

## 자바스크립트 예외처리

자바스크립트에서 예외 처리는 try...catch 문을 사용하여 수행됩니다.

블록	설명
try	예외가 발생할 수 있는 코드를 포함합니다. 예외가 발생하면 try 블록의 나머지 코드는 실행되지 않고, 제어기가 catch 블록으로 이동합니다.
catch	try 블록에서 예외가 발생했을 때 실행됩니다. 발생한 예외에 대한 정보를 담고 있는 오류 객체를 인자로 받을 수 있으며, 이를 통해 오류를 처리합니다.
finally (선택적)	예외 발생 여부와 관계없이 try 블록 이후에 항상 실행됩니다. 자원을 해제하거나 정리하는 작업 등을 수행하기에 적합합니다.

```
try {
  // 예외가 발생할 수 있는 코드
  const num = 1;
  const result = num / 0;          // 예시를 위한 연산
  if (result === Infinity) {
    throw new Error("무한대 값 발생"); // 예외 발생
  }
} catch (error) {                  // 예외 처리 코드
  console.error(error.message);    // 에러 메시지 출력
} finally {                       // 마무리 작업 코드
  console.log("예외 처리 완료");
}
```

이 예제에서는 무한대 값을 초래하는 연산 후 예외를 throw하고, catch 블록에서 이를 처리하고 있습니다. finally 블록에서는 예외 처리가 완료되었음을 알리는 로그를 출력합니다. 예외 처리는 코드의 안정성과 가독성을 높이는 데 도움을 줍니다.

```
function findElement(arr, element) {
  try {
    const index = arr.indexOf(element);
    if (index === -1) {
      throw new Error("해당 요소를 찾을 수 없습니다.");
    }
    return index;
  } catch (e) {
    console.error(e.message);
    return -1; // 예외 발생 시 -1 반환
  }
}

// 사용 예시
const fruits = ['사과', '바나나', '오렌지'];
console.log(findElement(fruits, '바나나')); // 1 반환 (바나나의 인덱스)
console.log(findElement(fruits, '체리'));   // 에러 메시지 출력 및 -1 반환
```

