

04. 제어문

제어문

제어문 : 소스 실행 순서를 결정하는 명령문

조건에 따라 순서 조절하기 : if문, if...else문, switch문, 조건 연산자

반복 횟수 조절하기 : for문, while문, do...while문

소스 흐름에 영향을 주는 문 : continue문, break문

조건에 따라 흐름 조절하기

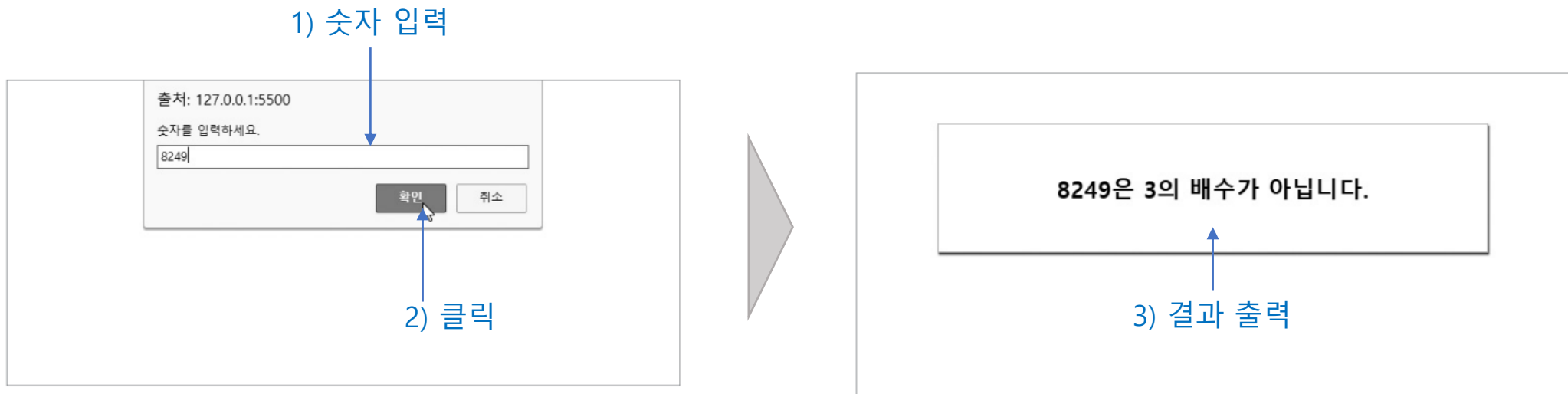
if 문

괄호 안의 조건이 true이면 { } 사이의 명령을 처리하고, false 이면 { } 안의 명령 무시

if ... else 문

if() 문의 괄호 안의 조건이 true이면 if 다음에 있는 { }의 명령을 처리하고,
false 이면 else 다음에 있는 { } 안의 명령 실행

[미리보기] 3의 배수 검사기 만들기

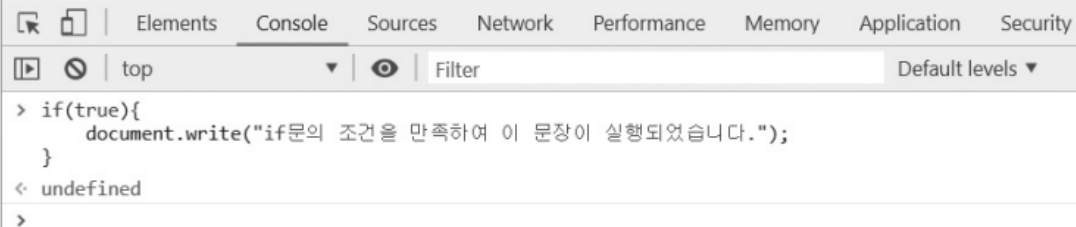


if 문

괄호 안의 조건이 true이면 { } 사이의 명령을 처리하고, false 이면 { } 안의 명령 무시

```
> if(true) {  
    document.write("if문의 조건을 만족하여 이 문장이 실행되었습니다.");  
}
```

if문의 조건을 만족하여 이 문장이 실행되었습니다.



if ... else 문

if() 문의 괄호 안의 조건이 true이면 if 다음에 있는 { }의 명령을 처리하고, false 이면 else 다음에 있는 { } 안의 명령 실행

```
> var number = prompt("숫자를 입력하세요.");  
> if(number < 0) {  
    alert("0 이상의 수를 입력하세요");  
}  
else {  
    document.write("입력한 숫자: " + number);  
}
```

조건 연산자

조건이 하나이고 실행할 명령도 하나일 때
(조건)? 명령1 : 명령2

```
> var score = 75;  
> (score >= 60) ? alert("통과") : alert("실패");
```

조건

조건이 true일 때 실행

조건이 false일 때 실행

truthy 값, falsy 값

일반 값 중에서도 'true로 인정할 수 있는 값' 과 'false로 인정할 수 있는 값'

"truthy하다"

"falsy하다"

falsy 값을 제외한 모든 값은 truthy

- ❶ 0 // 숫자
- ❷ "" // 빈 문자열
- ❸ NaN // 숫자가 아님(Not a Number)
- ❹ undefined
- ❺ null

```
> var input = prompt("이름을 입력하세요.");  
< if(input) {  
    alert("이름을 입력했습니다 : " + input);  
}  
else {  
    alert("이름을 입력하지 않았습니다.");  
}
```

input에 값이 들어 있으면 truthy합니다
(true와 동일합니다).

input에 값이 들어 있지 않으면 falsy합니다
(false와 동일합니다).

3의 배수 검사기 만들기

```
13  <script>
14      var userNumber = prompt("숫자를 입력하세요.");
15
16      if(userNumber != null) {
17          // [확인]을 눌렀을 때 실행할 명령
18      }
19      else {
20          alert("입력이 취소되었습니다."); // [취소]를 눌렀을 때 실행할 명령
21      }
22  </script>
```

3의 배수 검사기 만들기

```
13 <script>
14     var userNumber = prompt("숫자를 입력하세요.");
15     var displayArea = document.querySelector('#result');
16     ...
23 </script>
```

결과를 표시할 영역 가져오기

```
17 if(userNumber != null) {
18     if (userNumber % 3 === 0) {
19         displayArea.innerHTML = userNumber + "은 3의 배수입니다.";
20     }
21     else {
22         displayArea.innerHTML = userNumber + "은 3의 배수가 아닙니다.";
23     }
24 }
```

조건이 많을 때 switch문으로 흐름 조절하기

```
11 <script>
12     var session = prompt("관심 세션을 선택해 주세요. 1-마케팅, 2-개발, 3-디자인", "1");
13
14     switch(session) { ❶
15         case "1" : document.write("<p>마케팅 세션은 <strong>201호</strong>에서 진행됩니다.</p>"); ❷
16             break; ❸
17         case "2" : document.write("<p>개발 세션은 <strong>203호</strong>에서 진행됩니다.</p>");
18             break;
19         case "3" : document.write("<p>디자인 세션은 <strong>205호</strong>에서 진행됩니다.</p>");
20             break;
21         default: alert("잘못 입력했습니다."); ❹
22     }
23 </script>
```

- ❶ switch 예약어 오른쪽에 괄호와 함께 값을 확인할 변수를 지정합니다.
- ❷ 여러 가지 조건 값은 case문 다음에 콜론(:)을 붙여 지정하고, 해당 값일 때 실행할 명령도 콜론 다음에 나열합니다. 둘 이상의 명령을 실행한다면 명령을 중괄호 { }로 묶습니다.
- ❸ break문을 사용해서 명령을 실행한 다음에는 완전히 switch문을 빠져나오도록 소스를 작성합니다.
- ❹ 사용자가 입력한 값이 case문에서 지정한 조건 값과 전부 일치하지 않을 때 실행할 명령입니다. default:에서는 break문을 사용하지 않습니다.

명령 반복 실행하기 - for문

- 카운터 변수를 기준으로 명령을 여러 번 실행
- 여러 명령을 늘어 놓지 않고 소스를 간단하게 작성할 수 있음
- 소스의 양이 줄어 실행 속도가 빨라짐

```
> var sum = 0;
  sum += 1;
  sum += 2;
  sum += 3;
  sum += 4;
  sum += 5;
  document.write("1부터 5까지 더하면 " + sum);
```

반복문으로 바꾸면



```
10 <script>
11   var sum = 0;
12
13   for(var i = 1; i < 6; i++) { // i = 1부터 i = 5까지 총 5번 반복
14     sum += i; // sum = sum + i;를 줄여서 표현
15   }
16   document.write("1부터 5까지 더하면 " + sum);
17 </script>
```

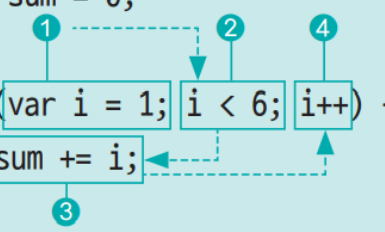
Diagram illustrating the execution flow of the for loop:

- 1: Initialization (var i = 1)
- 2: Condition check (i < 6)
- 3: Loop body execution (sum += i)
- 4: Increment (i++)

The loop repeats 5 times, incrementing i from 1 to 5.

명령 반복 실행하기 - for문

```
10 <script>
11   var sum = 0;
12
13   for(var i = 1; i < 6; i++) { // i = 1부터 i = 5까지
14     sum += i; // sum = sum + i;를
15   }
16   document.write("1부터 5까지 더하면 " + sum);
17 </script>
```



❶ 카운터 변수 선언

for문에서만 사용할 카운터 변수를 선언합니다.

❷ for문의 조건식

for문 안에 있는 소스를 실행할지 판단하는 조건입니다. 조건이 true라면 중괄호 안의 자바스크립트 소스를 실행하고, false라면 for문을 벗어나 다음에 나오는 소스를 실행합니다.

❸ 반복 실행할 자바스크립트 소스

for문에서 실행할 중괄호 안의 자바스크립트 소스입니다.

❹ 카운터 변수 조절

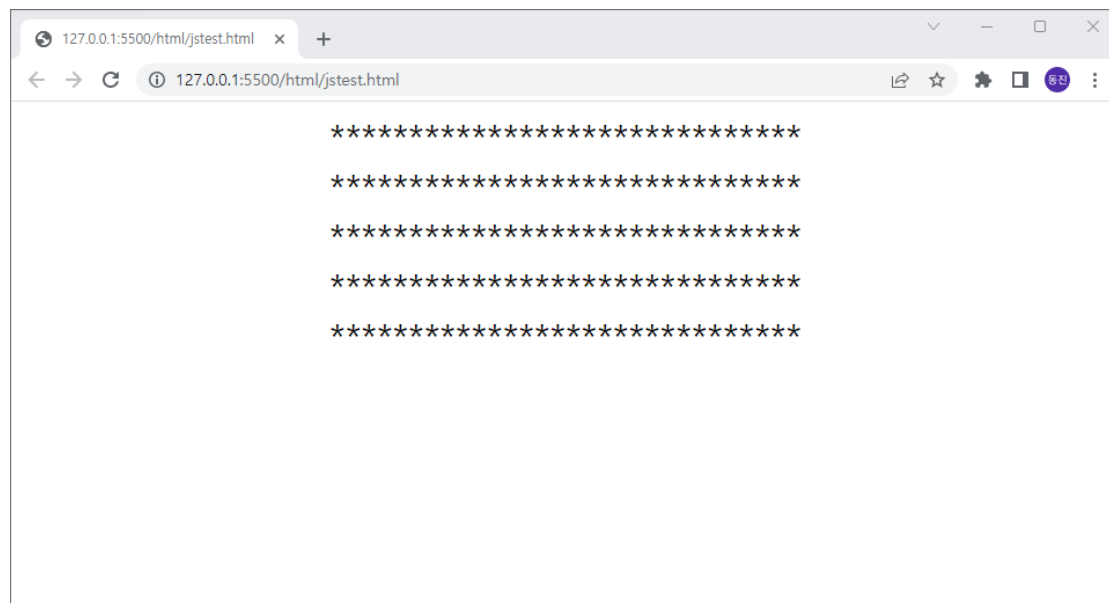
증감 연산자를 사용해 카운터 변수를 조절합니다. 위 소스에서는 중괄호를 실행한 이후 카운터 변수 i를 1만큼 증가(i++)시켰습니다.

중첩 for문

실습 파일 : 04Wfor.html

여러 줄의 별(*) 찍기 - *를 30개씩 5줄 표시하려면?

먼저 실행하는 for문(별 30개 찍기)을 안쪽에, 나중에 실행하는 for문(5줄 찍기)을 바깥쪽에 작성



중첩 for문

실습 파일 : 04Wfor.html

여러 줄의 별(*) 찍기 - *를 30개씩 5줄 표시하려면?

먼저 실행하는 for문(별 30개 찍기)을 안쪽에, 나중에 실행하는 for문(5줄 찍기)을 바깥쪽에 작성

```
34  ① for(var k = 0; k < 5; k++) {  
35      for(var i = 0; i < 30; i++) {  
36          ② document.write('*');  
37      }  
38      ③ document.write("<br>");  
39  }
```

- ① 바깥쪽 for문을 실행합니다(k = 0).
- ② 안쪽 for문을 실행해 30번 반복하고 빠져나옵니다.
- ③
 태그를 추가해서 줄을 바꿉니다.
- ④ 바깥쪽 for문의 조건식이 false가 될 때까지 반복합니다.

중첩 for문 - 구구단 만들기

구구단			
나중에 실행할 for문			
2단	3단	4단	5단
2 X 1 = 2	3 X 1 = 3	4 X 1 = 4	5 X 1 = 5
2 X 2 = 4	3 X 2 = 6	4 X 2 = 8	5 X 2 = 10
2 X 3 = 6	3 X 3 = 9	4 X 3 = 12	5 X 3 = 15
2 X 4 = 8	3 X 4 = 12	4 X 4 = 16	5 X 4 = 20
2 X 5 = 10	3 X 5 = 15	4 X 5 = 20	5 X 5 = 25
2 X 6 = 12	3 X 6 = 18	4 X 6 = 24	5 X 6 = 30
2 X 7 = 14	3 X 7 = 21	4 X 7 = 28	5 X 7 = 35
2 X 8 = 16	3 X 8 = 24	4 X 8 = 32	5 X 8 = 40
2 X 9 = 18	3 X 9 = 27	4 X 9 = 36	5 X 9 = 45
먼저 실행할 for문			

중첩 for문 - 구구단 만들기

실습 파일 :
04Wgugudan.html

```
13  for(var i = 2; i <= 9; i++) {  
14      document.write("<div>");  
15      document.write("<h3>" + i + "단</h3>");  
16      for (var j = 1; j <= 9; j++) {  
17          document.write(i + " X " + j + " = " + i * j + "<br>");  
18      }  
19      document.write("</div>");  
20 }
```

i = 2, j = 1일 때 $2 \times 1 = 1$ 표시
i = 2, j = 2일 때 $2 \times 2 = 4$ 표시

[미리보기] 팩토리얼 계산기

팩토리얼(!)이란

주어진 수	표현	계산식
1	1!	1
2	2!	1 * 2
3	3!	1 * 2 * 3

127.0.0.1:5500 내용:

숫자를 입력하세요.

확인

취소



5!= 120

[미리보기] 팩토리얼 계산기

팩토리얼(!)이란

주어진 수	표현	계산식
1	1!	1
2	2!	1 * 2
3	3!	1 * 2 * 3

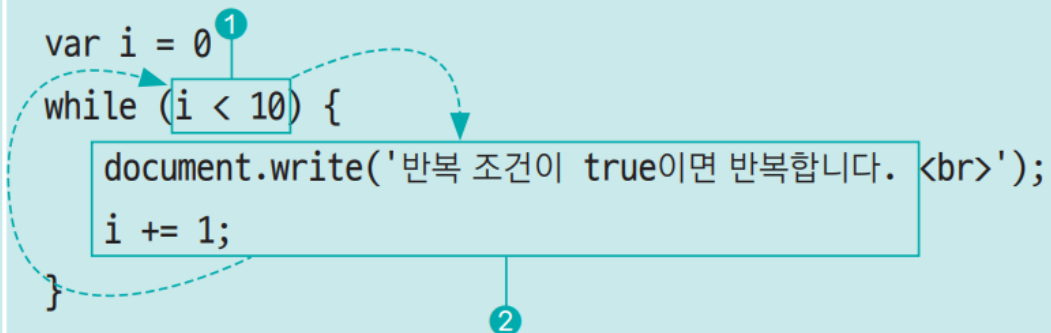
```
var input = prompt("숫자를 입력하세요.", 5)
let sum = 1;

for (i = 1; i <= input; i++) {
    sum *= i
}

document.write(`<p>${input} != ${sum} </p>`)
```

while문

```
> var i = 0
> while (i < 10) {
    document.write('반복 조건이 true이면 반복합니다. <br>');
    i += 1;
}
```



❶ 조건 검사

조건식을 검사합니다.

❷ 중괄호 안의 자바스크립트 소스 실행

조건식이 참이면 중괄호 안의 자바스크립트 소스를 실행합니다. 이때 조건식에 있는 카운터 변수 `i`에 1을 더하는 방법으로 반복 횟수를 제어해야 합니다. 위 예의 경우 `i = 9`까지 총 10회 반복합니다.

do...while문

- 조건의 결괏값이 false라고 해도 일단 문장이 최소 한번은 실행됨

```
> var i = 0
> do {
    document.write('반복 조건이 true이면 반복합니다. <br>'); // 명령 실행
    i += 1;
} while (i < 10); // 조건이 false가 되면 반복 종료
```

팩토리얼 계산 프로그램

```
12 var n = prompt("숫자를 입력하세요.");
13 var nFact = 1; // 1을 기본 값으로 지정
14 var i = 2;      // 1! = 1이기 때문에 i = 2부터 시작
15
16 while(i <= n) { // i = n까지 반복
17     nFact *= i; // nFact = nFact * i
18     i++;
19 }
20
21 documen.write(n + "!= " + nFact);
```

```
var input = prompt("숫자를 입력하세요.", 5)
var i = 1;
var sum = 1;

while(i <= input) {
    sum *= i;
    i = i + 1;
}
document.write(`<p>${input}!= ${sum} </p>`);
```

break문, continue문

break문

- 반복문의 흐름에서 바로 빠져나올 때 사용

continue문

- 주어진 조건에 맞는 값을 만났을 때 실행하던 반복 문장을 건너뛰고 반복문의 맨 앞으로 되돌아갑니다.

짝수 더하기 프로그램

실습 파일 : 04Weven.html

```
14  var n = 10;
15  var sum = 0;
16
17  for (var i = 1; i <= n; i++) {
18      if (i % 2 == 1) { // i를 2로 나눈 나머지가 1
19          continue;
20      }
21      sum += i;
22
23      document.write(i + " ----- " + sum + "<br>");
24  }
```