



애플리케이션 테스트 수행 part 1



명세/구조 기반 테스트 케이스 작성

학습내용

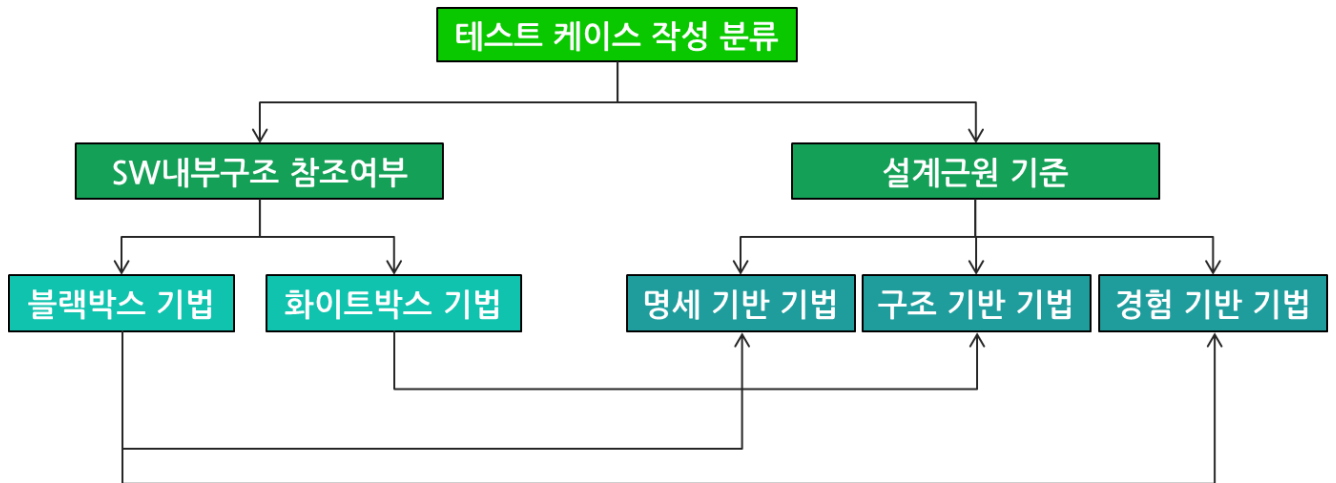
- 테스트 케이스 작성법

학습목표

- ❖애플리케이션 테스트 케이스 작성법의 종류를 설명할 수 있다.
- ❖명세 기반 테스트 케이스 작성법과 구조 기반 테스트 케이스 작성법을 활용할 수 있다.

테스트 케이스 작성법

1 테스트 케이스 작성법 분류



테스트 케이스 작성법

2 화이트박스 및 블랙박스 테스트 기법

화이트박스 테스트 케이스 작성법

- 프로그램의 **내부 구조를 알고** 테스트 케이스를 작성하는 기법

블랙박스 테스트 케이스 작성법

- 프로그램의 **내부 구조를 모르고** 오직 외부의 기능 동작 명세만 가지고 테스트 케이스를 작성하는 기법

테스트 케이스 작성법

3 명세 기반 테스트 케이스 작성법

1 명세 기반 테스트 케이스 작성법이란?

명세 기반
테스트
케이스
작성법

- 프로그램 요구 분석 명세서 혹은 설계사양서 등의 명세를 기반으로 테스트 케이스를 작성
- 기능 중심 테스트 케이스 작성법

테스트 케이스 작성법

③ 명세 기반 테스트 케이스 작성법

② 명세 기반 테스트 케이스 작성법 종류

종류	설명
동등 분할 기법	<ul style="list-style-type: none"> 입력값 및 출력값 영역을 상호 독립적 집합으로 나누어 시험사례 유형들을 분할하여 테스트 예) 입력데이터 x값이 0 ~ 100 사이여야 한다면 : 테스트케이스를 ($x < 0$), ($x = 50$), ($x > 100$)으로 분할하여 적용
경계값 분석 기법	<ul style="list-style-type: none"> 경계 부분에 해당하는 입력값에서 결함이 발생될 확률이 높기 때문에 경계값을 포함하여 테스트 예) x값이 0 ~ 100 사이여야 한다면 : 테스트케이스를 ($x=0$), ($x=100$), ($x=-0.01$), ($x=100.1$)로 정의
결정 테이블 기법	<ul style="list-style-type: none"> 주요한 의사결정 요소들을 표로 만들고, 요소들 간의 결합에 의한 테스트 케이스 설계 각 의사결정 요소들의 조합을 통해 다양한 형태의 테스트 시나리오를 도출
상태전이 기법	<ul style="list-style-type: none"> 상태 전이 다이어그램을 통해 이벤트, 액션, 활동, 상태 변화로 발생하는 관계, 동작을 파악하여 검증 임베디드 소프트웨어 테스트 시 많이 적용됨
유스케이스 기법	<ul style="list-style-type: none"> 유스케이스를 통해 도출되는 비즈니스 시나리오 (기본 흐름, 대체 흐름)를 기반으로 테스트를 명세화 컴포넌트/단위 레벨 유스케이스 테스트 시스템 레벨 유스케이스 테스트

테스트 케이스 작성법

③ 명세 기반 테스트 케이스 작성법

② 명세 기반 테스트 케이스 작성법 종류

종류	설명
페어와이즈 기법	<ul style="list-style-type: none"> • ‘관찰 결과 대부분의 결함이 2개의 상호작용에 기인한다’에 착안 • 2개의 요소의 모든 조합을 다룸 • 페어와이즈 조합의 의미는 테스트를 하는데 필요한 각 값들이 다른 파라미터의 값과 최소한 한 번씩은 조합을 이룸
직교 배열 기법	<ul style="list-style-type: none"> • ‘각 행과 열이 페어와이즈 하다는 것으로 어느 행의 조합도 다른 행의 조합과 서로 다르고, 어느 열의 조합도 다른 열의 조합과 서로 다르다’에 착안 • 직교 배열의 원리를 테스트 기법에 적용하여 조합의 수를 줄여 테스트케이스 수를 줄이는 테스트 기법

테스트 케이스 작성법

3 명세 기반 테스트 케이스 작성법

3 동등 분할 기법

로밍 부가서비스 중복 등록 기능 개발	
현황	<ul style="list-style-type: none"> 데이터 정액권 1,3,5만원의 경우 유효기간(14일) 동안 중복 등록이 불가 고객이 각 서비스의 데이터량 소진에 따라 추가 등록 요청하여도 현재는 서비스 제공 불가함
개선요청	<ul style="list-style-type: none"> 유효기간(14일) 이내에도 중복 등록이 가능하도록 수정

동등 분할 기법
적용

테스트 케이스(14일 이후)	예상 결과
<ul style="list-style-type: none"> 유효기간(14일) 이후 데이터 정액권 중복 등록 후 서비스 제공 가능 여부 확인 	서비스 제공 가능

테스트 케이스(14일)	예상 결과
<ul style="list-style-type: none"> 유효기간(14일)에 데이터 정액권 중복 등록 후 서비스 제공 가능 여부 확인 	서비스 제공 가능

테스트 케이스(14일 이내)	예상 결과
<ul style="list-style-type: none"> 유효기간(14일) 이내 데이터 정액권 중복 등록 후 서비스 제공 가능 여부 확인 	서비스 제공 가능

번호이동 인증 오류/한도 횟수 체크	
요구사항	<ul style="list-style-type: none"> 번호이동 인증 오류 1일 5회 초과이면 당일 내 번호이동 처리가 불가하여 익일 처리를 해야 함

동등 분할 기법
적용

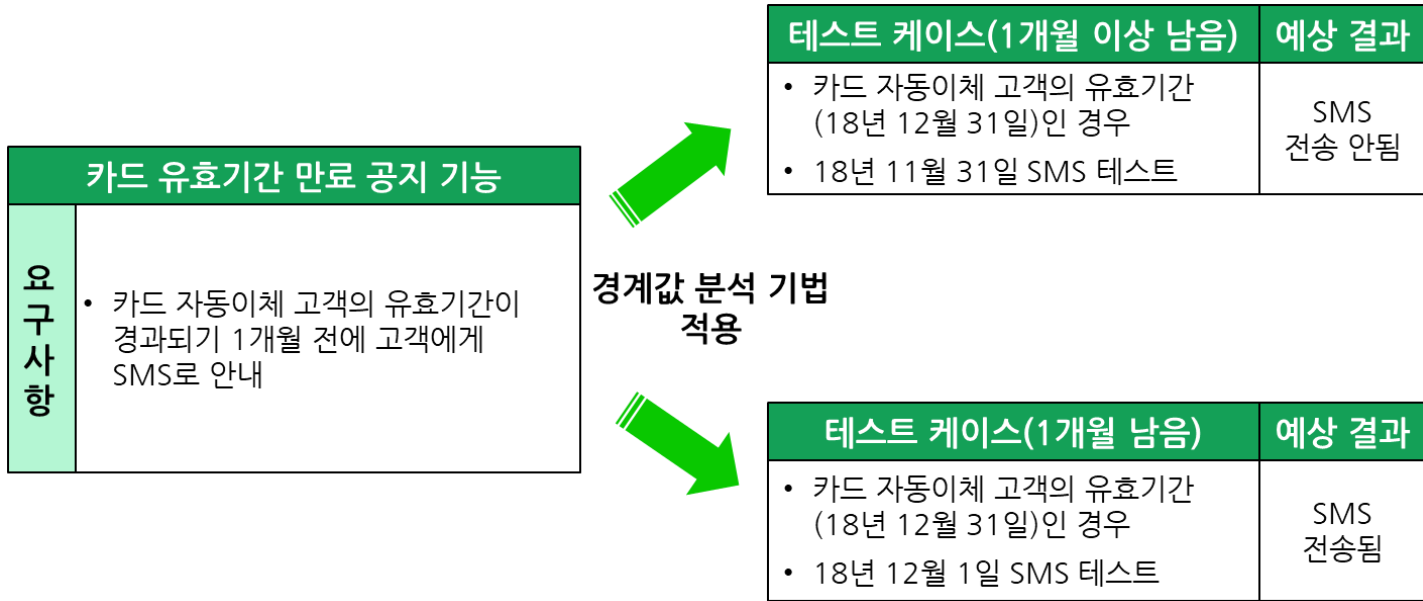
테스트 케이스(1일 이내)	예상 결과
<ul style="list-style-type: none"> 번호이동 인증 오류 4회 발생 후 번호이동 처리 시도 	번호이동 가능

테스트 케이스(1일 이내)	예상 결과
<ul style="list-style-type: none"> 번호이동 인증 오류 5회 발생 후 번호이동 처리 시도 	번호이동 불가

테스트 케이스 작성법

③ 명세 기반 테스트 케이스 작성법

④ 경계값 분석 기법



테스트 케이스 작성법

③ 명세 기반 테스트 케이스 작성법

⑤ 유스케이스 기반 기법

1

물품담당자는 부품청구내역을 조회하기 위해 다음 조건을 입력하고 조회 버튼 클릭

- 사업소, 청구번호, 청구일자, 수불번호, 수불일자, 청구부서, 처리상태

2

시스템은 조회 조건에 해당되는 리스트를 화면에 제시

- 청구번호, 청구일자, 수불번호, 수불일자, 청구(부서, 사번, 이름), 승인(부서, 사번, 이름), 처리상태, 청구사유

3

물품담당자는 단계 4a~4c 어떤 순서로든 처리하고자 하는 작업 수행

테스트 케이스 작성법

3 명세 기반 테스트 케이스 작성법

5 유스케이스 기반 기법

4a. 등록

4b. 수정

- 4a1. 물품담당자는 신규 입력을 하기 위하여 청구등록 버튼을 클릭
- 4a2. 시스템은 신규 입력 가능하도록 화면 처리를 수행하고 다음과 같은 기본정보를 화면에 제시
 - 사업소, 청구일자, 처리상태, 청구부서, 청구담당
- 4a3. 물품담당자는 다음과 같은 정보를 입력 또는 선택하고 저장 버튼을 클릭
 - 청구일자, 사용편성, 청구사유 분류코드, 부품코드, 청구수량
- 4a4. 시스템은 청구번호를 생성하고 입력한 데이터를 등록
- 4a5. 시스템은 메시지(C0001:등록이 완료되었습니다.)를 처리

4a. 등록

4b. 수정

- 4b1. 물품담당자는 수정하고자 하는 행을 선택하고 청구상세 버튼을 클릭
- 4b2. 시스템은 선택된 데이터의 수정이 가능하도록 화면 처리를 하고 상세 내역을 화면에 제시
 - 사업소, 청구번호, 청구일자, 사용편성, 처리상태, 청구부서, 청구담당, 청구사유, 분류코드, 부품코드, 품명, 규격, 단위, 참고재고, 청구수량, 단가, 금액
- 4b3. 물품담당자는 항목을 변경하고 저장버튼을 클릭

테스트 케이스 작성법

③ 명세 기반 테스트 케이스 작성법

⑤ 유스케이스 기반 기법

테스트 케이스	예상 결과
1. 사업소, 청구번호, 청구일자, 수불번호, 수불일자, 청구부서, 처리상태를 입력하고 조회 버튼을 클릭	2. 리스트 Display
3-1. 물품담당자는 신규 입력을 위하여 청구 등록 버튼을 클릭	3-1. 신규 입력 가능하도록 화면 처리
3-2. 물품담당자는 다음과 같은 정보를 입력하고 저장 버튼을 클릭 청구일자, 사용편성, 청구사유 분류코드, 부품코드, 청구수량	3-2. 등록완료 메시지 출력
4-1. 물품담당자는 수정하고자 하는 행을 선택하고 청구 상세 버튼을 클릭	4-1. 상세 내역 Display
4-2. 항목을 변경하고 저장 버튼을 클릭	4-2. 저장완료 메시지 출력

테스트 케이스 작성법

4 구조기반 테스트 케이스 작성법

① 구조기반 테스트 케이스 작성법이란?

구조기반
테스트
케이스
작성법

- 프로그램 구조에 기반하여 테스트 케이스 작성
- 소스코드를 분석하면서 테스트 케이스 작성

테스트 케이스 작성법

4 구조기반 테스트 케이스 작성법

② 구조기반 테스트 케이스 작성법 종류

종류	설명
기본 경로 테스트 기법	<ul style="list-style-type: none"> • 프로그램의 기본 흐름을 테스트 • 조건/분기, 반복이 없다고 가정했을 때에 기본적으로 흘러가는 경로에 대한 테스트
제어 구문 테스트 기법	<ul style="list-style-type: none"> • If-Then-Else 구문에 따라 바뀌는 경로를 테스트 • Switch-Case문에 따라 바뀌는 경로를 테스트
반복 구문 테스트 기법	<ul style="list-style-type: none"> • For-Loop, While-Loop, Repeat 등 반복적인 구문에 대한 테스트 • 반복횟수, 빠져 나오는 조건 테스트

테스트 케이스 작성법

4 구조기반 테스트 케이스 작성법

③ 기본 경로 테스트 기법

기본 경로 테스트 기법

- 분기문(if-then-else), 반복문(for-loop, while-loop, repeat-loop)이 포함되지 않은 구문 테스트
- 가장 기본적인 구문에 대한 테스트

```
name = a.name;
city = a.city;
salary = a.salary;
age = a.age;
```

코드

```
If name == wojoo or city == sungnam
then salary = 200;
else salary = 150;
end-if
if tasks performed == y and age >= 18
then raise 100;
else if female == y
    then raise 50;
    end-if
end-if
for (int i = 0 to 100; i++) {
    salary = salary + i * 100;
}
```


테스트 케이스 작성법

4 구조기반 테스트 케이스 작성법

④ 제어 경로 테스트 기법

제어 경로 테스트 기법

- if-then-else 등의 **제어 구문의 결과**에 따라서 로직이 바뀌는 경로를 테스트
- **코드 커버리지**와 연관성이 높음

코드

```

name = a.name;
city = a.city;
salary = a.salary;
age = a.age;
If name == wojoo or city == sungnam
then salary = 200;
else salary = 150;
end-if
if tasks performed == y and age >= 18
then raise 100;
else if female == y
    then raise 50;
    end-if
end-if
for (int i = 0 to 100; i++) {
    salary = salary + i * 100;
}
  
```

테스트 케이스	예상 결과
name = wojoo인 경우	• Salary = 200
city = sungnam인 경우	• Salary = 200
name <> wojoo and city <> sungnam인 경우	• Salary = 150

테스트 케이스 작성법

4 구조기반 테스트 케이스 작성법

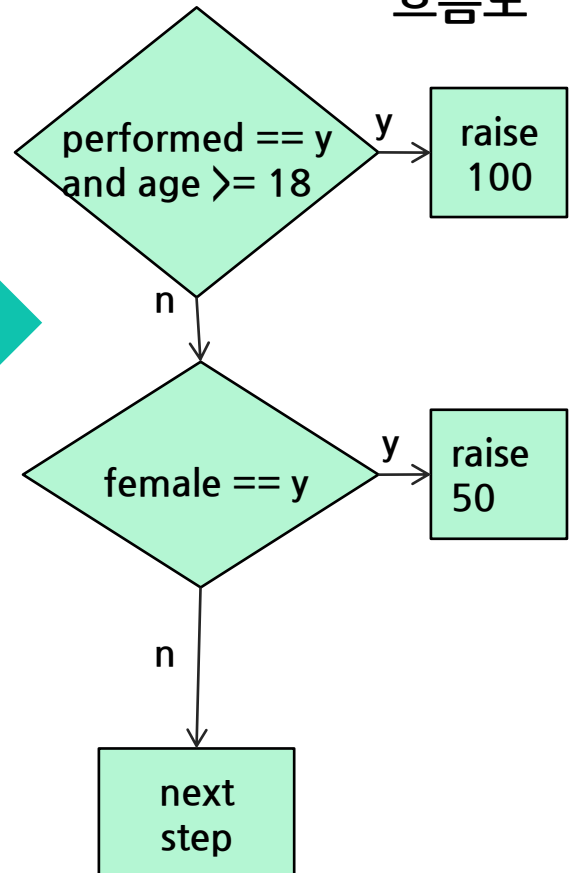
④ 제어 경로 테스트 기법

코드

```

name = a.name;
city = a.city;
salary = a.salary;
age = a.age;
If name == wojoo or city == sungnam
then salary = 200;
else salary = 150;
end-if
if tasks performed == y and age >= 18
then raise 100;
else if female == y
    then raise 50;
    end-if
end-if
for (int i = 0 to 100; i++) {
    salary = salary + i * 100;
}
    
```

흐름도



테스트 케이스 작성법

4 구조기반 테스트 케이스 작성법

④ 제어 경로 테스트 기법

테스트 케이스	예상 결과 (performed == y and age >= 18)
performed = n	• false
performed = y, age = 20	• true

테스트 케이스	예상 결과(female == y)
female = n	• false
female = y	• true

테스트 케이스 작성법

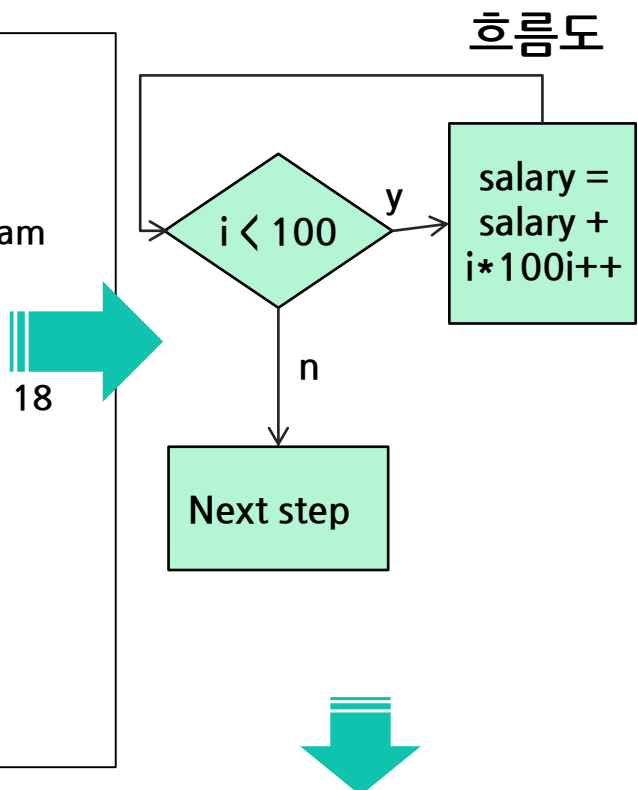
4 구조기반 테스트 케이스 작성법

5 반복 구문 테스트 기법

반복 구문
테스트
기법

for-loop, while-loop, repeat-loop 등
반복적으로 이루어지는 구문에 대한 테스트
케이스 작성

```
name = a.name;
city = a.city;
salary = a.salary;
age = a.age;
If name == wojoo or city == sungnam
then salary = 200;
else salary = 150;
end-if
if tasks performed == y and age >= 18
then raise 100;
else if female == y
    then raise 50;
end-if
end-if
for (int i = 0 to 100; i++) {
    salary = salary + i * 100;
}
```



테스트 케이스	예상 결과(i < 100)
i = 0	• false
i = 1	• false

1. 테스트 케이스 작성법

- 명세 기반 테스트 케이스 작성:
소프트웨어 내부 구조를 모르는 상태에서 외부에 드러난 명세를 기반으로 테스트 케이스를 작성하는 기법
 - 동등 분할 기법: 입력값 및 출력값 영역을 상호 독립적 집합으로 나누어 시험사례 유형들을 분할하여 테스트
 - 경계값 분석 기법: 경계 부분에 해당하는 입력값에서 결함이 발생할 확률이 높기 때문에 경계값을 포함하여 테스트
 - 상태전이 기법: 상태 전이 다이어그램을 통해 이벤트, 액션, 활동, 상태 변화로 발생하는 관계, 동작을 파악하여 검증
 - 유스케이스 기반 기법: 유스케이스를 통해 도출되는 비즈니스 시나리오(기본 흐름, 대체 흐름)를 기반으로 테스트를 명세화
 - 결정 테이블 기법: 주요한 의사결정 요소들을 표로 만들고, 요소들 간의 결합에 의한 테스트 케이스 설계

1. 테스트 케이스 작성법

- 구조 기반 테스트 케이스 작성:
소프트웨어 내부 구조를 파악한 상태에서 내부 구조에 기반한 테스트 케이스를 작성하는 기법
 - 기본 경로 테스트 : 조건/분기, 반복이 없다고 가정했을 때에 기본적으로 흘러가는 경로에 대한 테스트
 - 제어 구문 테스트 : if-then-else 구문과 switch-case문에 따라 바뀌는 경로를 테스트
 - 반복 구문 테스트 기법 : for-loop, while-loop, repeat 등 반복적인 구문에 대해 반복횟수, 빠져 나오는 조건 테스트