

# JavaScript (ES6) For Framework

18. 예외처리 (try catch finally)



예외 (Exception)



예기치 못한 상황



“

예측하고 대비하면 프로그램 멈출 필요 없다

예외 상황

1부터 10까지의 수를 입력하세요!



예외 상황

1부터 10까지의 수를 입력하세요!

11

앗

## Exception (Error) in JavaScript

- `SyntaxError` - 문법 등 자바스크립트 파싱 중 에러
- `TypeError` - 사용할 수 없는 변수를 썼을 때
- `ReferenceError` - 허용되지 않은 참조 사용
- `RangeError` - 허용 숫자 범위 초과
- `EvalError` - `eval()`의 잘못된 활용
- `URIError` - `encodeURIComponent()` 혹은 `decodeURI()`의 잘못된 활용
- 사용자 정의 에러
- 등등

## Try catch finally

```
try {
```

일단 이 부분 실행

```
}
```

```
catch {
```

에러 생기면 이 부분 실행


```
}
```

```
finally {
```

마침내 이 부분 실행

```
}
```

예외상황 발생





## Try catch finally

```
try {
```

예외상황이 생길 수 있는 코드

```
}
```

```
catch {
```

```
}
```

```
finally {
```

마지막으로 무조건 실행될 코드

```
}
```

예외없으면



## Try catch finally

```
try {
```

예외상황이 생길 수 있는 코드

```
}
```

```
catch {
```

예외를 처리하는 코드

```
}
```

```
finally {
```

마지막으로 무조건 실행될 코드

```
}
```

예외상황 발생

= 에러 객체 생성

다음강의

## Section 18-2

예외 실습

Lecturer Github : <https://github.com/kangtegong/>

수업자료링크 : <https://github.com/kangtegong/develop-javascript>

Email : [tegongkang@gmail.com](mailto:tegongkang@gmail.com)