, 컬렉션의 요소를 순차적으로 꺼낼 때 사용.

```
int[] numbers = {1, 2, 3, 4};  
for (int n : numbers) {  
    System.out.println(n);  
}
```

◆ 반복 제어문 (반복문 안에서 흐름 제어)

반복문 내부에서 흐름을 강제로 바꿀 때 사용.

- **break** → 반복문을 즉시 종료
- **continue** → 현재 반복만 건너뛰고 다음 반복 실행
- **return** → 메서드 자체 종료

```
for (int i = 1; i <= 5; i++) {  
    if (i == 3) continue; // 3은 건너뛴  
    if (i == 5) break;    // 5에서 반복 종료  
    System.out.println(i);  
}
```

if 문(조건)을 활용해서 결과값은 항상 불린(t,f)으로 리턴한다.

※조건문. html (if문)

실습 1. 단일 if문 (프롬프트) 점수를 입력받아,

60점 이상이면 (경고창)필기 합격 , 60점 미만이면 (브라우저)"필기 불합격"

```
/* 실습 1. 단일 if문 (프롬프트) 점수를 입력받아,  
   만약 , 60점 이상이면 (경고창)필기 합격 , 60점 미만이면 (브라우저)"필기 불합격"  
   */  
  
var score = prompt("점수를 입력하세요",0);  
if (score >= 60) {  
    alert("필기 합격");  
    document.write("<h1>필기 합격</h1>");  
}  
if(score < 60) {  
    alert(" 불합격");  
    document.write("<h1>필기 불합격</h1>");  
}
```

실습 2. 단일 if문 (프롬프트)숫자를 입력받아,
숫자가 짝수이면 "짝수입니다"(경고창),
홀수면 "홀수입니다."(경고창)

```
/*
실습2. 단일 if문 (프롬프트)숫자를 입력받아,
숫자가 짝수이면 "짝수입니다"(경고창), 홀수면 "홀수입니다."(경고창)
*/
var num = prompt("숫자를 입력", 1);

console.log("변환 전" + typeof num); // string
num = Number(num); // string -> number 변환

if (num % 2 == 0) {
    alert("짝수");
}

if (num % 2 != 0) {
    alert("홀수");
}

console.log("변환 후" + typeof num); // number
```

0, "", null, undefined, NaN : false

그 외 : true

※조건문02. html (if문 실습)

실습3. (프롬프트) 나이를 입력받아,

(경고창) 만약 , 40세 이상이면 "건강검진대상" , 아니면 "건강검진대상아님"

```
<title>if문 실습</title>
<script>
  /* 실습1. (프롬프트) 나이를 입력받아,
   (경고창) 만약 , 40세 이상이면 "건강검진대상" , 아니면 "건강검진대상아님"
   */

  let age = prompt("나이를 입력하세요" ,33);
  if (age >= 40) {
    alert("건강검진대상");
  } else {
    alert("건강검진대상아님");
  }
}
```

127.0.0.1:5500 내용:

나이를 입력하세요

확인 취소

127.0.0.1:5500 내용:

건강검진대상아님

확인

실습4. (프롬프트) 나이를 입력받아,

(경고창) 만약 , 20세 이상이고 40세 미만이면 "서류심사통과" , 아니면 "서류심사보류"

```

/* 실습2. (프롬프트) 나이를 입력받아,
(경고창) 만약 , 20세 이상이고 40세 미만이면 "서류심사통과" , 아니면 "서류심사보류"
*/

let age2 = prompt("나이를 입력하세요", 33);
if (age2 >= 20 && age2 < 40) {
    alert("서류심사통과");
} else {
    alert("서류심사보류");
}

```

if (age2 >= 20 && age2 < 40) { // 20 이상 40 이하인 경우 실행

실습5.

(프롬프트) ID를 입력받아,

(브라우저) 만약, ID가 입력되었으면, "???님 반갑습니다!!",

(경고창) ID를 입력하지않았으면, "ID를 입력하세요

```

/* 실습3. (프롬프트) ID를 입력받아,
(브라우저) 만약, ID가 입력되었으면, "???님 반갑습니다!!",
(경고창) ID를 입력하지않았으면, "ID를 입력하세요"
*/

let userId = prompt ("아이디를 입력하세요.");

if(userId) {
    document.write(userId + "님 반갑습니다!!");
} else {
    alert("ID를 입력하세요");
    location.reload(); // 페이지 새로고침
}

```

기술문서 개념 (**location** , **API** , **AI**)

1 **location**

- 브라우저에서 현재 페이지의 **주소(URL)** 정보를 나타내는 객체
- 보통 ****주소창(URL)****과 관련 있음
- 예시 (JavaScript):

```
console.log(location.href); // 현재 페이지 전체 URL
console.log(location.hostname); // 도메인 이름
console.log(location.pathname); // 경로
```

2 **API** (Application Programming Interface)

- 응용 프로그램이 서로 통신할 수 있게 제공하는 인터페이스
- 개발자가 복잡한 기능을 직접 구현하지 않고 사용할 수 있도록 제공되는 함수/서비스
- 예시:
 - 브라우저에서 날씨 데이터를 가져오는 **Weather API**
 - 결제 기능을 제공하는 **Payment API**
- 구조:
 1. 요청(Request) → 서버
 2. 서버 처리 → 응답(Response)

3 **AI** (Artificial Intelligence, 인공지능)

- 인간의 지능을 모방하여 학습, 추론, 판단, 문제 해결을 수행하는 기술
- 예시:
 - 챗봇 (ChatGPT)
 - 이미지 생성 AI (DALL·E, MidJourney)
 - 추천 시스템 (유튜브, 넷플릭스)

실습6.

(프롬프트) 국어,영어,수학 각각의 점수를 입력받아,

(경고창) 만약, 3과목 평균이 80점 이상이고, 각각 과목의 점수가 60점 이상이면 "시험통과",

(경고창) 아니면 "재시험"

```
/* 실습4. (프롬프트) 국어,영어,수학 각각의 점수를 입력받아,  
(경고창) 만약, 3과목 평균이 80점 이상이고, 각각 과목의 점수가 60점 이상이면 "시험통과",  
(경고창) 아니면 "재시험"  
*/  
  
let kor = Number(prompt("국어점수를 입력하세요", 80));  
let eng = Number(prompt("영어점수를 입력하세요", 80));  
let math = Number(prompt("수학점수를 입력하세요", 78));  
  
let scores = [kor, eng, math]; // 총 과목  
  
if (scores.every(i => i >= 60)) {  
    alert("시험통과");  
} else {  
    alert("재시험");  
    location.reload(); // 페이지 새로고침  
}
```

```
let kor = Number(prompt("국어점수를 입력하세요", 80));  
let eng = Number(prompt("영어점수를 입력하세요", 80));  
let math = Number(prompt("수학점수를 입력하세요", 78));  
  
if ((kor + eng + math) / 3 >= 60 && kor >= 60 && eng >= 60 && math >= 60) {  
    // 실행문  
}
```

※ 코드 복잡함을 줄이기위해

```
let scores = [kor, eng, math]; // 총 과목

if (scores.every(score => score >= 60)) {
  // 실행문
}
```

이런 방식도 가능하다.

127.0.0.1:5500 내용:

영어점수를 입력하세요

확인 취소

127.0.0.1:5500 내용:

수학점수를 입력하세요

확인 취소

127.0.0.1:5500 내용:
국어점수를 입력하세요

확인 취소

127.0.0.1:5500 내용:
시험통과

확인

수업에서 다루지 않았지만 여기서 나오는 every()에 대한 정의는 다음과 같다.

◆ every() 정의

- 배열의 모든 요소가 조건을 만족하는지 검사
- 모든 요소가 조건을 만족하면 → `true`
- 하나라도 만족하지 않으면 → `false`

배열.`every`(콜백함수)

- 콜백함수: 각 요소를 검사하는 함수
- 인자: (요소, 인덱스, 배열)
- 반환값: Boolean (`true` / `false`)

1 배열 + every() 활용 (추천)

```
if ([kor, eng, math].every(score => score >= 60)) {
  // 실행문
}
```

- `[kor, eng, math]` → 점수 배열
- `.every(score => score >= 60)` → 모든 점수가 60 이상인지 체크
- 평균 계산 없이, 개별 점수가 60 이상이면 자동으로 평균도 60 이상이다.

※ 중요 : 모든 함수는 안쪽부터 계산식으로 캐스팅 들어간다 !!

◆ 예제 1 — 숫자 배열

```
let scores = [70, 80, 90];

let allPass = scores.every(score => score >= 60);
console.log(allPass); // true, 모든 점수가 60 이상
```

- 배열의 모든 점수가 60 이상이면 `true`
- 하나라도 60 미만이면 `false`

◆ 예제 2 — 문자열 배열

```
let words = ["apple", "banana", "cherry"];

let allLong = words.every(word => word.length >= 5);
console.log(allLong); // true, 모든 단어 길이가 5 이상
```

※조건문03. html (else if : 그외 밖에)

조건문, else if문 | 중첩 if문

```

/* 1. else if문 = 조건이 여러개일 경우 사용 */
/*
var grade = Number(prompt('점수입력', 90));

if (grade >= 90) {
    alert('A학점');
} else if (grade >= 80) {
    alert('B학점');
} else if (grade >= 70) {
    alert('C학점');
} else if (grade >= 60) {
    alert('D학점');
} else {
    alert('재수강');
    location.reload(); // 재수강일 때만 새로고침
}
*/

/* 2. 중첩 if문 */
var dbId = "ezen";
var dbPw = "1234";
var id = prompt('ID를 입력하세요', 'ezen');

if (dbId == id) {
    // id가 일치하는 경우에만 비밀번호를 체크
    var pw = prompt('비밀번호를 입력해주세요', '1234');
    if (dbPw === pw) {
        alert('로그인성공');
    } else {
        alert('로그인실패');
    }
} else {
    alert('ID가 일치하지 않습니다.')
}
}

```

※조건문04. html (switch-case문)

선택문, switch-case문

```
switch (조건식) {  
  case n 조건식의 값 일치 여부:  
    실행문;  
    break;  
  ...  
  default;  
    실행문  
}
```

1. 입력받은 수가 홀수인지 짝수인지 판별

```
var num = Number(prompt('숫자입력', 0));  
console.log(num);  
console.log(typeof (num));  
switch (num % 2) {  
  case 0:  
    alert('짝수입니다.');    break  
  case 1:  
    alert('홀수입니다.');    break  
  default:  
    alert('숫자가 아닙니다');}
```

2. 포털사이트 접속하기

```

let site = prompt('구글, 네이버, 다음 중 하나 입력', '구글');
let url; // 미리 변수 선언

switch (site) { // 입력창 > 해당 케이스 > 리턴 > 콘솔
  case '구글':
    url = 'https://www.google.com';
    break;
  case '네이버':
    url = 'https://www.naver.com';
    break;
  case '다음':
    url = 'https://www.daum.net';
    break;
  default:
    alert('다시 입력해 주세요. ');
    location.reload();
}

console.log(site);
console.log(url);

if (url) {
  window.open(url, '_blank'); // 새 탭 열기
}

```

1 전역 변수 (Global Variable)

- 모든 코드에서 접근 가능한 변수
- 스크립트 최상단(함수나 블록 밖)에 선언
- 어디서든 읽고 수정 가능

```

var globalVar = '전역 변수';

function test() {
  console.log(globalVar); // 접근 가능
}

```

```
test(); // "전역 변수"
console.log(globalVar); // "전역 변수"
```

- 주의: 너무 많이 쓰면 다른 코드와 충돌 위험하다

2 지역 변수 (Local Variable)

- 함수 내부에서 선언한 변수
- 함수 내부에서만 사용 가능, 밖에서는 접근 불가

```
function test() {
  var localVar = '지역 변수';
  console.log(localVar); // 접근 가능
}

test(); // "지역 변수"
console.log(localVar); // ReferenceError
```

- `var` 는 함수 스코프(function scope)
- `let` 과 `const` 는 블록 스코프(block scope)

3 블록 스코프 (Block Scope)

- 중괄호 `{ }` 안에서만 유효한 범위
- `let` 과 `const` 변수에서 적용

```
if (true) {
  let blockVar = '블록 변수';
  const blockConst = 123;
  console.log(blockVar); // OK
  console.log(blockConst); // OK
}
```



```
console.log(blockVar); // ReferenceError
console.log(blockConst); // ReferenceError
```

- `var` 는 블록 스코프를 무시하고 **함수 스코프**만 가짐

```
if (true) {
  var x = 10;
}
console.log(x); // 10, 블록 밖에서도 접근 가능
```

스탠드얼론 - 웹브라우저 창이 바뀌면 정보가 안넘어가는 현상

1 스탠드얼론(Standalone) 정의

Standalone이란, 다른 프로그램이나 시스템에 의존하지 않고 **독립적으로 실행될 수 있는 프로그램, 장치, 앱**을 의미

- **독립 실행 가능** → 설치 후 별도의 서버, 네트워크, 다른 소프트웨어 없이 동작 가능
- **자체 완결형** → 단일 시스템/장치 안에서 모든 기능 수행 가능

2 예시

구분	설명	예시
소프트웨어	다른 프로그램 없이 혼자 실행 가능	메모장, 계산기, 단일 실행 파일(.exe)
모바일 앱	인터넷 연결 없이 동작 가능한 앱	계산기 앱, 오프라인 게임, 단어장 앱
장치	외부 지원 없이 기능 수행	USB 드라이브에 바로 실행되는 앱, 독립형 디지털 카메라

3 반대 개념

- 웹 기반 / 클라우드 기반: 서버나 네트워크 없이는 동작하지 않는 시스템
 - 예: Google Docs, Gmail, SaaS 서비스 등
-

4 요약

- 스탠드얼론 = 혼자서 동작 가능
- 장점: 설치 후 바로 사용 가능, 네트워크 의존성 없음
- 단점: 업데이트/동기화가 수동적일 수 있음

실습 (프롬프트) 지역명 '서울', '경기도','부산' 중 하나를 입력받아,
'서울'일 경우 지역번호 02,
'경기도'일 경우 지역번호 031,
'부산'일 경우 051을 데이터로 사용.
그외 데이터 입력시 (경고창) 다시 입력하세요.
(도큐먼트) ???의 지역번호는 ??? 입니다.

```
<style>
  h1.area{
    font-size:20px;
    color:■ red;
    display: inline-block;
  }
  span.region{
    color:■ blue;
  }
</style>
```

```

/* ※ 전역변수 , 지역변수 이해하기 blockscope */
/*
실습 (프롬프트) 지역명 '서울', '경기도', '부산' 중 하나를 입력받아,
'서울'일 경우 지역번호 02,
'경기도'일 경우 지역번호 031,
'부산'일 경우 051을 데이터로 사용.
그외 데이터 입력시 (경고창) 다시 입력하세요.
(도큐먼트) ???의 지역번호는 ??? 입니다.
*/

let area, code;

area = prompt("지역명을 입력하세요. (서울, 경기도, 부산 중 하나)", "서울");
switch (area) {
  case "서울":
    code = "02";
    break;
  case "경기도":
    code = "031";
    break;
  case "부산":
    code = "051";
    break;
  default:
    alert("다시 입력하세요.");
    break;
}

document.write("<h1 class='area'>${area}</h1>의 지역번호는 <span class='region'>${code}</span> 입니다.");

```

서울의 지역번호는 02 입니다.

※반복문01. html (while문)

1. 초깃값 , 2. 조건식 3. 증감식 으로 구성된다.

```

/* 1. 초기값 */
var i = 1;

/* 2. 조건식 */
while(i<5){
    document.write("안녕하세요" + i + "<br>")
    i++;
}
/* 3. 증감식 */
document.write("=====while문 종료=====5");

```

```

안녕하세요1
안녕하세요2
안녕하세요3
안녕하세요4
=====while문 종료=====5

```

i 는 초기값에 1을 갖고 , while 조건문을 통해 증감연산자 ++ ;
i 가 5에 해당하는 조건까지 document.write 문서에

안녕하세요 + i 를 5 반복 출력 해달라는 뜻이다.

※ 여기서 index 0,1,2,3,4 로 구분함으로 4번만 출력된다.

※조건문 종합실습 (1~5)



해당 자료는 코드를 보고 해석하고 이해하는것에 중점을 두어야
나중에 수정할수있는 능력을 키울수가 있다.

```

/*
실습1. (프롬프트)이름,신장,몸무게를 입력받아,
적정체중이 아니면, ???님 건강을 위해 체중관리 요망
적정체중이면, ???님 건강관리 잘 하셨습니다
적정체중 식 = 키 -100*0.9
(브라우저출력)
*/

let name = prompt("이름이 무엇입니까?", "김현수");
let height = Number(prompt("키는 몇입니까?", "180"));
let weight = Number(prompt("몸무게는 몇 입니까?", "78"));

let normalWeight = (height - 100) * 0.9;
console.log("적정 몸무게:", normalWeight);

if (weight === normalWeight) {
  document.write("<h1 style='color:green'>" + name + "님, 건강관리 잘 하셨습니다! :)</h1>");
} else {
  document.write("<h1 style='color:red'>" + name + "님, 건강을 위해 체중 관리 요망 :( </h1>");
}

document.write("<p>현재 몸무게: " + weight + " kg</p>");
document.write("<p>적정 몸무게: " + normalWeight + " kg</p>");

```

```

/*
실습2.
(프롬프트)1월급여, 2월급여, 3월급여를 입력받아,
3개월 평균급여가 1000이상이면, 고소득자
1000이하면, 평균임금자
(브라우저출력)
*/

//초기값
let totalSalary = 0;
// 조건 + 증감식
for (let i = 1; i <= 3; i++) {
  totalSalary += Number(prompt(`${i}월 급여를 입력하세요:`, 500));
}
// 결과 = 평균 급여 / 3
totalSalary = totalSalary / 3;

if (totalSalary >= 1000) {
  document.write("<h2>" + name + "님의 3개월 평균 급여는:" + totalSalary + "이며,</h2><span><br> 고소득자 입니다</span><br>");
} else {
  document.write("<h2>" + name + "님의 3개월 평균 급여는:" + totalSalary + "이며,</h2><span><br> 평균임금자 입니다</span><br>");
}

```

You, 1초 전 • Uncommitted changes

```
/*
실습3. (프롬프트) 나이와 해당연도를 입력받아,
40세이상이고, 해당연도가 짝수해이고, 나이도 짝수일 경우
"건강검진 대상자입니다."
그렇지 않으면 "건강검진 대상자가 아닙니다."
(도큐먼트출력)
*/

let age = Number(prompt("나이는 몇살입니까?", "34"));
let year = Number(prompt("해당연도를 입력하세요", "2026"));

// 나이가 40 >= 해당연도 짝수 === 나이도 짝수
if (age >= 40 && year % 2 === 0 && age % 2 === 0) {
    document.write("건강검진 대상자입니다.<br>");
} else {
    document.write("건강검진 대상자가 아닙니다.<br>");
}
```

```

/*
실습4. else if문사용 (프롬프트) 월(month)을 입력받아,
9~11월이면 '가을입니다.'
6~8월이면 '여름입니다.'
3~5월이면 '봄입니다.'
그외 월이면 '겨울입니다.'를 else if문을 사용하여 코딩
(도큐먼트출력)
*/

let month = Number(prompt("해당 월을 입력하세요(1~12):", "9"));

if (month >= 3 && month <= 5) {
  document.write("현재 계절은 봄입니다.<br>");
} else if (month >= 6 && month <= 8) {
  document.write("현재 계절은 여름입니다.<br>");
} else if (month >= 9 && month <= 11) {
  document.write("현재 계절은 가을입니다.<br>");
} else {
  document.write("현재 계절은 겨울입니다.<br>");
}

```

You, 3분 전 • 0911

```

let position = prompt("1-웹디자인,2-웹개발,3-웹기획 중 숫자 하나 선택", "2");
let room, message;

switch (position) {
  case "1":
    room = "401호";
    message = "웹디자인 섹션은";
    document.write("<div class='flex'>" + message + "<h2 style='color:yellow'>" + room + "</h2>에서 진행</div>");
    break;
  case "2":
    room = "402호";
    message = "웹개발 섹션은";
    document.write("<div class='flex'>" + message + "<h2 style='color:yellow'>" + room + "</h2>에서 진행</div>");
    break;
  case "3":
    room = "403호";
    message = "웹기획 섹션은";
    document.write("<div class='flex'>" + message + "<h2 style='color:yellow'>" + room + "</h2>에서 진행</div>");
    break;
  default:
    document.write("<div class='flex'>잘못 입력하셨습니다</div>");
}

```