



OBJECT 2



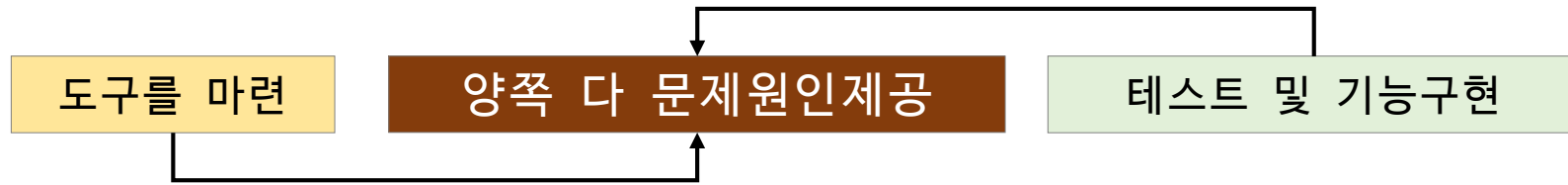
테스트 주도 주기 시작

최초 기능의 역설 및 동작하는 골격

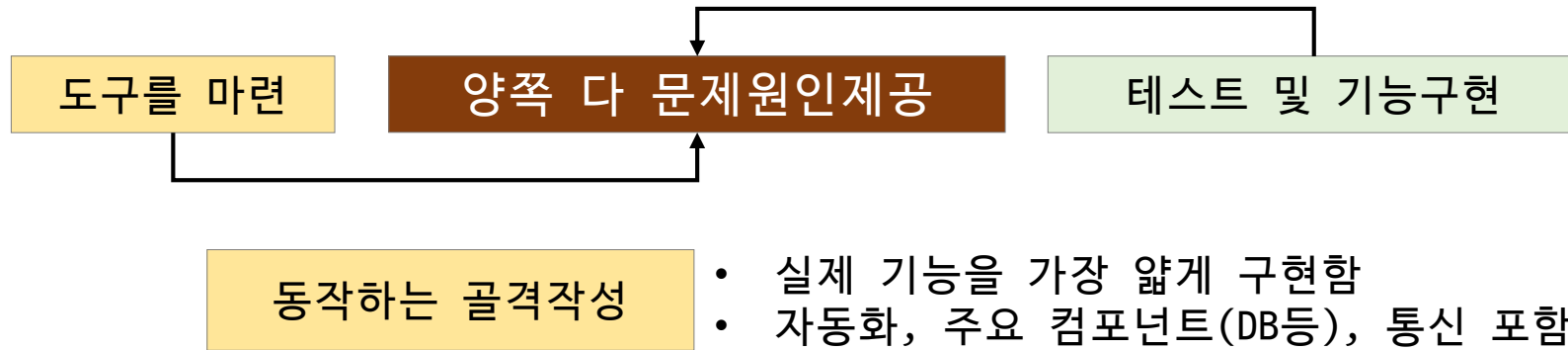
도구를 마련

테스트 및 기능구현

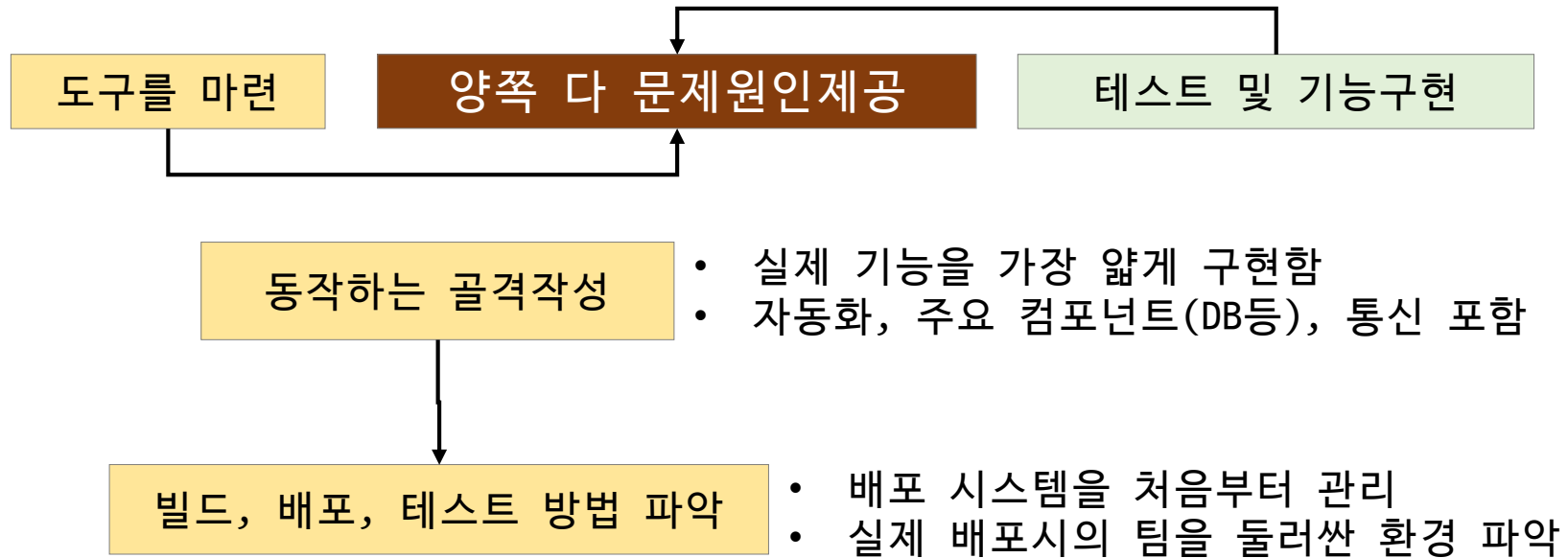
최초 기능의 역설 및 동작하는 골격



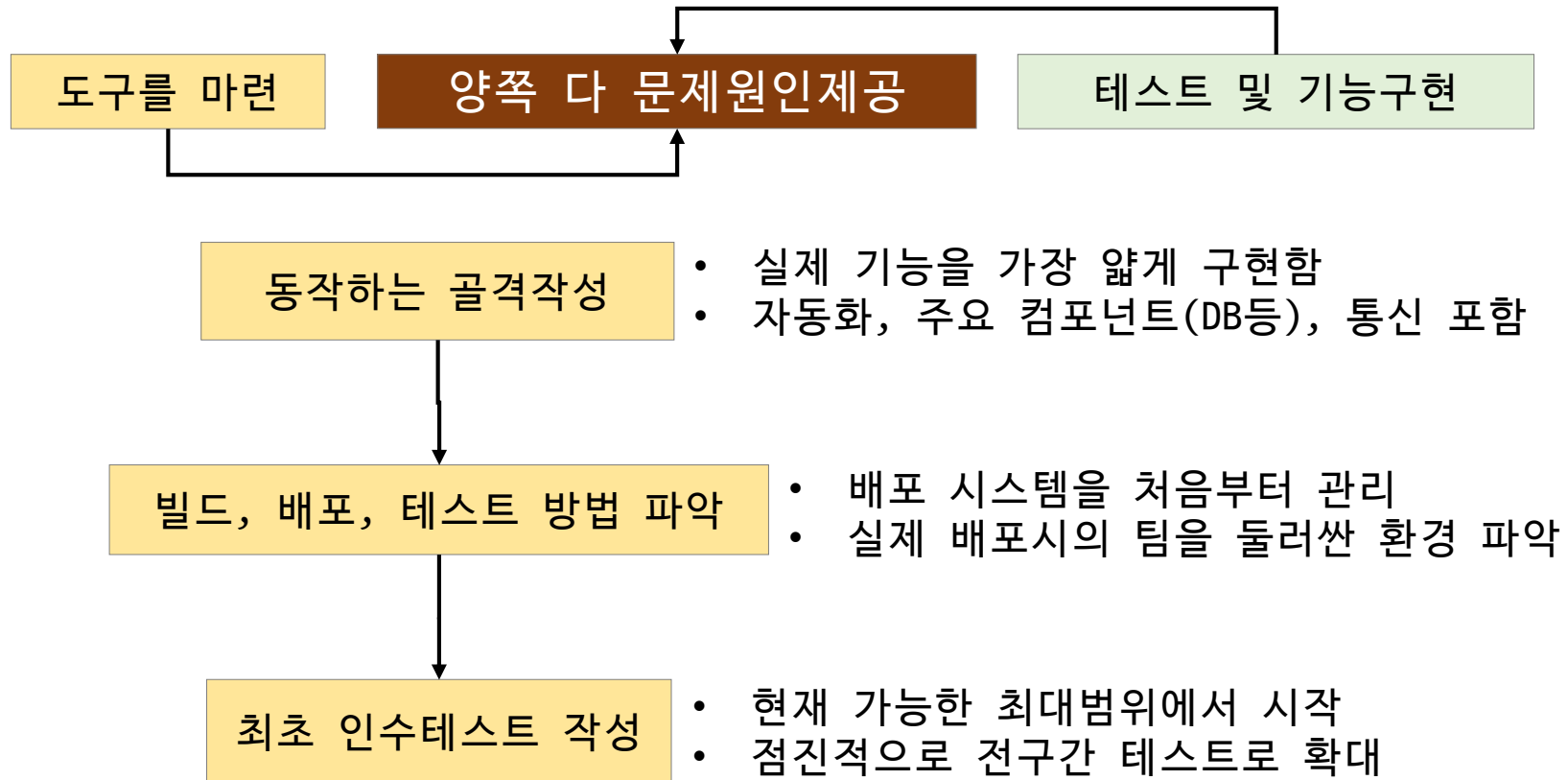
최초 기능의 역설 및 동작하는 골격



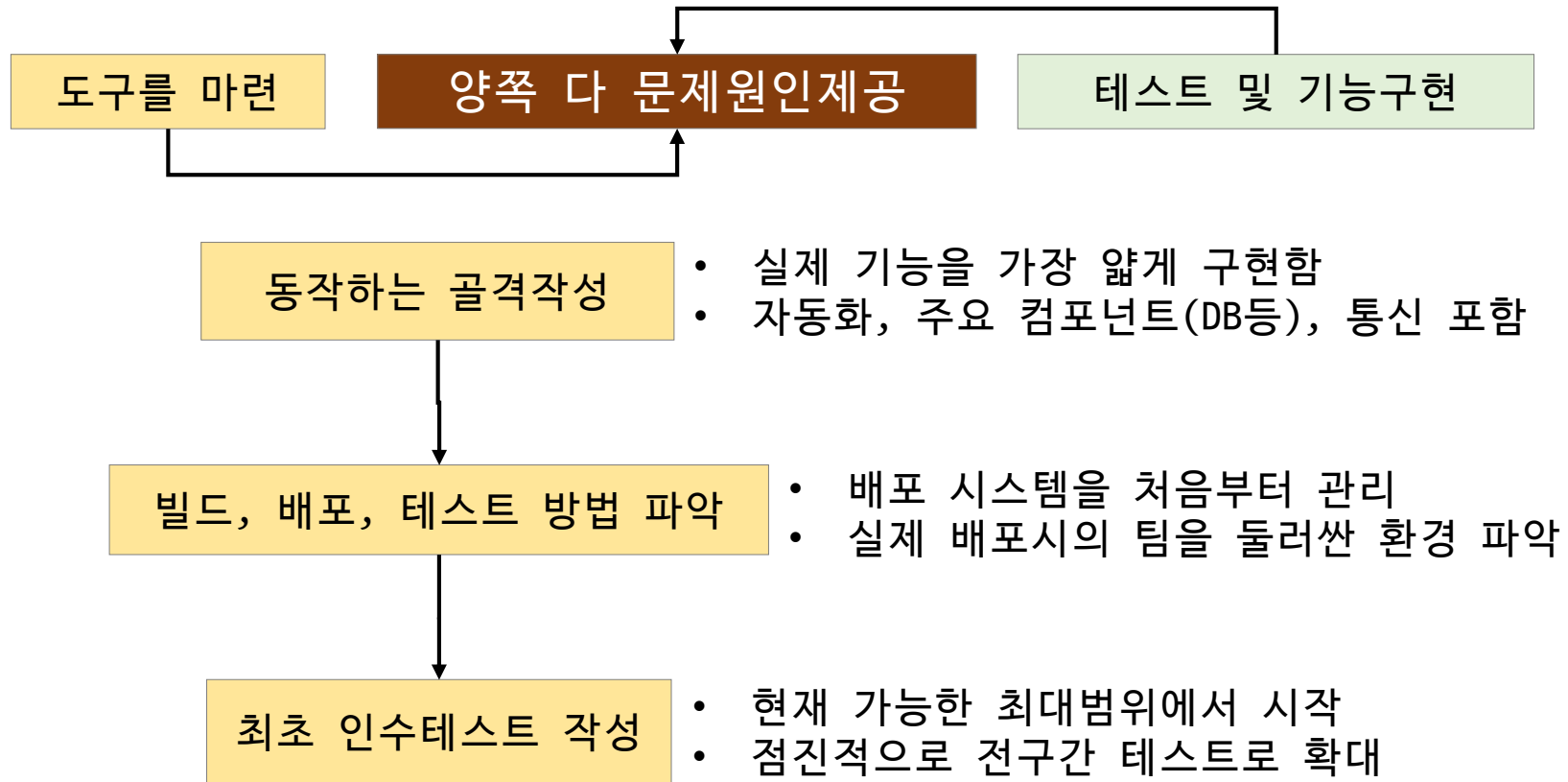
최초 기능의 역설 및 동작하는 골격



최초 기능의 역설 및 동작하는 골격



최초 기능의 역설 및 동작하는 골격



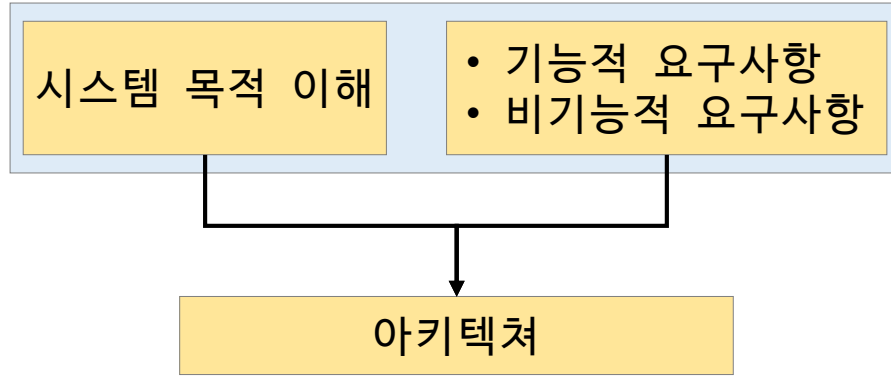
테스트의 명확성 보다 골격 구조에 집중

동작하는 골격과 불확실성

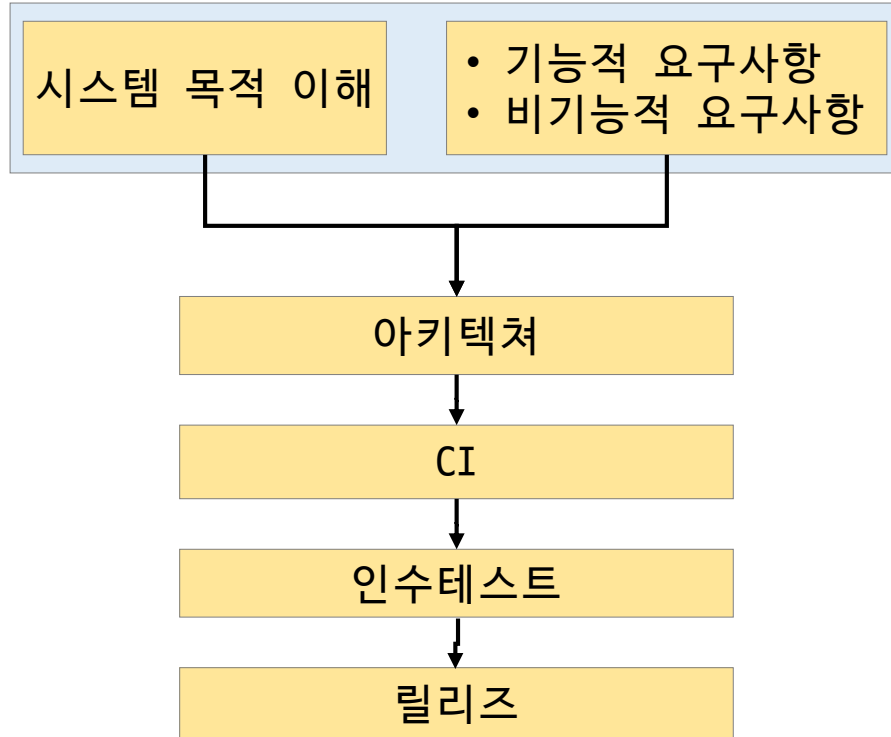
시스템 목적 이해

- 기능적 요구사항
- 비기능적 요구사항

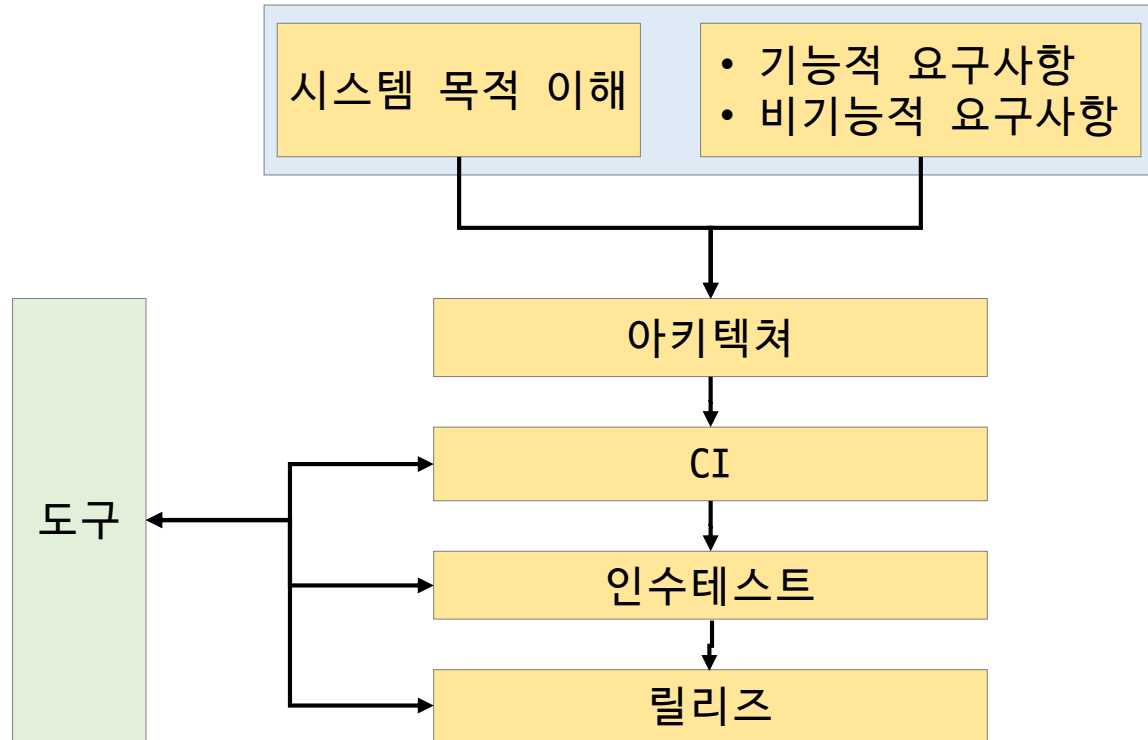
동작하는 골격과 불확실성



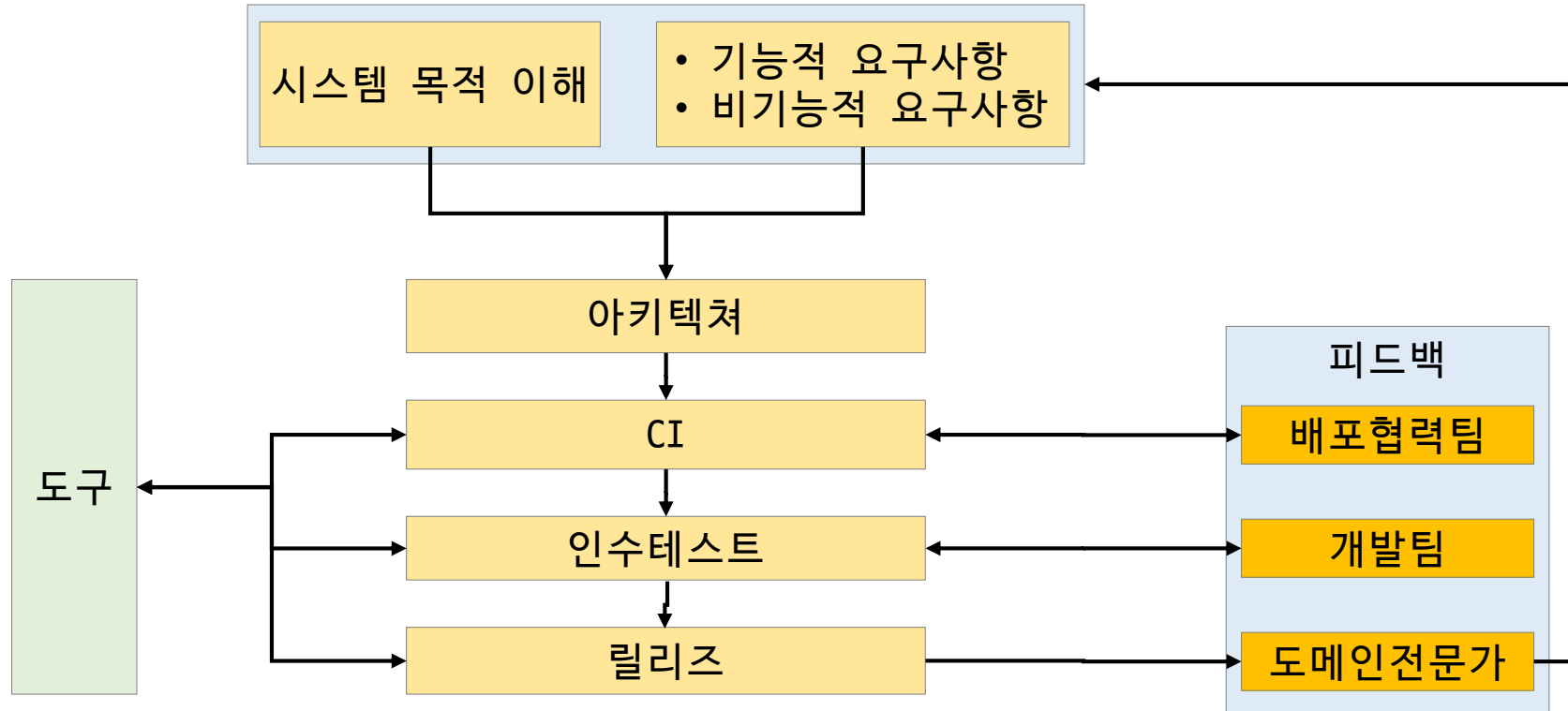
동작하는 골격과 불확실성



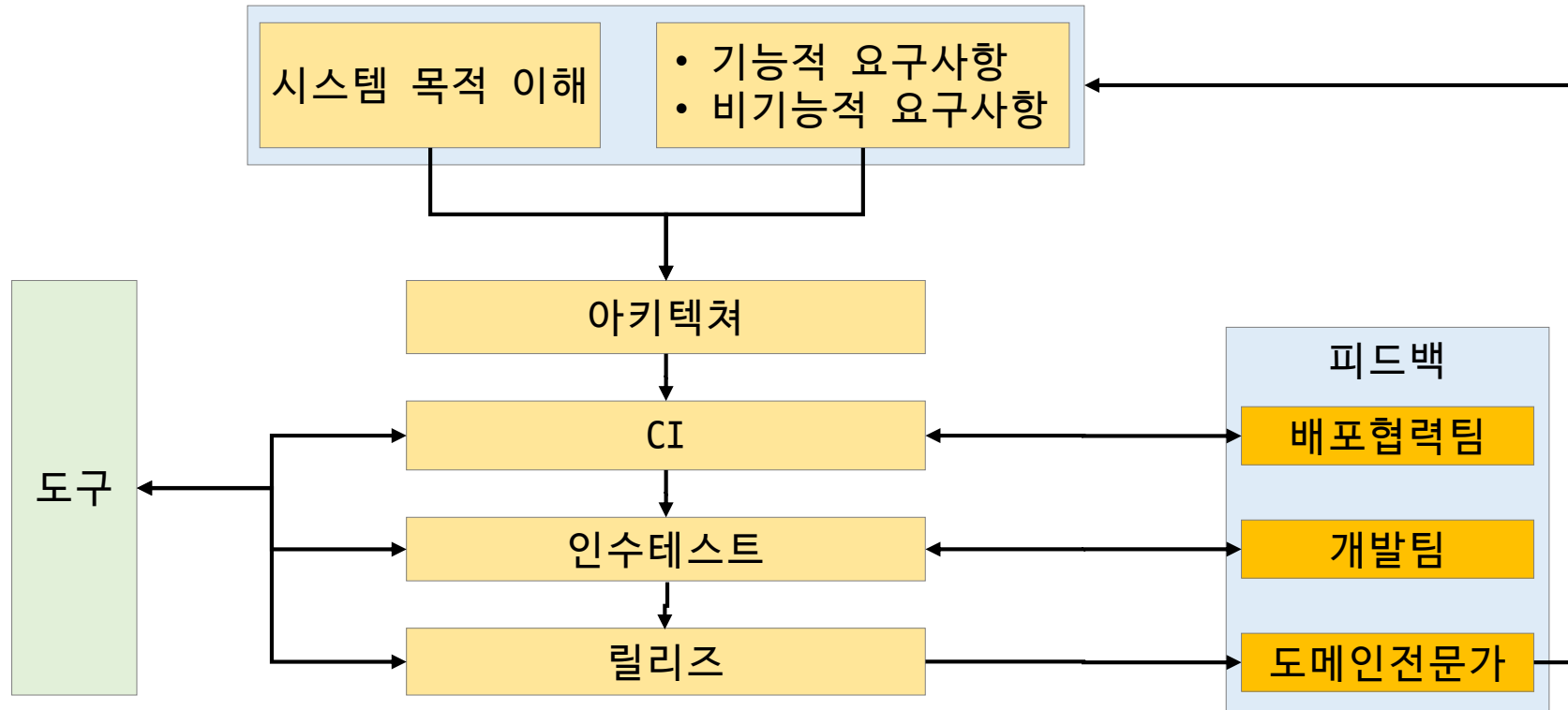
동작하는 골격과 불확실성



동작하는 골격과 불확실성



동작하는 골격과 불확실성

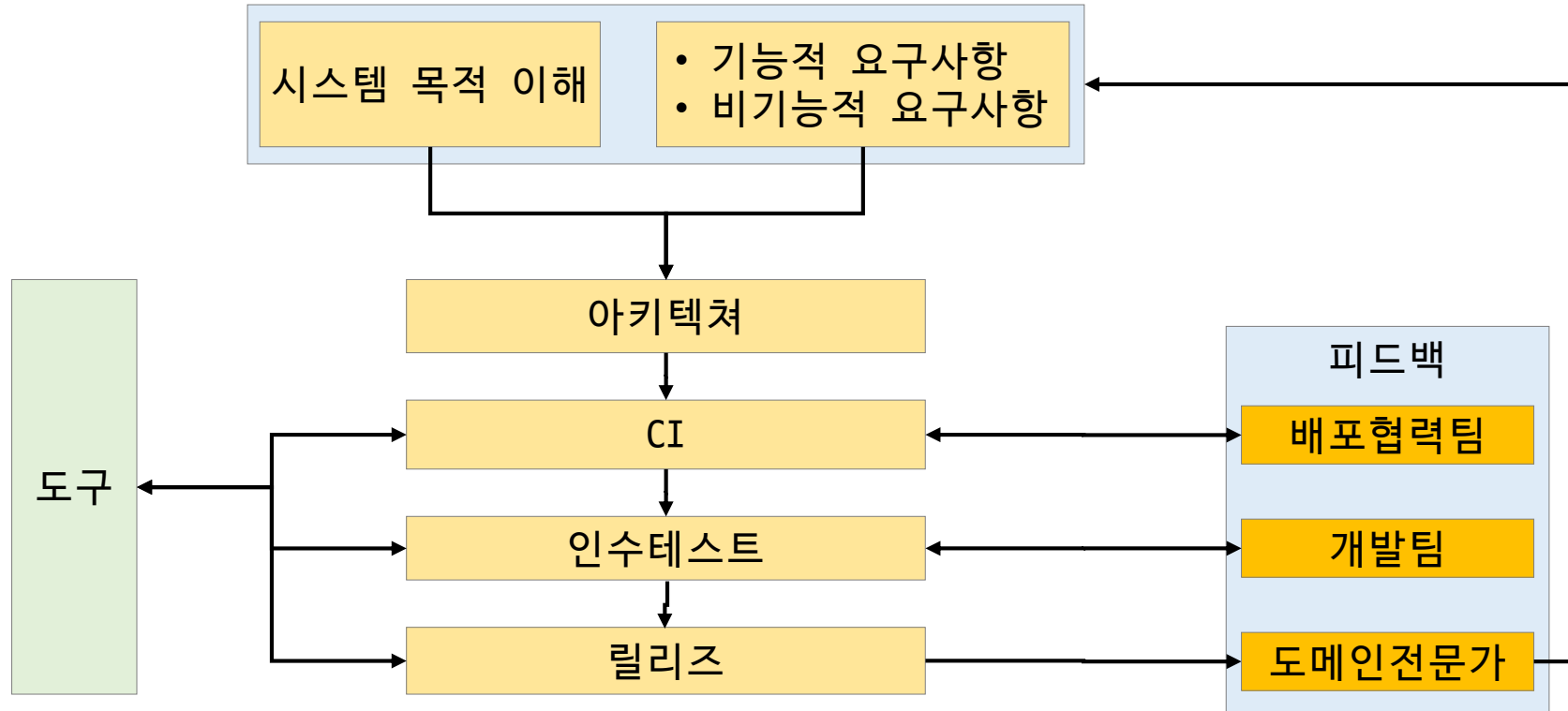


개발 외적 요소

수 많은 이해관계자

수 많은 절차와 규정

동작하는 골격과 불확실성



개발 외적 요소

수 많은 이해관계자

수 많은 절차와 규정

실제 기능구현 일정 예측

테스트 주도 개발 주기의 유지

테스트 운영요령

진행상황테스트

개발중

개발중

개발중

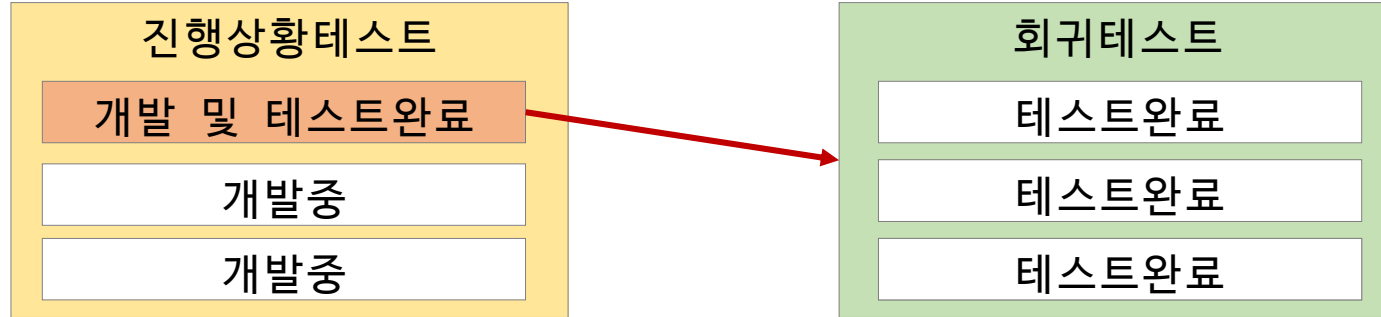
회귀테스트

테스트완료

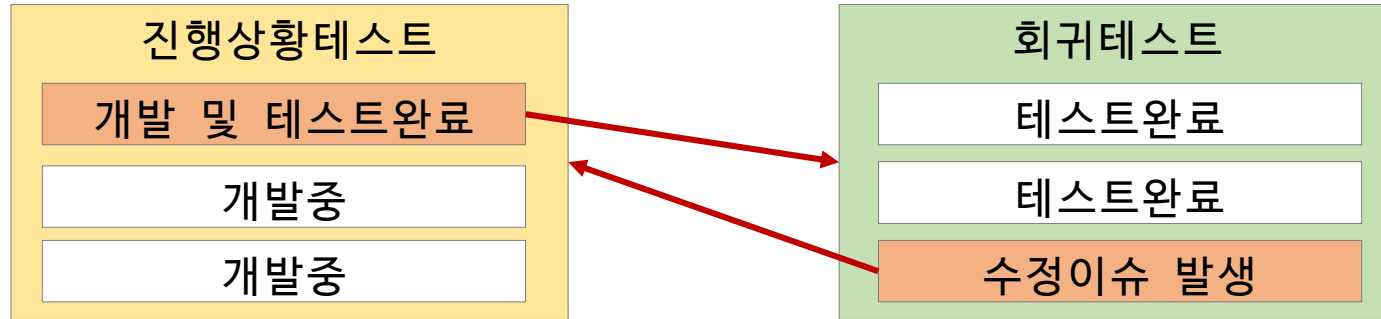
테스트완료

테스트완료

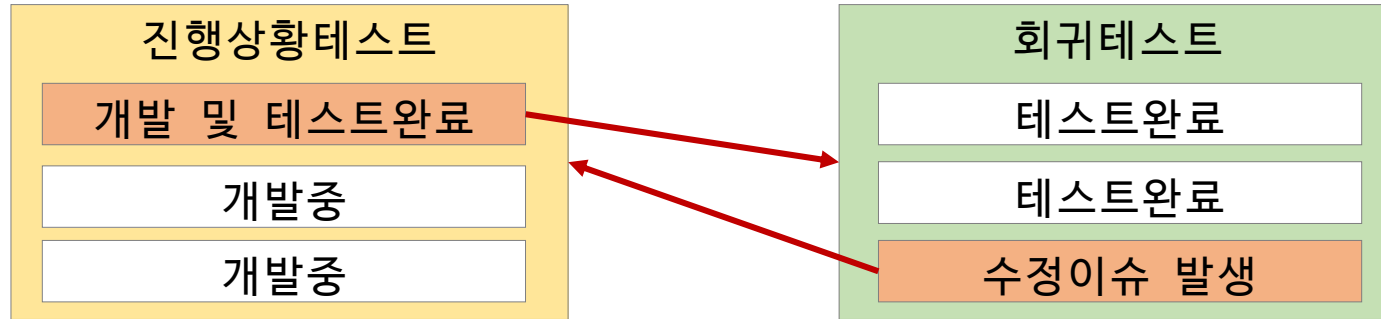
테스트 운영요령



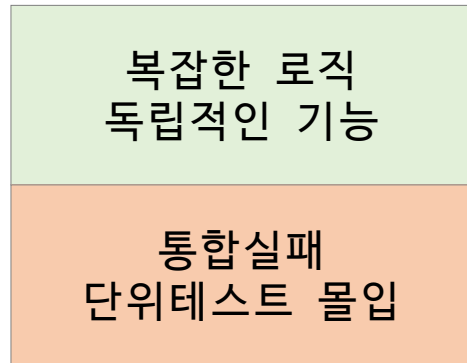
테스트 운영요령



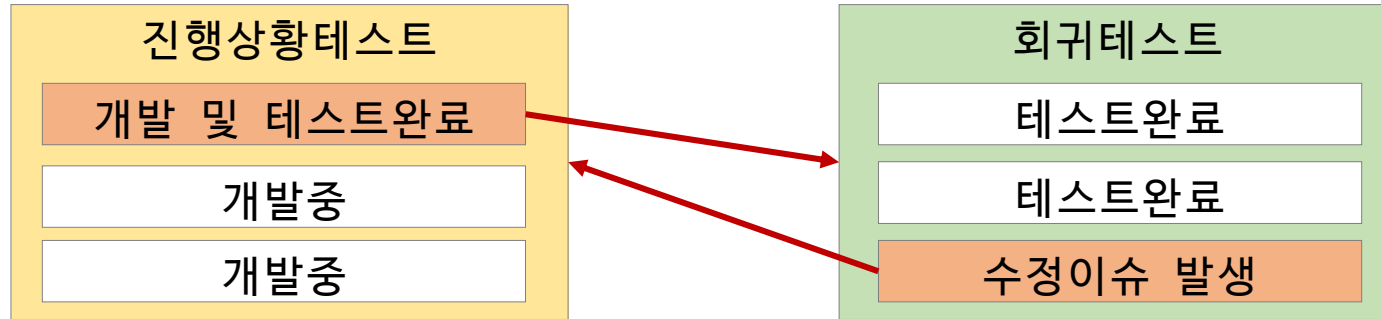
테스트 운영요령



단위테스트

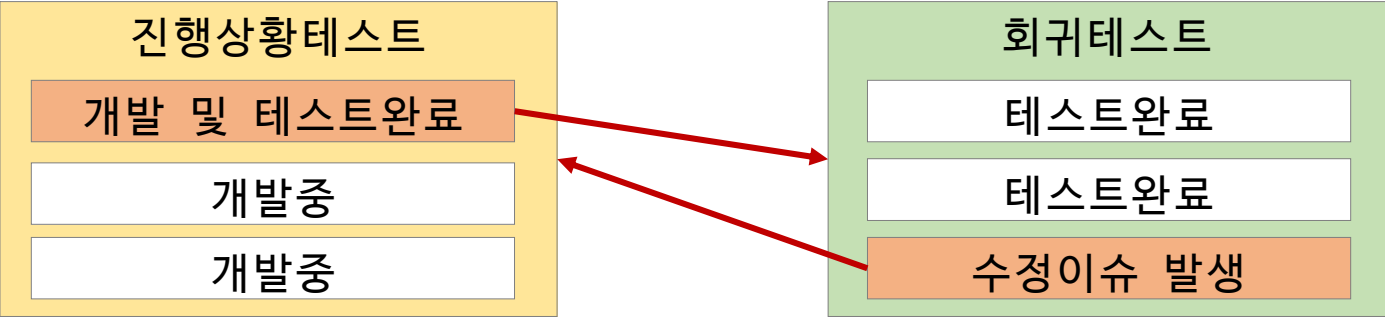


테스트 운영요령



단위테스트	통합테스트
복잡한 로직 독립적인 기능	다양한 예외 서드파티문제
통합실패 단위테스트 몰입	조합폭발 광범위한 원인조사

테스트 운영요령



단위테스트	통합테스트	인수테스트
복잡한 로직 독립적인 기능	다양한 예외 서드파티문제	시스템실패 컨텍스트오류조사
통합실패 단위테스트 몰입	조합폭발 광범위한 원인조사	더 큰 조합폭발 더 광범위한 원인조사

코드 작성요령

모든 문제를 해결하는 코드

코드 작성요령

모든 문제를 해결하는 코드



실제 필요한 기능테스트

코드 작성요령

모든 문제를 해결하는 코드



실제 필요한 기능테스트



필요한 문제를 해결하는 코드

테스트 작성요령

Right bicep 오른쪽 이두박근

Right - 올바른가

Boundary - 경계조건통과

Inverse - 역관계확인

Crosscheck - 교차확인

Error condition -에러조건 강제 생성

Performance -성능허용범위

테스트 작성요령

Right bicep 오른쪽 이두박근

Right - 올바른가

Boundary - 경계조건통과

Inverse - 역관계확인

Crosscheck - 교차확인

Error condition -에러조건 강제 생성

Performance -성능허용범위



행위를 테스트
책임과 역할을 드러낼 수 있는 테스트

테스트 작성요령

Right bicep 오른쪽 이두박근

Right - 올바른가

Boundary - 경계조건통과

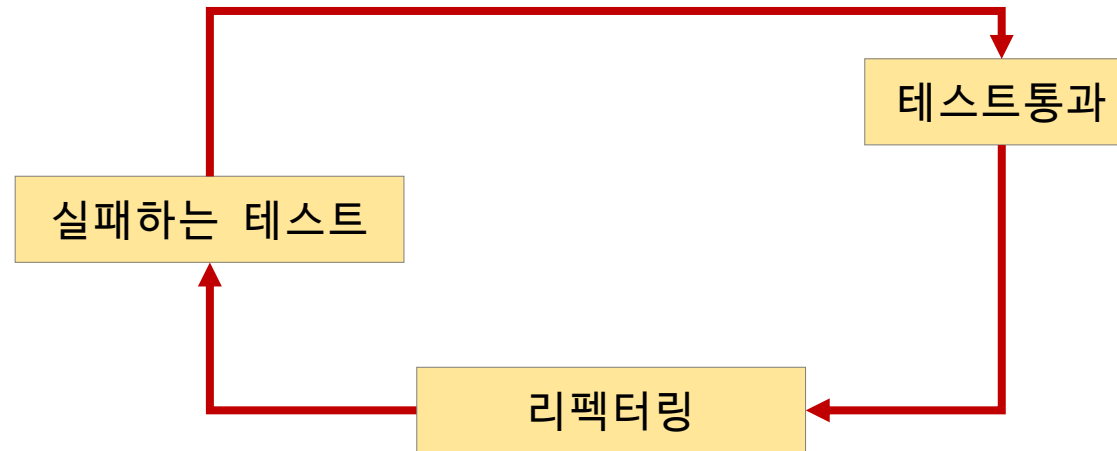
Inverse - 역관계확인

Crosscheck - 교차확인

Error condition -에러조건 강제 생성

Performance -성능허용범위

행위를 테스트
책임과 역할을 드러낼 수 있는 테스트



테스트 작성요령

Right bicep 오른쪽 이두박근

Right - 올바른가

Boundary - 경계조건통과

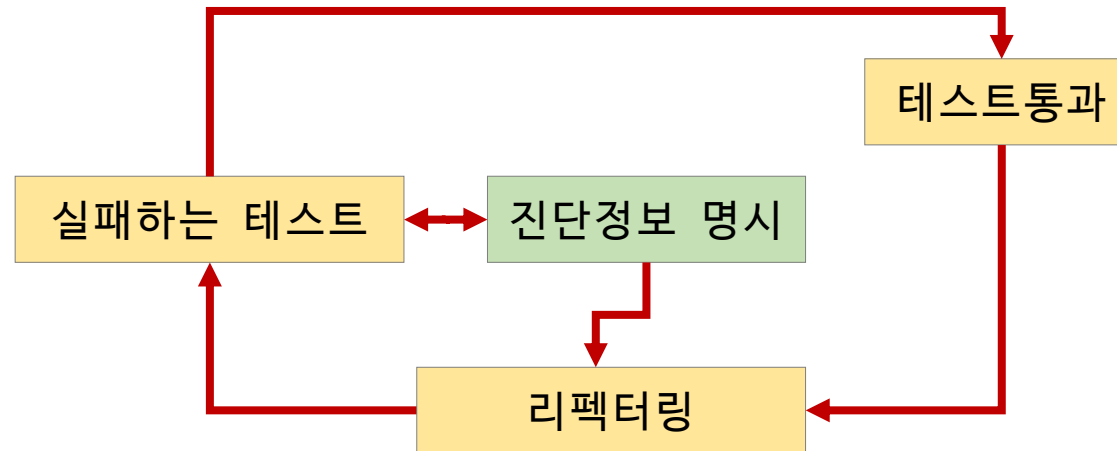
Inverse - 역관계확인

Crosscheck - 교차확인

Error condition -에러조건 강제 생성

Performance -성능허용범위

행위를 테스트
책임과 역할을 드러낼 수 있는 테스트



테스트 작성요령

Right bicep 오른쪽 이두박근

Right - 올바른가

Boundary - 경계조건통과

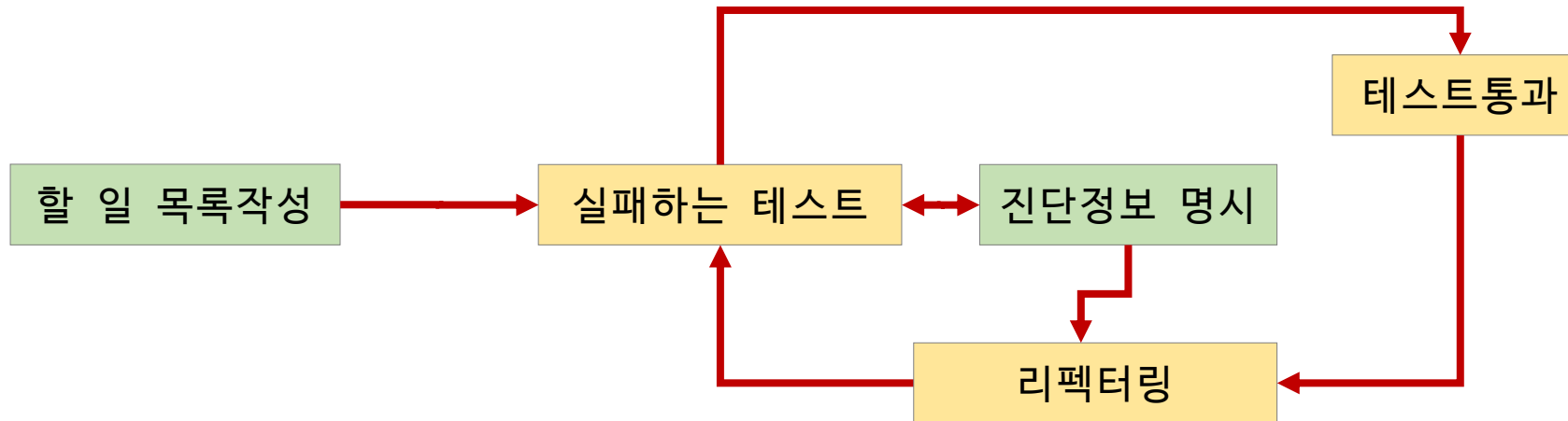
Inverse - 역관계확인

Crosscheck - 교차확인

Error condition -에러조건 강제 생성

Performance -성능허용범위

행위를 테스트
책임과 역할을 드러낼 수 있는 테스트



테스트 작성요령

Right bicep 오른쪽 이두박근

Right - 올바른가

Boundary - 경계조건통과

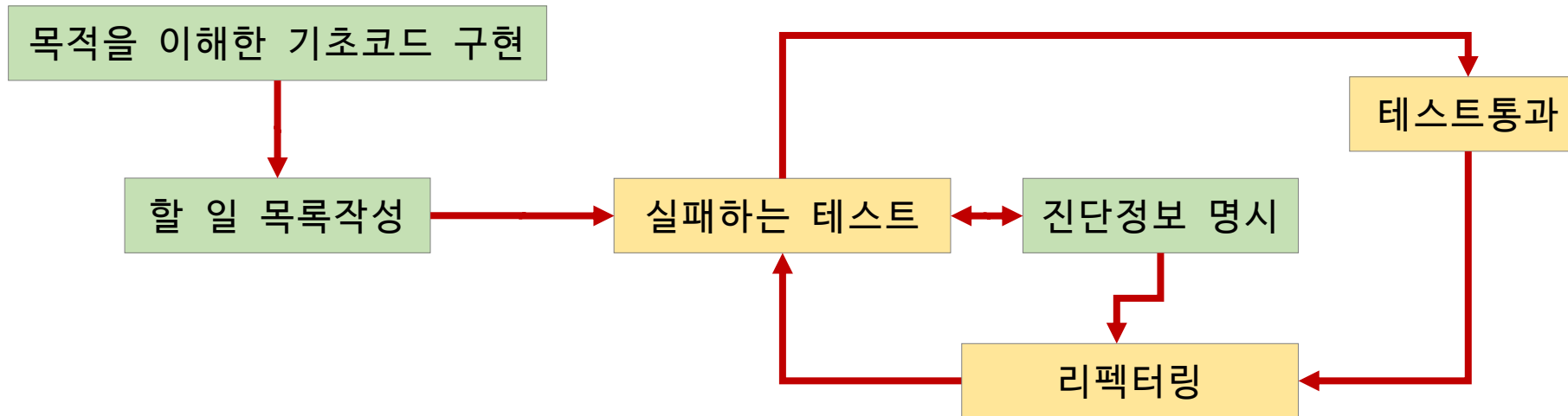
Inverse - 역관계확인

Crosscheck - 교차확인

Error condition -에러조건 강제 생성

Performance -성능허용범위

행위를 테스트
책임과 역할을 드러낼 수 있는 테스트



객체 지향 스타일

규모 큰 코드의 유지보수



관심사의 분리

SOLID
역할모델

더 높은 수준의 추상화

카테고리화
계층화
모델링

객체작성의 원칙

객체망으로 문제해결

단일책임원칙

관계

의존성

Multiplicity, Aggregation, Composition, Dependency

알림

Observable, publisher

조정

delegate, stratege, sub component

전체는 부분의 합보다 단순

특정 역할이 충분히 은닉성있는가

컨텍스트 독립성

잘못된(역할외적인) 요소를 내포하고 있는가

객체 지향 설계의 달성

테스트와 객체설계 및 발견

메인(호스트)코드부터 작성 추상화수준 및 은닉성에 기여(입력에서 출력으로:객체)

테스트와 객체설계 및 발견

메인(호스트)코드부터 작성 추상화수준 및 은닉성에 기여(입력에서 출력으로:객체)

작은 규모의 테스트유지 역할인식 및 분리촉진 - 단일책임원칙실현

테스트와 객체설계 및 발견

메인(호스트)코드부터 작성 추상화수준 및 은닉성에 기여(입력에서 출력으로:객체)

작은 규모의 테스트유지 역할인식 및 분리촉진 - 단일책임원칙실현

의존성주입 콘텍스트 독립성 촉진 및 의존성 정리

테스트와 객체설계 및 발견

메인(호스트)코드부터 작성	추상화수준 및 은닉성에 기여(입력에서 출력으로:객체)
----------------	-------------------------------

작은 규모의 테스트유지	역할인식 및 분리촉진 - 단일책임원칙실현
--------------	------------------------

의존성주입	컨텍스트 독립성 촉진 및 의존성 정리
-------	----------------------

목객체 테스트	필요서비스 파악, 객체 간 관계인식 및 객체의 은닉성 촉진 (입력에서 출력으로:객체망) (컨텍스트 독립성유도)
---------	--

테스트와 객체설계 및 발견

메인(호스트)코드부터 작성 추상화수준 및 은닉성에 기여(입력에서 출력으로:객체)

작은 규모의 테스트유지 역할인식 및 분리촉진 - 단일책임원칙실현

분해 구현편의를 위해 코드상 결합되어있던 책임들

의존성주입 콘텍스트 독립성 촉진 및 의존성 정리

목객체 테스트 필요서비스 파악, 객체 간 관계인식 및 객체의 은닉성 촉진
(입력에서 출력으로:객체망) (콘텍스트 독립성유도)

테스트와 객체설계 및 발견

메인(호스트)코드부터 작성 추상화수준 및 은닉성에 기여(입력에서 출력으로:객체)

작은 규모의 테스트유지 역할인식 및 분리촉진 - 단일책임원칙실현

분해 구현편의를 위해 코드상 결합되어있던 책임들

포장 새로운 추상층이 기존의 책임 여러개를 묶어 의존성 책임을 새로 만들어냄

의존성주입 콘텍스트 독립성 촉진 및 의존성 정리

목객체 테스트 필요서비스 파악, 객체 간 관계인식 및 객체의 은닉성 촉진
(입력에서 출력으로:객체망) (콘텍스트 독립성유도)

테스트와 객체설계 및 발견

메인(호스트)코드부터 작성 추상화수준 및 은닉성에 기여(입력에서 출력으로:객체)

작은 규모의 테스트유지 역할인식 및 분리촉진 - 단일책임원칙실현

분해 구현편의를 위해 코드상 결합되어있던 책임들

포장 새로운 추상층이 기존의 책임 여러개를 묶어 의존성 책임을 새로 만들어냄

의존성주입 콘텍스트 독립성 촉진 및 의존성 정리

목객체 테스트 필요서비스 파악, 객체 간 관계인식 및 객체의 은닉성 촉진
(입력에서 출력으로:객체망) (콘텍스트 독립성유도)

파생 도메인에 이해가 깊어지고 오해한 책임, 새롭게 인지한 책임

값타입과 설계

- 구체적인 형
- 도메인의 개념에 보다 근접하게 표현
 - 이후 행위(메소드)가 추가될 후보가 됨

값타입과 설계

구체적인 형 • 도메인의 개념에 보다 근접하게 표현
 • 이후 행위(메소드)가 추가될 후보가 됨

분해

복잡한 역할을 분해함 - 단일책임원칙

값타입과 설계

구체적인 형 • 도메인의 개념에 보다 근접하게 표현
 • 이후 행위(메소드)가 추가될 후보가 됨

분해	복잡한 역할을 분해함 - 단일책임원칙
----	----------------------

파생	마커타입으로부터 실 도메인역할 수행
----	---------------------

값타입과 설계

구체적인 형 • 도메인의 개념에 보다 근접하게 표현
 • 이후 행위(메소드)가 추가될 후보가 됨

분해	복잡한 역할을 분해함 - 단일책임원칙
----	----------------------

파생	마커타입으로부터 실 도메인역할 수행
----	---------------------

포장	트랜잭션 - 전체는 부분의 합보다 단순
----	-----------------------

값타입과 설계

구체적인 형 • 도메인의 개념에 보다 근접하게 표현
 • 이후 행위(메소드)가 추가될 후보가 됨

분해	복잡한 역할을 분해함 - 단일책임원칙
----	----------------------

파생	마커타입으로부터 실 도메인역할 수행
----	---------------------

포장	트랜잭션 - 전체는 부분의 합보다 단순
----	-----------------------

인터페이스 설계

값타입과 설계

구체적인 형 • 도메인의 개념에 보다 근접하게 표현
 • 이후 행위(메소드)가 추가될 후보가 됨

분해	복잡한 역할을 분해함 - 단일책임원칙
----	----------------------

파생	마커타입으로부터 실 도메인역할 수행
----	---------------------

포장	트랜잭션 - 전체는 부분의 합보다 단순
----	-----------------------

인터페이스 설계

책임분할	구상클래스 제작곤란, 단일책임원칙위반, 과도한 의존성
------	-------------------------------

값타입과 설계

구체적인 형 • 도메인의 개념에 보다 근접하게 표현
 • 이후 행위(메소드)가 추가될 후보가 됨

분해	복잡한 역할을 분해함 - 단일책임원칙
파생	마커타입으로부터 실 도메인역할 수행
포장	트랜잭션 - 전체는 부분의 합보다 단순

인터페이스 설계

책임분할	구상클래스 제작곤란, 단일책임원칙위반, 과도한 의존성
병합	<ul style="list-style-type: none">• 단일 개념에 대한 여러 인터페이스를 통합• 프로토콜의 재설계

객체망 설계

목객체 테스트 • 각 의존성 파악 및 컨텍스트독립성 확립
 • 객체의 재활용 촉진 및 다양한 시나리오 검증

객체망 설계

목객체 테스트

- 각 의존성 파악 및 컨텍스트독립성 확립
- 객체의 재활용 촉진 및 다양한 시나리오 검증

구현계층

클래스측, 구상코드, 서비스

클래스의 복잡한 관계(객체 그래프)
어떻게 작동하는가

객체망 설계

목객체 테스트

- 각 의존성 파악 및 콘텍스트독립성 확립
- 객체의 재활용 촉진 및 다양한 시나리오 검증

구현계층

클래스측, 구상코드, 서비스

클래스의 복잡한 관계(객체 그래프)
어떻게 작동하는가



객체를 직접 구성하며 중복제거

객체망 설계

목객체 테스트

- 각 의존성 파악 및 콘텍스트독립성 확립
- 객체의 재활용 촉진 및 다양한 시나리오 검증

구현계층

클래스측, 구상코드, 서비스

클래스의 복잡한 관계(객체 그래프)
어떻게 작동하는가

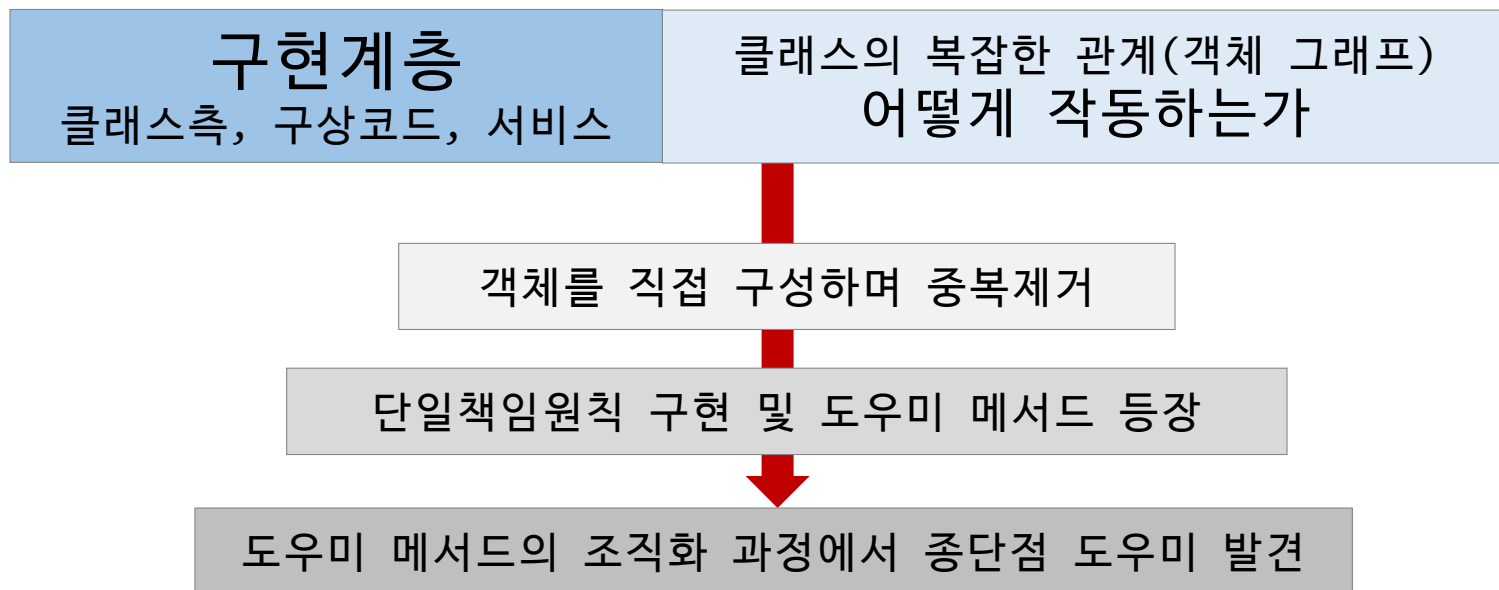
객체를 직접 구성하며 중복제거

단일책임원칙 구현 및 도우미 메서드 등장

객체망 설계

목객체 테스트

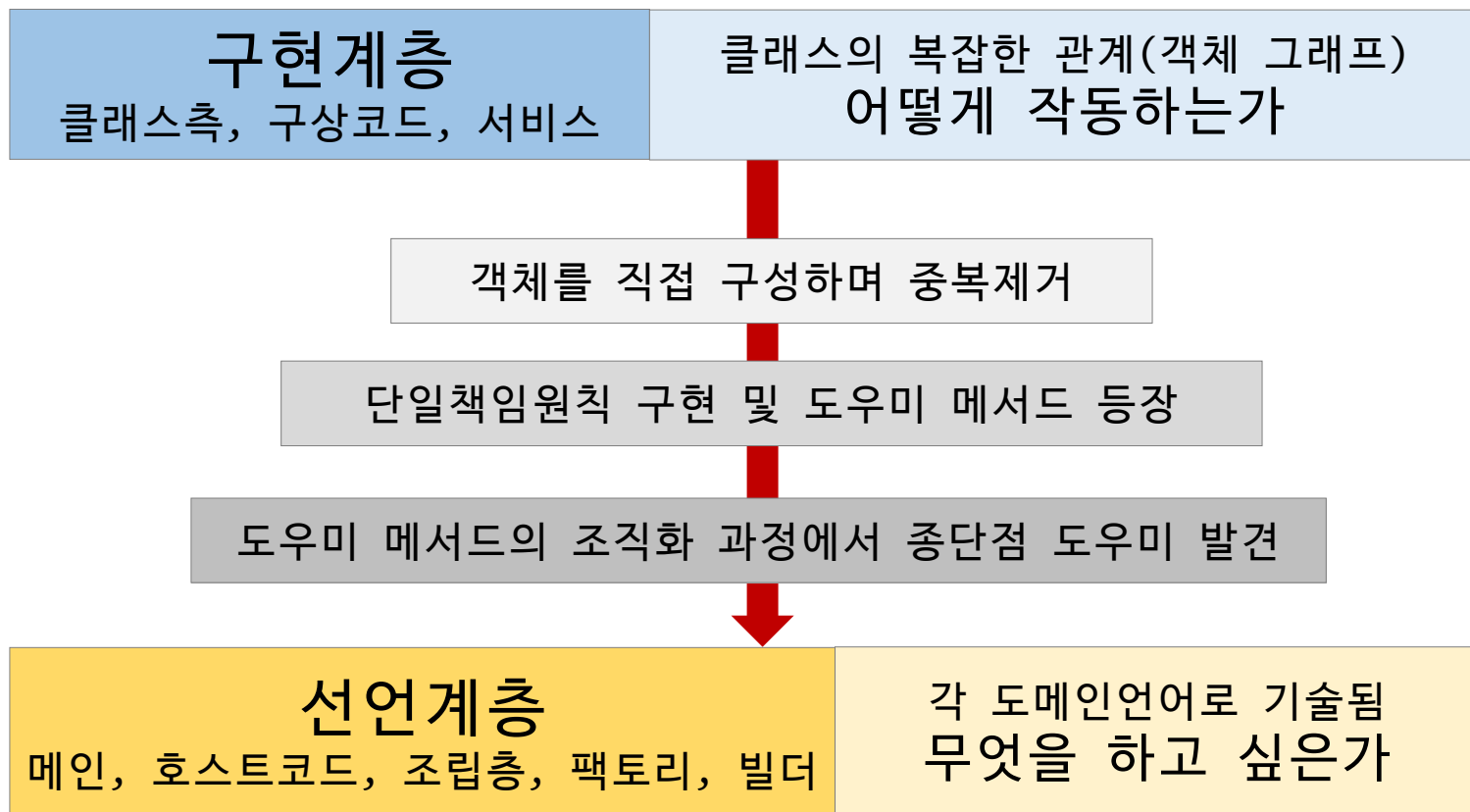
- 각 의존성 파악 및 콘텍스트독립성 확립
- 객체의 재활용 촉진 및 다양한 시나리오 검증



객체망 설계

목객체 테스트

- 각 의존성 파악 및 컨텍스트독립성 확립
- 객체의 재활용 촉진 및 다양한 시나리오 검증



서드 파티 코드를 기반으로 한 개발

어댑터 계층

서드파티코드 수정불가, 에러내포, 자체 버전업, API변경가능성

어댑터 계층

서드파티코드 수정불가, 에러내포, 자체 버전업, API변경가능성



어댑터 - 직접 쓰지 말고 감싸서 사용할 것

어댑터 계층

서드파티코드 수정불가, 에러내포, 자체 버전업, API변경가능성



어댑터 - 직접 쓰지 말고 감싸서 사용할 것

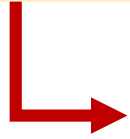
단순한 어댑터

- 서드파티기능과 1:1매칭
- 구현이 쉽고 테스트가 직관적
- 서드파티변경시 추약

어댑터 계층

서드파티코드

수정불가, 에러내포, 자체 버전업, API변경가능성



어댑터 - 직접 쓰지 말고 감싸서 사용할 것

단순한 어댑터

- 서드파티기능과 1:1매칭
- 구현이 쉽고 테스트가 직관적
- 서드파티변경시 취약

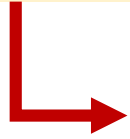
어댑터 레이어

- 애플리케이션 객체와 통신할 책임으로서 기능객체(브릿지)
- 다수의 서드파티API를 사용할 수 있고 내부적인 로직으로 따로 보유함
- 보다 서드파티변경에 안전하며 다른 서드파티로 변경 용이
- 책임역할모델의 완전한 구현과 어댑터에 대한 복잡한 테스트 필요

어댑터 계층

서드파티코드

수정불가, 에러내포, 자체 버전업, API변경가능성



어댑터 - 직접 쓰지 말고 감싸서 사용할 것

단순한 어댑터

- 서드파티기능과 1:1매칭
- 구현이 쉽고 테스트가 직관적
- 서드파티변경시 취약

델리게이터 어댑터

- 단순 어댑터에 이벤트 위임 기능을 추가함
- 이벤트 기반의 서드파티에서 직접 애플리케이션의 바인딩을 보호

어댑터 레이어

- 애플리케이션 객체와 통신할 책임으로서 기능객체(브릿지)
- 다수의 서드파티API를 사용할 수 있고 내부적인 로직으로 따로 보유함
- 보다 서드파티변경에 안전하며 다른 서드파티로 변경 용이
- 책임역할모델의 완전한 구현과 어댑터에 대한 복잡한 테스트 필요