

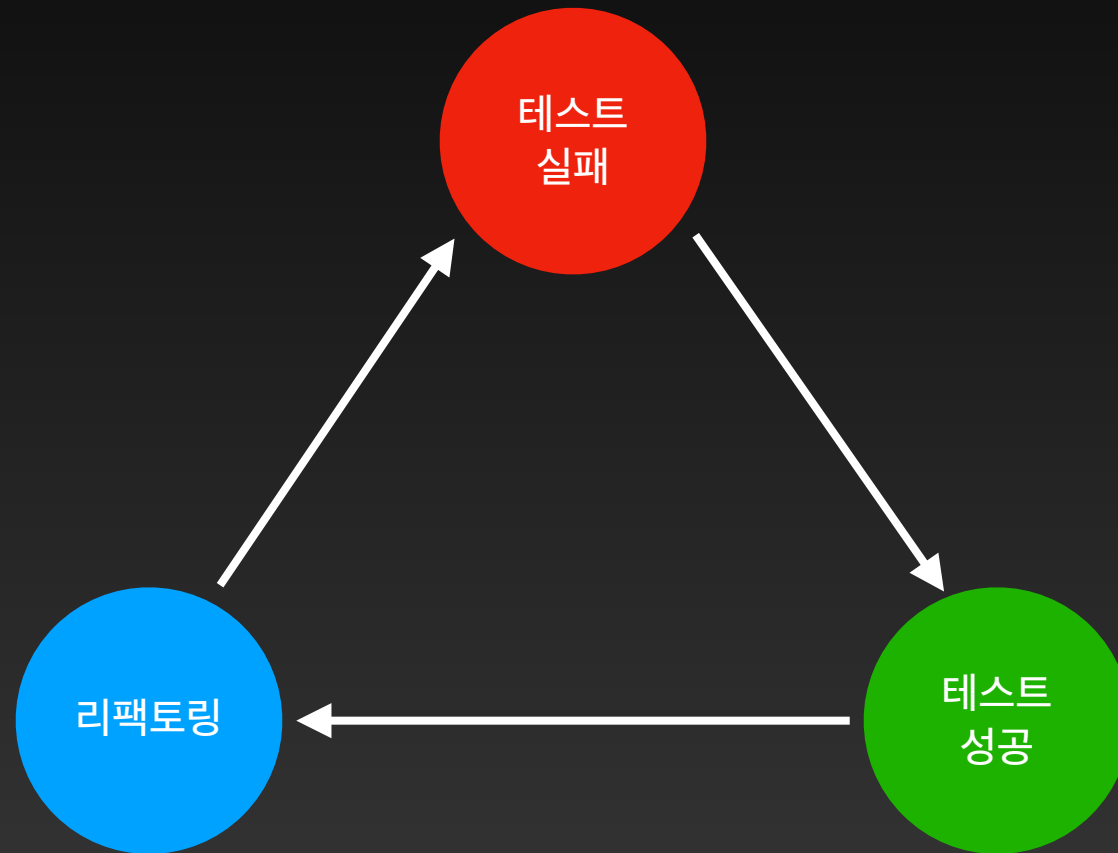
# 테스트 주도 개발(TDD)은 무엇일까?

실무에 바로 적용하는 프론트엔드 테스트

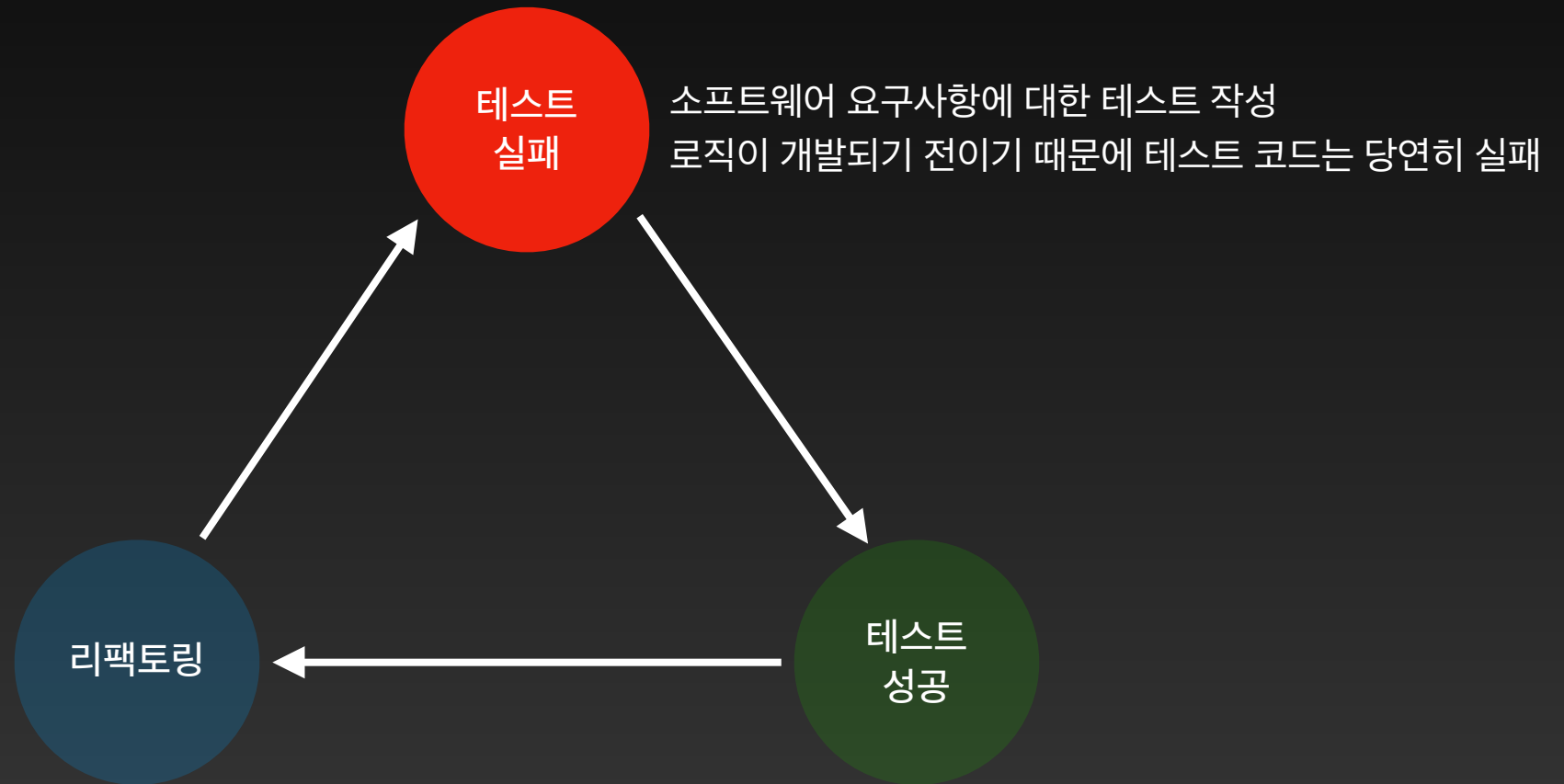
# 테스트 주도 개발이란?

소프트웨어가 완전히 개발되기 전  
요구사항을 테스트 케이스로 작성하고  
테스트 케이스에 대해 테스트 작성 후 검증  
그리고 리팩토링 단계를 반복하며 개발하는 방법론

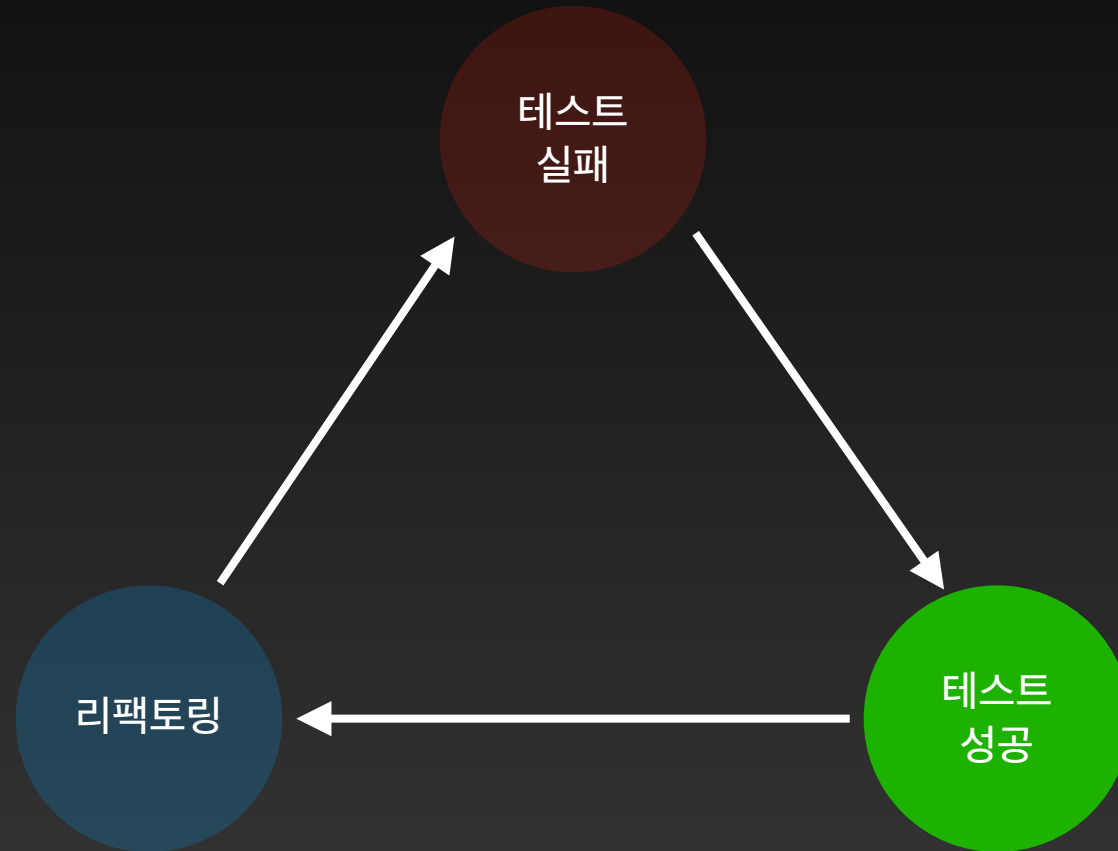
# 테스트 주도 개발이란?



# 테스트 주도 개발이란?

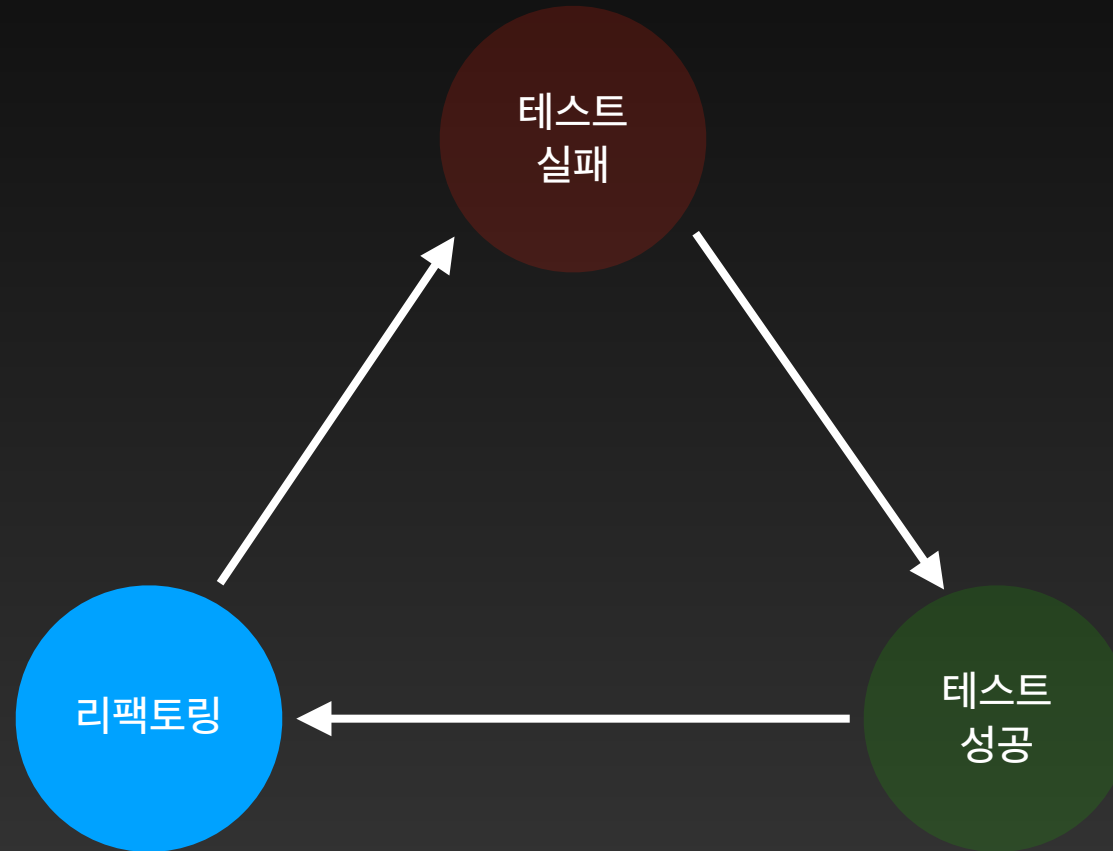


# 테스트 주도 개발이란?



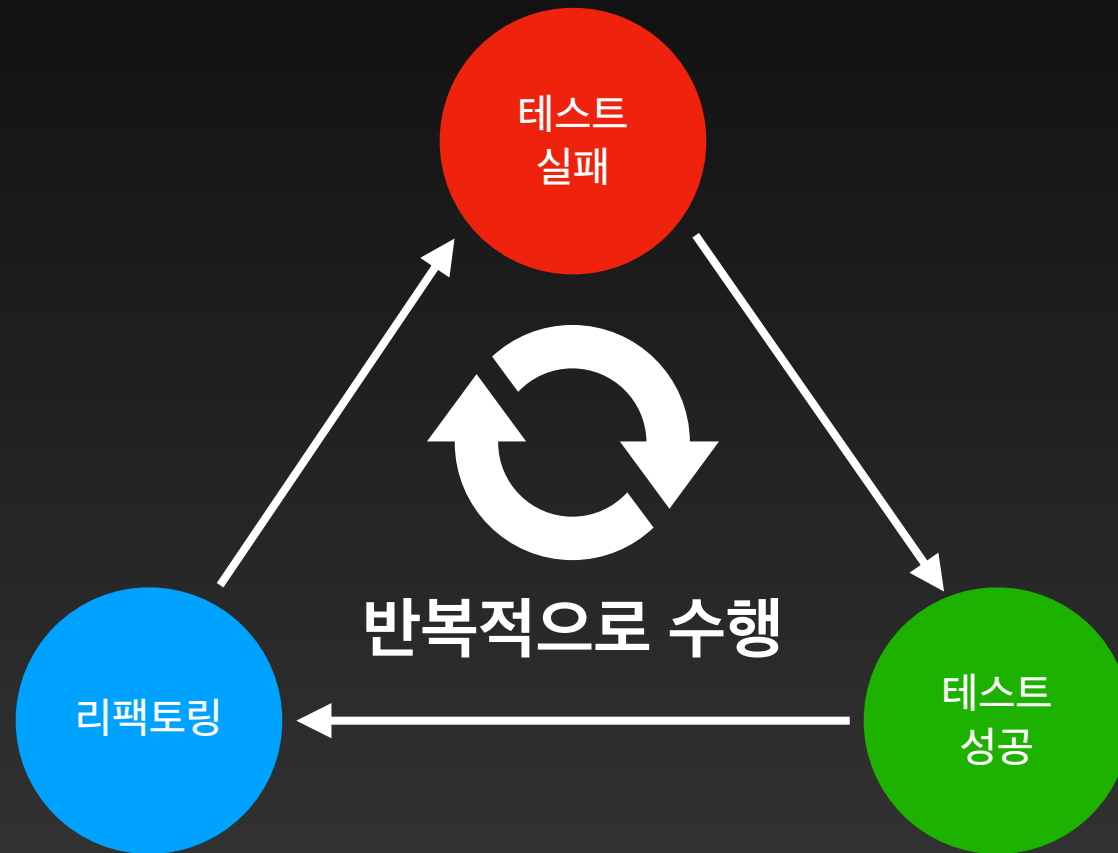
실제 코드를 작성해 테스트가 성공하도록 함  
테스트 된 기능 외에는 코드를 작성해선 안됨

# 테스트 주도 개발이란?



테스트가 모두 통과된 상태에서 코드를 정리하는 단계  
가독성, 성능에 관한 개선이나 코드를 분리하는 등의 작업을 수행

# 테스트 주도 개발이란?



# 테스트 주도 개발의 장점

- ✓ 기능 단위로 테스트를 진행하기 때문에 정식 QA(인수 테스트)를 하기 전 개발 단계에서 버그의 원인을 찾고 수정할 수 있다.
- ✓ 테스트를 통해 개발 코드를 지속적으로 검증하기 때문에 안정성있게 작업할 수 있다.
- ✓ 테스트 코드를 기반으로 안정성있게 리팩토링을 진행할 수 있다.
- ✓ 리팩토링 단계에서 효율적인 테스트 단위나 가독성 등을 고민하다보면 자연스럽게 좋은 설계에 대한 사고로 이어진다.
- ✓ 잘 작성된 테스트 코드를 통해 앱의 기능 명세를 이해하는 데 도움을 준다.



# 테스트 주도 개발의 장점

- ✓ 기능 단위로 테스트를 진행하기 때문에 정식 QA(인수 테스트)를 하기 전 개발 단계에서 버그의 원인을 찾고 수정할 수 있다.
- ✓ 테스트를 통해 개발 코드를 지속적으로 검증하기 때문에 안정성있게 작업할 수 있다.
- ✓ 테스트 코드를 기반으로 안정성있게 리팩토링을 진행할 수 있다.
- ✓ 리팩토링 단계에서 효율적인 테스트 단위나 가독성 등을 고민하다보면 자연스럽게 좋은 설계에 대한 사고로 이어진다.
- ✓ 잘 작성된 테스트 코드를 통해 앱의 기능 명세를 이해하는 데 도움을 준다.
- ✓ 초기 개발 비용이 크지만, 장기적으로 효과적이고 안정적이게 앱을 유지할 수 있다.

# 테스트를 작성한다면 반드시 TDD를 도입해야 할까?

🤔 모든 프로젝트에는 한정된 리소스와 정해진 기한이 존재  
현실적인 부분에 있어 TDD를 도입하는 것이 무리일 수 있음

🤔 UI만 보여주는 특별한 기능이 없는 앱일수도 있음

🤔 스타일 검증처럼 TDD 적용이 불가능한 부분이 있을수도 있음

# 테스트를 작성한다면 반드시 TDD를 도입해야 할까?

🤔 모든 프로젝트에는 한정된 리소스와 정해진 기한이 존재  
현실적인 부분에 있어 TDD를 도입하는 것이 무리일 수 있음

🤔 UI만 보여주는 특별한 기능이 없는 앱일수도 있음

🤔 스타일 검증처럼 TDD 적용이 불가능한 부분이 있을수도 있음

🔍 개발 후 중요한 워크플로우에 대해서만

E2E 테스트를 작성하는 등 현실적인 테스트 작성법을 찾아야 함

# 정리

## 테스트 주도 개발은

- 개발 코드를 작성하기 전에 요구 사항을 테스트 케이스로 먼저 작성한 뒤, 실제 기능을 추가하고 리팩토링하는 과정을 반복하며 지속 검증하며 개발하는 방법론
- 테스트 실패 ➡ 테스트 성공 ➡ 리팩토링 사이클로 개발 진행

## 테스트 주도 개발의 장점

- 개발 단계에서 버그의 원인을 찾고 수정할 수 있다.
- 지속적으로 테스트를 통해 검증하기 때문에 안정성있는 작업을 할 수 있다.
- 효율적인 테스트 단위나 코드의 가독성 등 여러 방면을 고민하게 되어 자연스럽게 좋은 설계에 대한 사고로 이어진다.
- 초기 테스트 작성 비용이 많이 들지만, 앱의 장기적인 관점에서 봤을 때 효과적이다.

하지만, 테스트를 작성한다고 반드시 TDD를 도입할 필요는 없다.

- 현실적인 리소스 문제가 있다면 일부 중요한 기능의 단위 테스트만 작성하거나 개발 후에 중요한 특정 워크플로우에 만 E2E 테스트를 적용하는 등 앱에 맞는 현실적인 테스트 작성법을 찾아야 한다.