

04. 조건문

조건문을 사용하면 특정 조건이 만족됐을 때 특정 코드를 실행할 수 있습니다.

if 문

가장 기본적인 조건문은 if 문입니다.

if문은, "~~하다면 ~~를 해라" 를 의미합니다.

예시 코드를 한번 봅시다.

```
const a = 1;
if (a + 1 === 2) {
  console.log('a + 1 이 2 입니다.');
```

결과는, "a + 1 이 2 입니다." 이 출력됩니다.

하지만, 만약에 여기서 a 를 0 으로 바꾼다면 어떨까요?

```
const a = 0;
if (a + 1 === 2) {
  console.log('a + 1 이 2 입니다.');
```

결과는, 아무것도 출력되지 않습니다.

if문을 사용하면, 이렇게 특정 조건이 만족 될 때에만 특정 코드를 실행 시킬 수 있습니다.

```
if (조건) {
  코드;
}
```

조건이 만족됐을 때 실행시킬 코드가 { } 로 감싸져있는데요, 이를 코드 블록이라고 합니다.

만약에 조건이 true 가 된다면 우리가 지정한 코드가 실행되는 것이고, false 가 된다면 코드가 실행되지 않습니다.

우리가 이전에 let 과 const 를 배울 때, 다른 블록 범위에서는 똑같은 이름으로 선언 할 수도 있다고 배웠었습니다.

다음 코드를 한번 실행해보세요.

```
const a = 1;
if (true) {
  const a = 2;
  console.log('if문 안의 a 값은 ' + a);
}
console.log('if문 밖의 a 값은 ' + a);
```

위 코드에서는 if문에 조건을 true 로 설정했기 때문에 코드 블록 내부의 코드가 무조건 실행이 됩니다.

결과는 다음과 같습니다

```
"if문의 안의 a 값은 2"
"if문 밖의 a 값은 1"
```

if-else 문

if-else문은 "~~하다면 ~~하고, 그렇지 않다면 ~~해라." 를 의미합니다.

예시 코드를 볼까요?

```
const a = 10;
if (a > 15) {
  console.log('a 가 15 큼니다.');
```

```
} else {
  console.log('a 가 15보다 크지 않습니다.');
```

```
}
```

위 코드의 결과는 다음과 같습니다.

```
"a 가 10보다 크지 않습니다."
```

만약에 특정 조건이 만족할 때와 만족하지 않을 때 서로 다른 코드를 실행해야 된다면 if-else 구문을 사용 할 수 있습니다.

if-else if 문

if-else if 문은 여러 조건에 따라 다른 작업을 해야 할 때 사용합니다.

예시 코드를 따라 적어보세요.

```
const a = 10;
if (a === 5) {
  console.log('5입니다!');
```

```
} else if (a === 10) {
  console.log('10입니다!');
```

```
} else {
  console.log('5도 아니고 10도 아닙니다.');
```

```
}
```

결과는 다음과 같습니다.

```
"10입니다!"
```

한번 a 값을 5로 바꾸고 실행해보고, 7 로 바꾸고 실행해보세요.

```
a = 5 -> "5입니다!"
a = 7 -> "5도 아니고 10도 아닙니다."
```

switch/case 문

switch/case 문은 특정 값이 무엇이냐에 따라 다른 작업을 하고 싶을 때 사용합니다.

다음 예시 코드를 실행해보세요.

```
const device = 'iphone';

switch (device) {
  case 'iphone':
    console.log('아이폰!');
```

```
    break;
  case 'ipad':
    console.log('아이패드!');
```

```
    break;
  case 'galaxy note':
    console.log('갤럭시 노트!');
```

```
    break;
  default:
    console.log('모르겠네요..');
```

```
}
```

device 값을 'iphone', 'ipad', 'galaxy note', 'macbook' 으로 순서대로 바꿔가면서 코드를 실행해보세요.

device 값에 따라서 다른 결과가 출력되고 있나요?

switch/case 문은 이와 같이 특정 값이 무엇이냐에 따라 다른 작업을 수행 할 수 있게 해줍니다.

switch/case 문에서는 각 case 에서 실행할 코드를 작성하고 맨 마지막에 break; 를 해주어야 합니다. break 를 하지 않으면 그 다음 case 의 코드까지 실행해버립니다.

그리고, 맨 아래의 default: 는 device 값이 우리가 case 로 준비하지 않은 값일 경우를 의미합니다.