



# 4 반복문

모던 웹을 위한 Javascript jQuery 입문

## ❖ 반복문

- 여러 번 반복해야 하는 일을 간편하게 처리하는 반복문

```
<script>
  alert('출력');
  alert('출력');
  alert('출력');
  alert('출력');
  alert('출력');
</script>
```

```
<script>
  for (var i = 0; i < 100; i++) {
    alert('출력');
  }
</script>
```



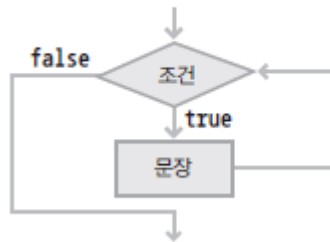
# 4.1 while 반복문

## ❖ while 반복문의 형태

- 조건이 변화하지 않는다면 반복문 안을 무한히 반복
- 조건을 거짓으로 만들 수 있는 내용이 문장 안에 포함되어야 함

```
while (불리언 표현식) {  
    문장  
}
```

그림 4-1 while 반복문



- 조건이 항상 참이므로 경고창이 무한 반복되는 잘못된 코드

```
while (true) {  
    alert('반복문');  
}
```



# 4.1 while 반복문

## ❖ 무한 루프

- 반복문이 무한 반복되는 것
  - 이 코드는 절대 실행하지 말 것

## ❖ while 반복문 종료 예제 1)

- 조건을 변화시킬 수 있는 요소 필요
  - 코드 4-1은 경고창 다섯 번 출력

코드 4-1 while 반복문 - 내부에서 조건 변경

```
<script>
  // 변수를 선언합니다.
  var value = 0;

  // 반복문
  while (value < 5) {
    alert(value + '번째 반복');
    value++;
  }
</script>
```



# 4.1 while 반복문

## ❖ while 반복문 종료 예제 2)

- 코드 4-2는 시간을 사용해 조건을 변화시켜 반복문을 빠져나감
  - 현재 시간을 변수 `startTime`에 저장 후 1000밀리초(1초) 후 반복문 종료
  - 외부에서 종료 조건을 변경할 수도 있음

코드 4-2 while 반복문 - 외부에서 조건 변경

```
<script>
    // 변수를 선언합니다.
    var value = 0;
    var startTime = new Date().getTime();

    // 반복문
    while (new Date().getTime() < startTime + 1000) { value++; }

    // 출력합니다.
    alert(value);
</script>
```



## 4.2 do while 반복문

### ❖ while 반복문

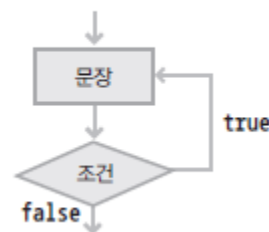
- 조건을 먼저 검사한 후 코드 블록 반복 실행
- 처음 조건을 검사했을 때 조건이 거짓이라면 내부의 문장을 한 번도 실행하지 않음

### ❖ do while 반복문

- 조건이 참인지 거짓인지와 상관없이 내부의 문장을 최소한 한 번은 실행해야 하는 경우 사용
- 조건을 비교하는 부분이 마지막에 위치
- do while 반복문의 기본적인 형태

```
do {  
    문장  
} while (불리언 표현식);
```

그림 4-2 do while 반복문



## 4.2 do while 반복문

### ❖ while 반복문을 do while 반복문으로 바꾸는 예제

코드 4-3 do while 반복문

```
<script>
  // 변수를 선언합니다.
  var value = 0;

  // 반복문
  do {
    alert(value + '번째 반복문');
    value++;
  } while (value < 5)
</script>
```



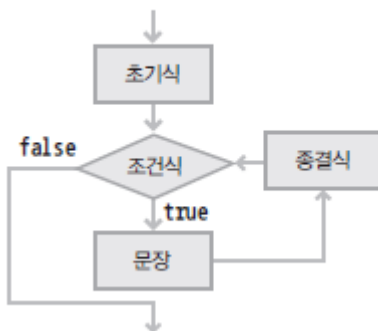
## 4.3 for 반복문

### ❖ for 반복문

- while 반복문은 조건에 비중을 두는 반복문
- 조건보다 횟수에 비중을 둘 때 for 반복문 사용
- for 반복문의 형태
  - while 반복문과 달리 초기식과 종결식 있음

```
for (초기식; 조건식; 종결식) {  
    문장  
}
```

그림 4-3 for 반복문





### ❖ for 반복문의 실행 단계

- length만큼 문장을 쉽게 반복시키는 장점
- 1. 초기식을 실행
- 2. 조건식과 비교한다. 조건이 거짓이면 반복문 종료
- 3. 문장 실행
- 4. 종결식 실행
- 5. 2단계로 감

```
for (var i = 0; i < length; i++) {  
    문장  
}
```



### ❖ while 반복문 예제와 같은 기능

- for 반복문의 초기문 안에서 만들어지는 변수
  - i, j, k, m, n과 같이 간단한 한 글자의 식별자로 만드는 것이 일반적
  - 이 예제에서는 while 반복문에서의 예제와 비교 - 변수 value 사용
- 단순 for 반복문
  - for(var i = 0; i < length; i+ +)와 같은 형태의 for 반복문
  - 0으로 시작해서 length 바로 전까지 반복을 수행하는 이유
    - 배열의 인덱스가 0에서 시작

코드 4-4 for 반복문

```
<script>
  // 반복문
  for (var value = 0; value < 5; value++) {
    alert(value + '번째 반복문');
  }
</script>
```



## ❖ 예제 코드 4-5

- 단순 for 반복문을 사용해 배열의 요소를 모두 출력
  - 출력순서에 따른 코드의 변화

코드 4-5 단순 for 반복문

```
<script>
// 변수를 선언합니다.
var array = ['포도', '사과', '바나나', '망고'];

// 반복문
for (var i = 0; i < array.length; i++) {
    alert(array[i]);
}
</script>
```

코드 4-6 역 for 반복문

```
<script>
// 변수를 선언합니다.
var array = ['포도', '사과', '바나나', '망고'];

// 반복문
for (var i = array.length - 1; i >= 0; i--) {
    alert(array[i]);
}
</script>
```



### ❖ 브라우저의 성능을 측정하는 프로그램

- 초기식에서 선언한 변수를 조건식이나 종결식에 사용할 필요는 없음
- 예제 코드 4-7]
  - 1초 동안 반복문이 몇 회 반복되는지 표시하는 프로그램
    - while 반복문에서 살펴본 코드 4-2와 같은 예제
    - 변수 startTime을 구하고 반복문 시작
    - 반복문은 1000밀리초(=1초) 후 종료

코드 4-7 반복문의 특이 사용

```
<script>
// 변수를 선언합니다.
var startTime = new Date().getTime();

// 반복문
for (var CPS = 0; new Date().getTime() < startTime + 1000; CPS++) { }
alert('초 당 연산 횟수: ' + CPS);
</script>
```



## 4.4 for in 반복문

### ❖ 자바스크립트의 for in 반복문

- 배열이나 객체를 쉽게 다룰 수 있게 제공하는 반복문
- for in 반복문의 형태
  - for in 반복문은 다음 단순 for 반복문과 같은 기능 수행

```
for (var i in array) {  
    }  
  
for (var i = 0; i < array.length; i++) {  
    }  

```

코드 4-8 for in 반복문

```
<script>  
    // 변수를 선언합니다.  
    var array = ['포도', '사과', '바나나', '망고'];  
  
    // 반복문  
    for (var i in array) {  
        alert(array[i]);  
    }  
</script>
```

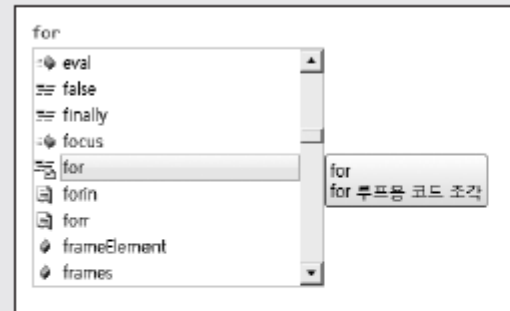


## 4.4 for in 반복문

### ❖ 코드 조각

- Visual Web Developer 2010 Express의 기능
- 조건문과 반복문을 손쉽게 빠르게 만들 수 있음

그림 4-4 for 루프용 코드 조각



for까지 입력한 후에 탭을 두 번 누르면 자동으로 다음과 같은 형태의 반복문이 만들어집니다.

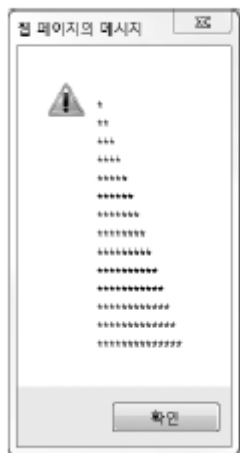
```
for (var i = 0; i < length; i++) {  
  
}
```



## ❖ 중첩 반복문이란?

- 반복문을 여러 겹 중첩해 사용하는 것
- 고전적 예제 - 그림 4-5의 피라미드

그림 4-5 피라미드 (1)



코드 4-9 만들어보세요.

```
<script>
    // 변수를 선언합니다.
    var output = '';

    // 반복문
    // 여기를 작성하세요.

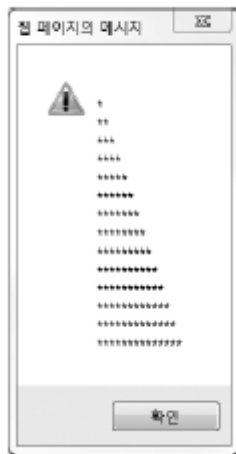
    // 출력합니다.
    alert(output);
</script>
```



## 4.5 중첩 반복문

### ❖ 피라미드 예제 (1)

그림 4-5 피라미드 (1)



코드 4-10 피라미드 (1)

```
<script>
    // 변수를 선언합니다.
    var output = '';

    // 반복문
    for (var i = 0; i < 10; i++) {
        for (var j = 0; j < i; j++) {
            output += '*';
        }
        output += '\n';
    }

    // 출력합니다.
    alert(output);
</script>
```





## 4.5 중첩 반복문

### ❖ 피라미드 예제 (2)

그림 4-6 피라미드 (2)



코드 4-11 피라미드 (2)

```
<script>
    // 변수를 선언합니다.
    var output = '';

    // 반복문
    for (var i = 0; i < 15; i++) {
        for (var j = 15; j > i; j--) {
            output += ' ';
        }

        for (var k = 0; k < 2 * i - 1; k++) {
            output += '*';
        }
        output += '\n';
    }

    // 출력합니다.
    alert(output);
</script>
```



### ❖ break 키워드

- switch 조건문이나 반복문을 벗어날 때 사용하는 키워드
- 다음 반복문은 조건이 항상 참이므로 무한 반복
  - 무한루프는 break 키워드를 사용해야 벗어날 수 있음
  - 코드 4-12]
    - 몇 번째 반복문인지 경고창을 출력한 후
    - 사용자에게 한 번 더 반복을 수행할지 물어봄

```
while (true) {  
  
}
```

코드 4-12 break 키워드

```
<script>  
  // 반복문  
  for (var i = 0; true; i++) {  
    alert(i + '번째 반복문입니다.');  
    // 진행여부를 물어봅니다.  
    if (!confirm('계속하시겠습니까?')) {  
      break;  
    }  
  }  
  alert('프로그램 종료');  
</script>
```

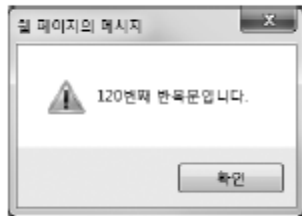


## 4.6 break 키워드

### ❖ break 키워드

- '확인'을 누르면 confirm() 함수가 true로 변환
  - 논리 부정 연산자를 사용해 false로 바꿔 반복문 계속 진행
- '취소'를 누르면 break 키워드가 실행돼 반복문 종료
- '확인'을 누른 뒤의 결과

그림 4-7 계속 진행됩니다.



## ❖ continue 키워드

- 반복문 내에서 반복을 멈추고 다음 반복을 진행시키는 키워드
  - 코드를 실행하면 경고창이 출력되지 않음
    - **continue 키워드를 만나면 바로 다음 반복으로 넘어가므로 alert() 함수를 실행하지 않음**

코드 4-13 continue 키워드

```
<script>
// 반복문
for (var i = 0; i < 5; i++) {
    // 현재 반복을 중지하고 다음 반복을 수행합니다.
    continue;
    alert(i); // (ㄱㄱ)
}
</script>
```



## 4.7 continue 키워드

### ❖ 예제 코드 4-14]

- 0부터 10까지의 짝수 합을 구하는 예제
  - 조건문을 사용해서 홀수일 때는 continue 키워드를 만나 바로 다음 반복으로 – 짝수의 합만 구해짐

코드 4-14 continue 키워드

```
<script>
  // 변수를 선언합니다.
  var output = 0;

  // 반복문
  for (var i = 1; i <= 10; i++) {
    // 조건문
    if (i % 2 == 1) {
      // 홀수면 현재 반복을 중지하고 다음 반복을 수행합니다.

      continue;
    }
    output += i;
  }

  // 출력합니다.
  alert(output);
</script>
```





# Thank You !

모던 웹을 위한 Javascript jQuery 입문