

1

## 단위 테스트의 정의와 필요성

# 01 단위 테스트의 정의와 필요성

## 1 구현의 연장



단위테스팅은 아직 공식적인 테스트 단계로  
진입하지 않은 구현단계의 업무임

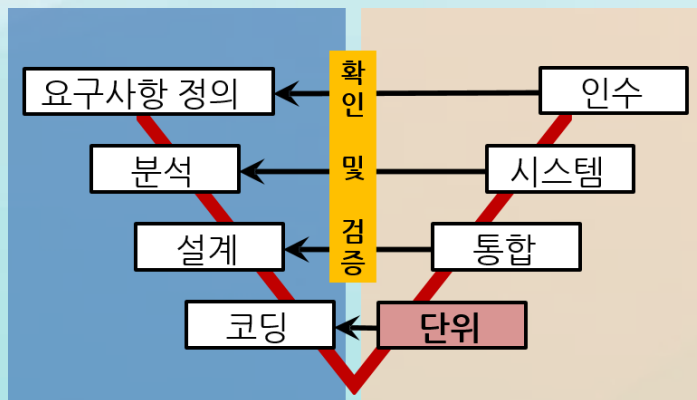
- 단위테스팅이 완료되지 않은  
프로그램은 구현이 완료된 것으로 보기 어려움
- 개발자 스스로 소스코드가 통합되어도  
문제가 없다는 확신이 필요

# 01 단위 테스트의 정의와 필요성

## 2 V모델

테스팅을 강조하기 위한 개념적 SW개발모델

독일의 공식적인 프로젝트  
관리방법 모델로 알려져 있음





※ 출처 : 실무에 바로 활용하는 소프트웨어공학, 김희영저, 21세기사

# 01 단위 테스트의 정의와 필요성

## 2 V모델

### V모델의 특징

-  폭포수 모델에서 시스템 검증/테스트를 강조한 모델으로서, 우측 오류시 좌측으로 돌아감
-  테스트는 한번에 이루어 지는 것이 아니라 각각의 개발 단계에 대응하는 테스트 레벨이 별도로 존재한다는 폭포수 모델에 기초를 두고 있음

# 01 단위 테스트의 정의와 필요성

## 2 V모델

### V모델의 장/단점 및 활용



#### 장점

- 모든 단계의 검증/확인(오류 감소/신뢰성 향상)
- 프로젝트 시작부터 테스트 가능



#### 단점

- 생명주기의 반복을 허용하지 않음으로서 변경을 다루기가 쉽지 않음






#### 활용

- 높은 신뢰성이 필요한 의료 및 공장 제어 분야




# 01 단위 테스트의 정의와 필요성

## 3 단위테스팅의 의미

-  프로그램의 기본단위인 모듈에 대한 테스트를 의미
-  모듈에 대한 설계를 기반으로  
모듈의 중요한 경로를 확인
-  구현단계의 마지막 활동

# 01 단위 테스트의 정의와 필요성







## 4 단위테스팅의 점검사항

-  인터페이스의 점검
-  처리데이터의 점검
-  오류의 점검

# 01 단위 테스트의 정의와 필요성

## 4 단위테스팅의 점검사항

### 인터페이스의 점검

-  입력변수의 개수 점검
-  입력변수의 타입을 확인
-  변수의 단위와 호출모듈 및 호출되는 모듈에서의 변수와 일치하는지 확인
-  입력변수의 순서 확인
-  모듈간 전역변수의 사용에 대한 확인
-  데이터베이스에 대한 사용방법이 정확한지 확인



# 01 단위 테스트의 정의와 필요성

## 4 단위테스팅의 점검사항

### 처리데이터 점검

- 🔍 부정확하거나 일치하지 않은 타입여부 점검
- 🔍 변수의 초기화에 대한 디폴트 값의 확인
- 🔍 변수이름의 정확성 확인
- 🔍 수식의 계산결과 언더플로우(Underflow) 혹은 오버플로우(Overflow) 발생여부 확인

# 01 단위 테스트의 정의와 필요성

## 4 단위테스팅의 점검사항

### 오류처리 점검

- 🔍 오류 메시지에 대한 이해가능성 점검
- 🔍 오류 조건이 정상처리 과정에 반영되는지 확인
- 🔍 예외처리가 정확한지 확인
- 🔍 오류에 대한 서술이 오류 위치를 찾는데 필요한 충분한 정보를 제공하는지 확인

## 2 단위 테스트의 절차







## 02 단위 테스트의 절차

### 1 단위 테스트의 절차



## 02 단위 테스트의 절차

### 2 단위 테스트의 전략

-  목표를 설정
-  테스트 범위를 추정
-  일정과 인원을 파악
-  수행환경 조성
-  수행결과를 정리
-  완료기준에 따른 완료여부 확인

## 02 단위 테스트의 절차

### 3 단위 테스트의 계획

- 🔍 수행범위와 담당자 지정
- 🔍 일정수립 및 보고체계를 가동, 테스트 결과를 위한 보고양식 정의
- 🔍 이슈사항 해결계획 수립
- 🔍 단위테스트 계획서 작성 및 승인

## 02 단위 테스트의 절차

### 3 단위 테스트의 계획



오리엔테이션

- 계획에 대한 정보 공유
- 테스트 기법에 대한 교육
- 이슈발생시 처리 가이드



테스팅 환경 점검

## 02 단위 테스트의 절차

### 3 단위 테스트의 계획



단위테스트 계획서 포함 내용

- 단위 테스트의 목적을 기술하고 단위 테스트 계획, 실행 및 평가 단계에 따른 절차와 담당자, 산출물을 기술
- 단위 테스트 실행 전에 선행될 기준과 단위 테스트가 완료되는 기준 및 단위 테스트 중단/재개 기준을 정의
- 단위 테스트케이스를 도출하기 위한 방안을 수립하고 점검항목을 기술



## 02 단위 테스트의 절차

### 3 단위 테스트의 계획



단위테스트 계획서 포함 내용

- 단위 테스트 수행을 위한 교육 일정을 수립
- 단위 테스트 범위 및 범위 밖의 기능을 선정
- 테스트 환경 및 테스트 조직과 역할을 정의하고 단위 테스트 수행 주요 활동에 대한 상세일정을 수립
- 단위 테스트 수행 시 결함, 진척관리 계획을 수립하고 발생하는 산출물을 정의

## 02 단위 테스트의 절차

### 3 단위 테스트의 계획



단위테스트 계획서 포함 내용

- 단위 테스트 케이스를 작성(계속)

- 단위 테스트 대상에 대한 테스트케이스를 작성하며, 이때 케이스 내용, 사전조건, 테스트 데이터 예상결과를 작성
- 테스트케이스는 값에 따른 분기와 업무 수행 경로로 고려하여 작성

## 02 단위 테스트의 절차

### 3 단위 테스트의 계획



단위테스트 계획서 포함 내용

- 단위 테스트 케이스 작성(계속)

- 단위 테스트 케이스를 실행하기 위해 테스트 데이터를 준비
- 만약 테스트 데이터를 특정한 제품 환경에서 가져온다면 동일한 환경에서 테스트를 수행하기 위해 필요한 모든 데이터 다운로드 할 수 있는 스크립트를 작성

## 02 단위 테스트의 절차

### 4 단위 테스트의 실행

- 🔍 단위테스팅을 위한 테스트용 데이터가 제공
  - 공통의 데이터를 개별 개발자가 생성하여 활용하는 것은 비효율적임
- 🔍 수행결과를 점검
- 🔍 테스트 기법에 대한 가이드를 수시로 제공
- 🔍 보고체계에 의해 결과를 취합 정리
- 🔍 결과를 계량적으로 분석하고 평가하기 위한 작업 완료

## 02 단위 테스트의 절차

### 5 단위 테스트의 평가

- 🔍 측정결과를 계량적으로 분석
- 🔍 계획과 실행 차이를 정리하여 원인 파악
- 🔍 테스트 커버리지 평가
  - 테스트 수행범위를 어느 정도의 수준에서 달성하였는지 평가하는 작업
  - 커버리지를 높이게 되면 그만큼 테스트 업무가 증가
- 🔍 요구사항 대비 결과 평가
- 🔍 미조치 결함 정리

### 3 단위 테스트를 위한 지원



## 03 단위 테스트를 위한 지원

### 1 단위 테스트를 위한 지원

- 단위테스팅 체크리스트
- 테스트용 데이터 제공
- 드라이버(Driver)와 스텝(Stub)
- 테스트 케이스(Test case)

## 03 단위 테스트를 위한 지원

### 2 단위테스팅 체크리스트

-  공통적인 점검사항을  
체크리스트로 만들어서 개발자에게 제공
-  과거 유사한 프로젝트에서 정리



## 03 단위 테스트를 위한 지원

### 2 단위테스팅 체크리스트

[단위테스팅 체크리스트 예]

번호	컴파일러 언어
1	검색어가 공백, 부분입력 데이터일 경우 검색기능을 확인하는가?
2	최대값이나 최소값에 대한 연산처리를 확인하는가?
3	데이터베이스의 기본적인 CRUD를 처리하는가?
4	데이터베이스의 기본키와 외래키에 의한 제약조건이 작용하는가?

## 03 단위 테스트를 위한 지원

### 3 테스트용 데이터 제공



누가 데이터를 제공할 것인가?

- 프로젝트에 투입된  
DA(Data Architect)가 제공하는 경우
- 프로젝트와 관련된  
DBA(DataBase Administrator)가 제공하는 경우
- 테스트 관리자가 제공하는 경우

## 03 단위 테스트를 위한 지원

### 3 테스트용 데이터 제공



#### 생성방법

- 자동생성프로그램(Generation program)을 통해 생성
- 고객으로부터 데이터 샘플을 수령하여 스كر램블 (개인신상정보 등을 변형)하여 생성
- 수작업으로 데이터입력부터 테스트하여 DB에 생성되도록 하는 방법

## 03 단위 테스트를 위한 지원

### 4 드라이버와 스텝

- 🔍 임시로 개발되어 테스트에 활용할 프로그램
  - 드라이버
    - : 테스트할 모듈을 호출하는  
상위 모듈을 임시로 개발한 것
  - 스텝
    - : 테스트할 모듈이 호출할  
하위 모듈을 임시로 개발한 것

## 03 단위 테스트를 위한 지원

### 4 드라이버와 스텝

임시로 개발되어 테스트에 활용할 프로그램

- 드라이버  
: 테스트할 모듈을 호출하는  
상위 모듈을 임시로 개발한 것
- 스텝  
: 테스트할 모듈이 호출할  
하위 모듈을 임시로 개발한 것

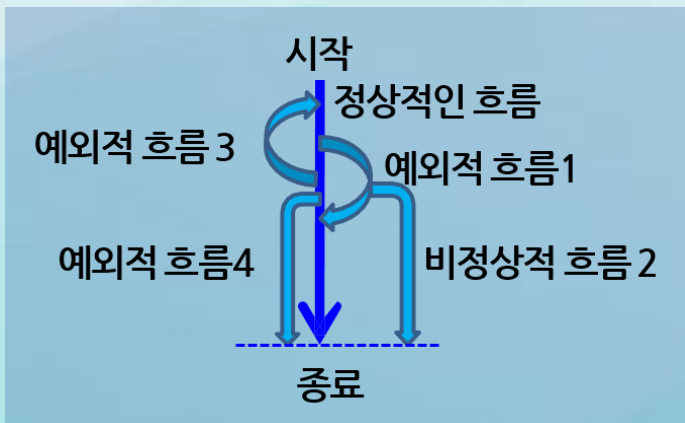


※ 출처 : 실무에 바로 활용하는 소프트웨어공학, 김희영저, 21세기사

## 03 단위 테스트를 위한 지원

### 5 테스트 케이스


- 설계서를 기반으로 테스트 케이스를 만들 수 있음
- 유스케이스를 기반으로도 테스트 케이스 생성이 가능



※ 출처 : 실무에 바로 활용하는 소프트웨어공학, 김희영저, 21세기사

## 03 단위 테스트를 위한 지원

### 5 테스트 케이스

 생성된 테스트 케이스는 예상결과와 대비하여 pass 혹은 fail을 판단할 수 있음

번호	케이스	조건1	조건2	예상결과
1	케이스 1	Valid	Valid	결과 1
2	케이스 2	Invalid	Valid	결과 2
3	케이스 3	Valid	Invalid	결과 3
4	케이스 4	Invalid	Invalid	결과 4