



조건문

1. if
2. switch

if

- if문은 조건이 참이냐, 거짓이냐에 따라 문장을 실행하거나 실행하지 않을 수 있음

```
<script>
if (조건) {
    // 실행문
}

if (4 < 5) {
    alert('참'); // if 조건이 참이므로 코드가 실행
}

</script>
```

if

- if문의 조건이 참일 때와 거짓일 때 수행할 문장을 각각 따로 정함

```
<script>
if (조건) {
    // 실행문1
} else {
    // 실행문2
}

if (2 > 9) {
    alert('참'); // if 조건이 거짓이므로 코드가 실행되지 않음
} else {
    alert('거짓'); // if 조건이 거짓이므로 else 코드가 실행됨
}
</script>
```

if

- 복수 조건을 검사할 경우에 사용

```
<script>
if (조건) {
    // 실행문1
} else if (조건) {
    // 실행문2
} else {
    // 실행문3
}
</script>
```

```
<script>
var a = 74;
if (a > 90) {
    alert('수');
} else if (a > 80) {
    alert('우');
} else {
    alert('낙제');
}
</script>
```

switch

- if 문의 복수 조건과 비슷

```
<script>
var a = 74;
switch (a) {
    case 10 :
        console.log('10');
        break;
    case 74 :
        console.log('74');
        break;
    case 97 :
        console.log('97');
        break;
    default :
        console.log('default');
        break;
}
</script>
```



반복문

1. **for**
2. **while**

for

- 비교조건에 따라서 반복되는 구문을 처리함

```
<script>
```

```
for (초기값; 비교조건; 초기값처리) {  
}
```

```
for (var a = 0; a < 10; a++) {  
    console.log(a);  
}
```

```
for (var b = 0, bmax = 10; b < bmax; b++) {  
    console.log(b);  
}
```

```
</script>
```

for

- 초기화랑 초기값처리는 () 바깥으로 선언 가능

```
<script>
var a = 0;
for (; a < 10; a++) {
    console.log(a);
}
```

```
var b = 0,
    bmax = 10;
for (; b < bmax; b++) {
    console.log(b);
}
</script>
```

```
<script>
var a = 0;
for (; a < 10;) {
    console.log(a);
    a++;
}
</script>
```

while

- 단순 반복문을 처리하기 적합

```
<script>  
var a = 0;  
while (a < 10) {  
    console.log(a);  
    a++;  
}  
</script>
```

while

```
<script>
var a = 0;
while (a < 10) {
    if (a > 5) {
        break;
    }
    console.log(a);
    a++;
}
</script>
```

```
<script>
var a = 0;
while (a < 10) {
    a++;
    if (a == 3 || a == 6) {
        continue;
    }
    console.log(a);
}
</script>
```

Thank you.

Question.