

# React 테스트팅 : 기초

# 테스트 코드를 작성하는 이유

# 작성하는 이유

01

## 코드의 안정성

각 기능이 의도한 대로 작동하는지 확인

02

## 리팩토링의 안정성

테스트 코드가 있다면 리팩토링 후 기존  
기능이 깨지지 않았음을 확인

03

## 코드의 문서화

코드의 동작을 명확하게 설명하는 문서

# 테스트 종류

## 유닛 테스트

- 단일 컴포넌트나 함수의 동작을 검증하는데 중점

## E2E 테스트

- 애플리케이션의 전체 흐름을 사용자 관점에서 테스트

# 프로젝트 세팅

# 프로젝트 세팅

```
npm create vite@latest
```

```
npm install -D vitest @testing-library/react @testing-library/jest-dom jsdom
```

```
Vitest 2.0.5 @testing-library/react 16.0.0 @testing-library/jest-dom 6.4.8 jsdom 24.1.1
```

# 테스트 코드 작성

# 테스트 프레임워크 함수

describe: 테스트 케이스들을 하나로 그룹화

it: 개별 테스트 케이스 정의

expect: 테스트 어설션 정의

```
describe('Test case', () => {  
  it('case 1', () => {  
    const result = 1 + 2;  
    expect(result).toBe(3);  
  })  
})
```



# 쿼리 함수 종류

01

## getBy

요소가 반드시 있어야 할 때

getText,  
getByRole, getLabelText, ...

02

## queryBy

요소가 없어도 될 때

queryByText,  
queryByRole, queryLabelText, ...

03

## findBy

비동기를 기다릴 필요가 있을 때

findByText,  
findByRole, findLabelText, ...