Software Modeling & Analysis

OSP stage 2050 & 2060

[Construct & Test]

- 1 to 10 CPT Tool -

Team.#	6
과목명	소프트웨어 모델링 및 분석
담당교수	유준범 교수님
팀원	201211938 황준익
	201310350 손성호
	201414135 이광제
	201212088 이용주

Contents

Activity 2050. Construct

Activity 2051. Implement Class & Methods Definition

- Class Definition

Type	Class
Name	Task
Purpose	전체 Category, Constraints, Property 들을 관리하는 클래스
Overview	N/A
Cross References	Functions : R 1.1, R 1.2, R 2.1, R 2.2, R 2.3, R 3.1, R 3.2, R 3.3, R 4.1, R 4.2, R 4.3, R 5.1, R 5.2, R 5.3, R 6.1, R 6.2.1, R 6.2.2, R 6.3, R 7.1 Use Cases : Load File, Save File, New Category, Remove Category, Modify Category, New Value, Remove Value, Modify Value, New Constraints, Remove Constraints, Modify Constraints, New Property, Remove Property, Modify Property, Calculate, Show Result, Filter Representative Value, Filter Property, Provide Help, Check True/False
Exceptional Courses of Events	N/A

Type	Class
Name	Category
Purpose	각 Category 들의 정보를 저장하는 클래스.
Overview	N/A
Cross References	Functions : R 1.1, R 2.1, R 2.2, R 2.3, R 3.1, R 3.2, R 3.3 Use Cases : Load File, New Category, Remove Category, Modify Category, New Rep. Value, Remove Rep. Value, Modify Rep. Value
Exceptional Courses of Events	N/A

Type	Class
Name	RepValue
Purpose	각 Representative Value 들의 정보를 저장하는 클래스.
Overview	N/A
Cross References	Functions : R 1.1, R 3.1, R 3.2, R 3.3, R 4.1, R 4.2, R 4.3, R 5.2, R 6.1, R 6.2, R 6.2.1, R 7.1 Use Cases : Load File, New Rep. Value, Remove Rep. Value, Modify Rep. Value, New Constraints, Remove Constraints, Modify Constraints, Calculate, Show Result, Filter Representative Value, Check True/False
Exceptional Courses of Events	N/A

Type	Class
Name	Constraints
Purpose	각 Constraints 들의 정보를 저장하는 클래스.
Overview	N/A
Cross References	Functions : R 1.1, R 2.2, R 3.2, R 4.1, R 4.2, R 4.3, R 5.2 Use Cases : Load File, Remove Category, Remove Rep. Value, New Constraints, Remove Constraints, Modify Constraints, Remove Property
Exceptional Courses of Events	N/A

Type	Class
Name	Property
Purpose	각 Property 들의 정보를 저장하는 클래스.
Overview	N/A
Cross References	Functions : R 1.1, R 2.2, R 3.2, R 4.2, R 5.1, R 5.2, R 5.3, R 6.1, R 6.2.2 Use Cases : Load File, Remove Category, ,Remove Rep. Value, Remove Constraints, New Property, Remove Property, Modify Property, Calculate, Filter Property
Exceptional Courses of Events	N/A

Type	Class
Name	Result
Purpose	Test Case 계산 결과를 저장하기 위한 클래스.
Overview	N/A
Cross References	Functions : R 6.1, R 6.2, R 6.2.1, R 6.2.2, R 7.1 Use Cases : Calculate, Show Result, Filter Representative Value, Filter Property, Check True/False
Exceptional Courses of Events	N/A

- Class_GUI Definition

Type	Class_GUI
Name	ComponenetSet
Purpose	Category, Representative Value, Property 의 리스트를 가시적으로 보여주며 sel, del, mdf 버튼을 통해 유저의 행동을 수행해주는 컴포넌트 조합 클래스
Overview	Type 인자를 이용해서 어떤 객체를 대변하는 지 식별한다.
Cross Reference	Functions : R 2.2, R 2.3, R 3.2, R 3.3, R 5.2, R 5.3 Use Cases : Remove Category, Modify Category, Remove Rep. Value, Modify Rep. Value, Remove Property, Modify Property
Exceptional Courses of Events	-

Type	Class_GUI
Name	CstComponenetSet
Purpose	선택된 Rep. Value 의 Constraints 관계에 대한 리스트를 가시적으로 보여주며, combo, del, sel 버튼을 통해 유저의 행동을 수행하는 컴포넌트 조합 클래스
Overview	Oldtype 인자를 통해 어떤 객체를 대변하는지 식별한다.
Cross Reference	Functions : R 4.1, R 4.2, R 4.3 Use Cases : New Constraints, Remove Constraints, Modify Constraints
Exceptional Courses of Events	-

- Method Definition

Type	Method
Name	Load_work
Purpose	User 가 기존에 작업 중이던 파일을 불러오도록 한다.
Cross References	Functions : R 1.1 Use Cases : Load File
Input (Method)	-
Output (Method)	-
Abstract Operation (Method)	1. 사용자가 선택한 경로의 파일을 불러와서 정보를 각 클래스에 저장한다.
Exceptional Courses of Events	N/A

Type	Method
Name	Save_work
Purpose	User 가 프로그램으로 작업 중이던 파일을 저장하도록 한다.
Cross References	Functions : R 1.2 Use Cases : Save File
Input (Method)	-
Output (Method)	-
Abstract Operation (Method)	1. 사용자가 선택한 경로에 현재 작업 중이던 Task 정보를 파일로 저장한다.
Exceptional Courses of Events	N/A

Type	Method
Name	Select_file
Purpose	파일 저장 혹은 불러오기 시, 파일의 저장 경로와 이름을 선택할 수 있도록 한다.
Cross References	Functions : R 1.1, R 1.2 Use Cases : Load File, Save Result
Input (Method)	-
Output (Method)	-
Abstract Operation (Method)	1. 파일 경로를 GUI 를 통해 선택한다.
Exceptional Courses of Events	N/A

Type	Method
Name	New_category
Purpose	User 가 새로운 Category 를 생성할 수 있도록 한다.
Cross References	Functions : R 2.1 Use Cases : New Category
Input (Method)	Name : String
Output (Method)	-
Abstract Operation (Method)	1. name 을 이름으로 갖는 카테고리를 생성한다.
Exceptional Courses of Events	N/A

Type	Method
Name	Remove_category
Purpose	User 가 Category 를 삭제할 수 있도록 한다.
Cross References	Functions : R 2.2 Use Cases : Remove Category
Input (Method)	Cindex : int
Output (Method)	-
Abstract Operation (Method)	1. Cindex에 해당하는 Category 를 삭제한다.
Exceptional Courses of Events	N/A

Type	Method
Name	Modify_category
Purpose	User 가 Category 정보를 수정할 수 있도록 한다.
Cross References	Functions : R 2.3 Use Cases : Modify Category
Input (Method)	Cindex : int, name : String
Output (Method)	-
Abstract Operation (Method)	1. Cindex에 해당하는 카테고리의 이름을 name 으로 수정한다.
Exceptional Courses of Events	N/A

Type	Method
Name	New_rep_value
Purpose	User 가 새로운 Rep. Value 를 생성할 수 있도록 한다.
Cross References	Functions : R 3.1 Use Cases : New Rep. Value
Input (Method)	Cindex : int
Output (Method)	-
Abstract Operation (Method)	1. cindex 에 해당하는 카테고리에 new_rep_value 메소드를 실행한다.
Exceptional Courses of Events	N/A

Type	Method
Name	Remove_rep_value
Purpose	User 가 Rep. Value 를 삭제할 수 있도록 한다.
Cross References	Functions : R 2.2, R 3.2 Use Cases : Remove Category, Remove Rep. Value
Input (Method)	Cindex : int, rindex : int
Output (Method)	-
Abstract Operation (Method)	1. cindex 에 해당하는 카테고리의 remove_rep_value 메소드를 실행한다.
Exceptional Courses of Events	

Type	Method
Name	Modify_rep_value
Purpose	User 가 Rep. Value 의 정보를 수정할 수 있도록 한다.
Cross References	Functions : R 3.3 Use Cases : Modify Rep. Value
Input (Method)	Cindex : int, rindex : int, name : String
Output (Method)	-
Abstract Operation (Method)	1. cindex 에 해당하는 카테고리의 modify_rep_value 메소드를 실행한다.
Exceptional Courses of Events	N/A

Type	Method
Name	New_const
Purpose	User 가 새로운 Constraints 를 생성할 수 있도록 한다.
Cross References	Functions : R 4.1 Use Cases : New Constraints
Input (Method)	Cindex : int, rindex : int
Output (Method)	-
Abstract Operation (Method)	1. cindex, rindex 에 해당하는 Rep. Value 를 찾는다. 2. Constraints 에 add_const 메소드를 실행한다. 3. 해당하는 Rep. Value 의 add_const 메소드를 실행한다.
Exceptional Courses of Events	N/A

Type	Method
Name	Remove_const
Purpose	User 가 Constraints 를 삭제할 수 있도록 한다.
Cross References	Functions : R 4.2 Use Cases : Remove Constraints
Input (Method)	Cindex : int, rindex : int, type : String
Output (Method)	-
Abstract Operation (Method)	1. cindex, rindex 에 해당하는 Rep. Value 를 찾는다. 2. Constraints 에 remove_const 메소드를 실행한다. 3. 해당하는 Rep. Value 의 remove_const 메소드를 실행한다.
Exceptional Courses of Events	E1. Type 이 property / if_property 인 경우 2. Property 에 remove_const 메소드를 실행한다. 3. 해당하는 Rep. Value 의 remove_const 메소드를 실행한다.

Type	Method
Name	Modify_const
Purpose	User 가 Constraints 의 상태를 변경할 수 있도록 한다.
Cross References	Functions : R 4.3 Use Cases : Modify Constraints
Input (Method)	Cindex : int, rindex : int, oldpindex : int, oldtype : String, type : String, prop : Property
Output (Method)	-
Abstract Operation (Method)	1. cindex, rindex 에 해당하는 Rep. Value 를 찾는다. 2. Constraints 에 remove_const 메소드를 실행한다. 3. Rep. Value 의 remove_const 메소드를 실행한다. 4. Constraints 에 add_const 메소드를 실행한다. 5. Rep. Value 의 add_const 메소드를 실행한다. 6. calculate 메소드를 실행한다.
Exceptional Courses of Events	E1. Oldtype 이 property / if_property 인 경우 2. Property 에 remove_const 메소드를 실행한다. 3. Rep. Value 의 remove_const 메소드를 실행한다. E2. Type 이 property / if_property 인 경우 4. Property 에 add_const 메소드를 실행한다. 5. Rep. Value 의 add_const 메소드를 실행한다.

Type	Method
Name	New_property
Purpose	User 가 새로운 Property 를 생성할 수 있도록 한다.
Cross References	Functions : R 5.1 Use Cases : New Property
Input (Method)	Name : String
Output (Method)	-
Abstract Operation (Method)	1. name 을 이름으로 갖는 Property 를 생성한다.
Exceptional Courses of Events	N/A

Type	Method
Name	Remove_property
Purpose	User 가 Property 를 삭제할 수 있도록 한다.
Cross References	Functions : R 2.2, R 5.2 Use Cases : Remove Category, Remove Property
Input (Method)	Pindex : int
Output (Method)	-
Abstract Operation (Method)	1. pindex 에 해당하는 Property 를 찾는다. 2. 해당 Property 를 갖고 있는 Rep. Value 들에 대해서 모두 해당 정보를 삭제한다. 3. calculate 메소드를 실행한다.
Exceptional Courses of Events	N/A

Type	Method
Name	Modify_property
Purpose	User 가 Property 정보를 변경할 수 있도록 한다.
Cross References	Functions : R 5.3 Use Cases : Modify Property
Input (Method)	Pindex : int, name : String
Output (Method)	-
Abstract Operation (Method)	1. pindex 에 해당하는 Property 를 찾는다. 2. 해당 Property 의 이름을 name 으로 변경한다.
Exceptional Courses of Events	N/A

Type	Method
Name	Calculate
Purpose	입력된 정보를 바탕으로 Test Case 개수를 계산한다.
Cross References	Functions : R 2.2, R 3.1, R 3.2, R 4.1, R 4.2, R 4.3, R 5.2, R 6.1, R 6.2 Use Cases : Remove Category, New Rep. Value, Remove Rep. Value, New Constraints, Remove Constraints, Modify Constraints, Remove Property, Calculate, Show Result
Input (Method)	-
Output (Method)	-
Abstract Operation (Method)	1. Task 에 저장된 정보를 바탕으로 Test Case 를 계산한다.
Exceptional Courses of Events	N/A

Type	Method
Name	Combination()
Purpose	입력된 정보를 바탕으로 Test Case 조합 결과를 생성한다.
Cross References	Functions : R 6.1, R 6.2 Use Cases : Calculate, Show Result
Input (Method)	-
Output (Method)	-
Abstract Operation (Method)	1. Task 에 저장된 정보를 바탕으로 Test Case 조합 결과를 생성한다.
Exceptional Courses of Events	N/A

Type	Method
Name	Filtering
Purpose	User 가 체크한 filter 를 바탕으로 Test Case 조합 결과를 보여준다.
Cross References	Functions : R 6.2.1, R 6.2.2 Use Cases : Filter Rep. Value, Filter Property
Input (Method)	-
Output (Method)	-
Abstract Operation (Method)	1. Task 에 저장된 Rep. Value, Property 들의 filter 여부를 확인해서 result List 를 변경한다.
Exceptional Courses of Events	N/A

Type	Method
Name	Check_filter
Purpose	User 가 Window2 에서 필터링할 Rep. Value 혹은 Property 를 체크할 수 있도록 한다.
Cross References	Functions : R 6.2.1, R 6.2.2 Use Cases : Filter Rep. Value, Filter Property
Input (Method)	Type : String, cindex : int, rindex : int, pindex : int
Output (Method)	-
Abstract Operation (Method)	1. cindex, rindex 에 해당하는 Rep. Value 를 찾는다. 2. 해당 Rep. Value 의 mark_filter 메소드를 실행한다. 3. Result 의 make_filter_list 메소드를 실행한다.
Exceptional Courses of Events	E1. Type 이 Property 인 경우 1. pindex 에 해당하는 Property 를 찾는다. 2. 해당 Property 의 mark_filter 메소드를 실행한다. …

Type	Method
Name	ProvideHelp
Purpose	User 에게 도움말 화면을 제공한다.
Cross References	Functions : R 6.3 Use Cases : Provide Help
Input (Method)	-
Output (Method)	-
Abstract Operation (Method)	1. User 에게 도움말 화면을 보여준다.
Exceptional Courses of Events	N/A

Type	Method
Name	Check_true_false
Purpose	User 가 Test Case 조합에 대해 테스팅 결과를 입력할 수 있도록 한다.
Cross References	Functions : R 7.1 Use Cases : Check True/False
Input (Method)	Index : int
Output (Method)	-
Abstract Operation (Method)	1. index 에 해당하는 Result 를 찾는다. 2. 해당 Result 의 check 메소드를 실행한다.
Exceptional Courses of Events	N/A

Type	Method
Name	New_rep_value
Purpose	해당하는 Category 에 하위 Rep. Value 를 생성한다.
Cross References	Functions : R 3.1 Use Cases : New Rep. Value
Input (Method)	-
Output (Method)	-
Abstract Operation (Method)	1. 새로운 Rep. Value 를 생성한다.
Exceptional Courses of Events	N/A
Type	Method
Name	Remove_rep_value
Purpose	해당하는 Category 의 하위 Rep. Value 를 삭제한다.
Cross References	Functions : R 3.2 Use Cases : Remove Rep. Value
Input (Method)	Rindex : int
Output (Method)	-
Abstract Operation (Method)	1. rindex 에 해당하는 Rep. Value 를 삭제한다.
Exceptional Courses of Events	N/A

Type	Method
Name	Modify_rep_value
Purpose	해당하는 Category 의 하위 Rep. Value 의 정보를 변경한다.
Cross References	Functions : R 3.3 Use Cases : Modify Rep. Value
Input (Method)	Rindex : int, name : String
Output (Method)	-
Abstract Operation (Method)	1. rindex 에 해당하는 Rep. Value 의 이름을 name 으로 수정한다.
Exceptional Courses of Events	N/A

Type	Method
Name	Add_const – Constraints
Purpose	Rep. Value 에 Constraints 를 새로 입력하거나 변경할 때 Constraints 에 해당 Rep. Value 를 저장한다.
Cross References	Functions : R 4.1, R 4.3 Use Cases : New Constraints, Modify Constraints
Input (Method)	Repv : RepValue, type : String
Output (Method)	-
Abstract Operation (Method)	1. type 에 해당하는 List 에 repv 를 추가한다.
Exceptional Courses of Events	N/A
Type	Method
Name	Remove_const – Constraints
Purpose	Rep. Value 에 Constraints 를 변경하거나 삭제할 때, 혹은 Rep. Value 를 삭제할 경우 Constraints 에 해당 Rep. Value 를 삭제한다.
Cross References	Functions : R 3.2, R 4.1, R 4.3 Use Cases : Remove Rep. Value, Remove Constraints, Modify Constraints
Input (Method)	Repv : RepValue, oldtype : String
Output (Method)	-
Abstract Operation (Method)	1. oldtype 에 해당하는 List 에서 repv 를 삭제한다.

Exceptional Courses of Events	N/A

Type	Method
Name	Add_const – Property
Purpose	Rep. Value 에 Property 가 할당된 경우, Property 에 해당 Rep. Value 를 저장한다.
Cross References	Functions : R 4.3 Use Cases : Modify Constraints
Input (Method)	Repv : RepValue, type : String
Output (Method)	-
Abstract Operation (Method)	1. type 에 해당하는 List 에 repv 를 추가한다.
Exceptional Courses of Events	N/A

Type	Method
Name	Remove_const - Property
Purpose	Rep. Value 에 할당되어 있던 Property 가 변경 혹은 삭제된 경우, Property 에 해당 Rep. Value 에 대한 정보를 삭제한다.
Cross References	Functions : R 3.2, R 4.2, R 4.3 Use Cases : Remove Rep. Value, Remove Constraints, Modify Constraints
Input (Method)	Repv : RepValue, oldtype : String
Output (Method)	-
Abstract Operation (Method)	1. oldtype 에 해당하는 List 에서 repv 를 삭제한다.

Exceptional Courses of Events	N/A

Type	Method
Name	Mark_filter - Property
Purpose	User 가 체크한 Filter 를 Property 에 적용한다.
Cross References	Functions : R 6.2.2 Use Cases : Filter Property
Input (Method)	-
Output (Method)	-
Abstract Operation (Method)	1. filter 값을 변경시킨다.
Exceptional Courses of Events	N/A

Type	Method
Name	Add_const – RepValue
Purpose	Rep. Value 에 Property 가 할당된 경우, Rep. Value 에 해당 Property 를 저장한다.
Cross References	Functions : R 4.3 Use Cases : Modify Constraints
Input (Method)	(Repv : RepValue), type : String
Output (Method)	-
Abstract Operation (Method)	1. type 에 해당하는 cst 값을 true로 변경한다.
Exceptional Courses of Events	E1. Repv 인자가 같이 주어진 경우 1. type 에 해당하는 List 에 해당 Property 를 추가한다.

Type	Method
Name	Remove_const - RepValue
Purpose	Rep. Value 에 할당되어 있던 Property 가 변경 혹은 삭제된 경우, Rep. Value 에 해당 Property 를 삭제한다.
Cross References	Functions : R 3.2, R 4.2, R 4.3 Use Cases : Remove Rep. Value, Remove Constraints, Modify Constraints
Input (Method)	(Repv : RepValue), type : String
Output (Method)	-
Abstract Operation (Method)	1. type 에 해당하는 cst 값을 false 로 변경한다.
Exceptional Courses of Events	E1. Repv 인자가 같이 주어진 경우 1. type 에 해당하는 List 에서 해당 Property 를 삭제한다.

Type	Method
Name	Mark_filter - RepValue
Purpose	User 가 체크한 Filter 를 Rep Value 에 적용한다.
Cross References	Functions : R 6.2.1 Use Cases : Filter Rep. Value
Input (Method)	-
Output (Method)	-
Abstract Operation (Method)	1. filter 값을 변경시킨다.
Exceptional Courses of Events	N/A

Type	Method
Name	isCompatible
Purpose	Rep. Value 에 Constraints 를 추가 혹은 변경할 때 정상적인 입력인 지를 확인한다.
Cross References	Functions : R 4.1, R 4.3 Use Cases : New Constraints, Modify Constraints
Input (Method)	-
Output (Method)	Boolean
Abstract Operation (Method)	1. 현재 입력된 constraints 데이터와 새로 입력된 constraints 데이터가 충돌하는 지 여부를 검사한다. 2. 충돌 여부를 리턴한다.
Exceptional Courses of Events	N/A

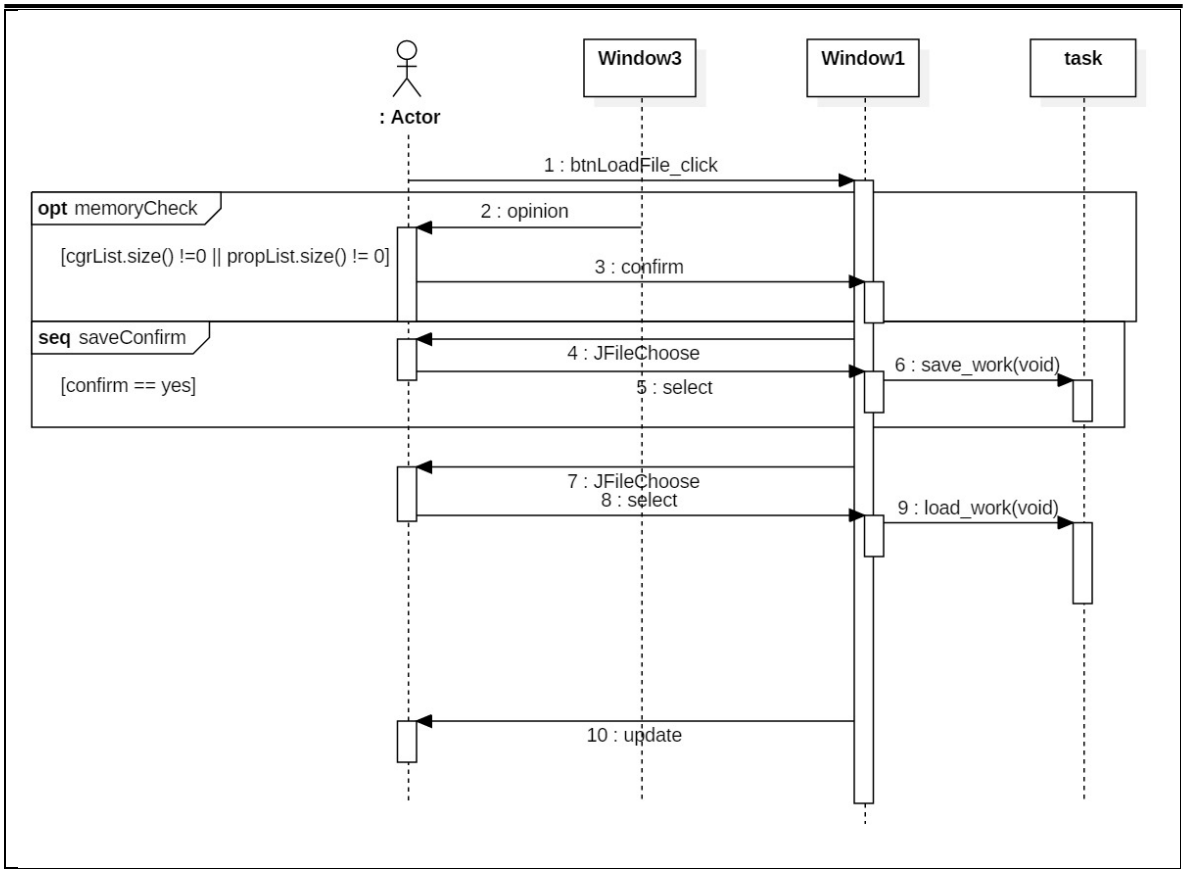
Type	Method
Name	New_resultList
Purpose	Test Case 계산할 때, 새로운 Test Case 조합 결과를 Result에 저장한다.
Cross References	Functions : R 6.1, R 6.2 Use Cases : Calculate, Show Result
Input (Method)	Repv : RepValue, prop : Property
Output (Method)	-
Abstract Operation (Method)	1. repv 혹은 prop 을 해당하는 List 에 추가한다.
Exceptional Courses of Events	N/A

Type	Method

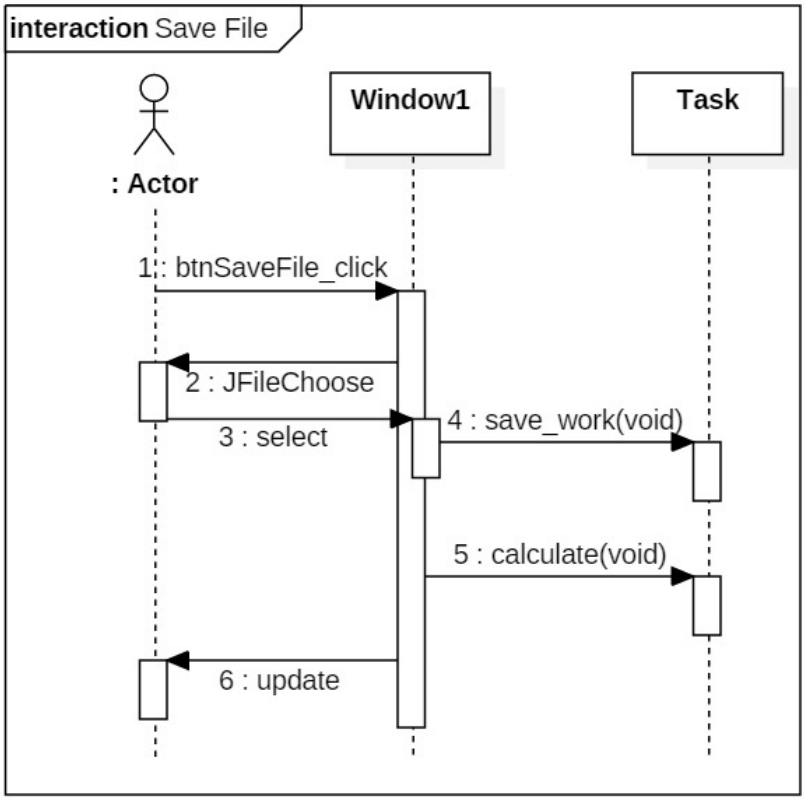
Name	Check
Purpose	User 가 Test Case 수행 결과를 저장할 수 있도록 한다.
Cross References	Functions : R 7.1 Use Cases : Check True/False
Input (Method)	-
Output (Method)	-
Abstract Operation (Method)	1. caseFlag 값을 변경시킨다.
Exceptional Courses of Events	N/A

Activity 2052. Implement Windows

Name	BtnLoadFile_click
Responsibilities	Load File 버튼을 누른다.
Type	GUI
Cross References	Functions : R 1.1 Use Cases : Load File
Notes	Load File 버튼을 눌러 저장할 파일을 선택 후 load_work 메소드를 호출한다.
Pre-Conditions	-
Post-Conditions	선택한 파일의 내용을 메모리로 불러온다.



Name	btnSaveFile_click
Responsibilities	Save File 버튼을 누른다.
Type	GUI
Cross References	Functions : R 1.2 Use Cases : Save File
Notes	Save File 버튼을 눌러 저장할 파일을 선택 후 Save_work 메소드를 호출한다.
Pre-Conditions	-
Post-Conditions	선택한 파일에 메모리 내용을 기록한다.



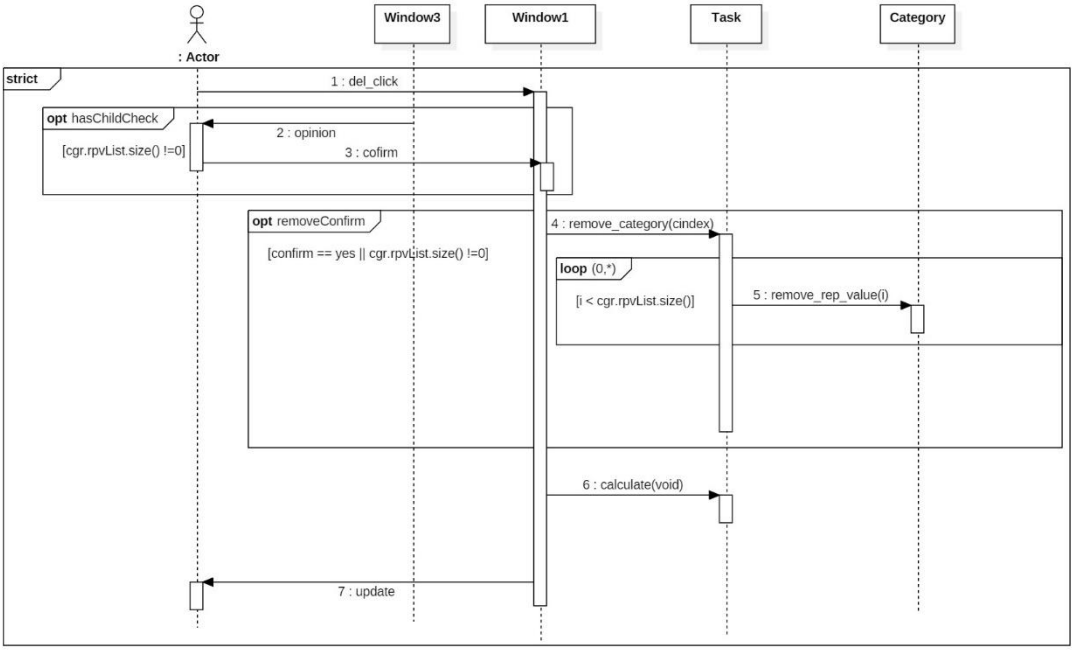
- **ComponentSet의 GUI 컴포넌트**

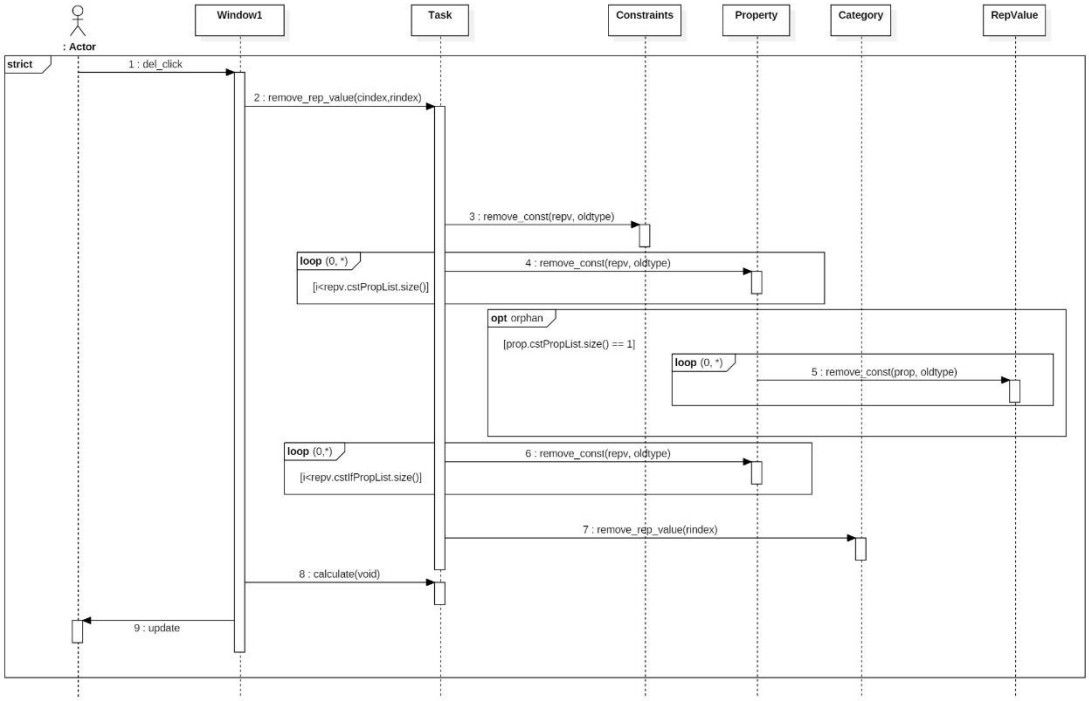
Name	Sel_click
Responsibilities	객체의 이름 옆의 체크모양 버튼을 누른다.
Type	GUI
Cross References	-
Notes	객체를 선택하여 이 객체가 공유하는 다른 관계에 대해 추가/제거 밑 수정이 가능하다.
Pre-Conditions	-

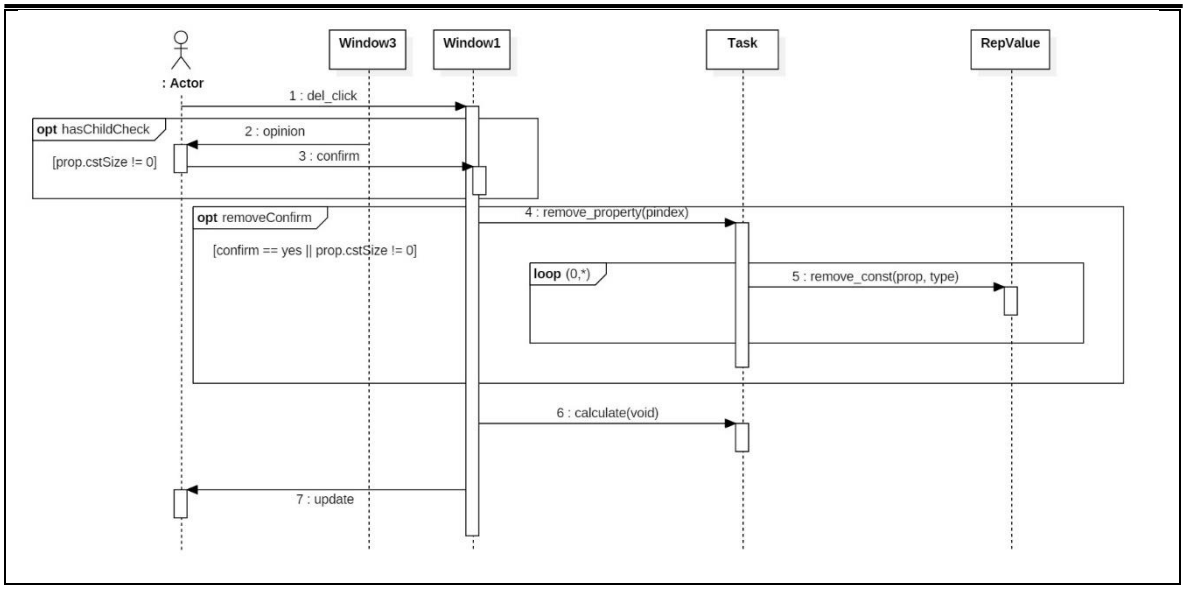
Post-Conditions	객체가 선택된다.

Name	Del_click
Responsibilities	객체의 이름 옆의 휴지통 버튼을 누른다.
Type	GUI
Cross References	Functions : R 2.2, R 3.2, R 5.2 Use Cases : Remove Category, Remove Rep. Value, Remove Property
Notes	객체를 제거한다.
Pre-Conditions	-
Post-Conditions	객체가 리스트에서 제거된다. 다른 객체의 관계도

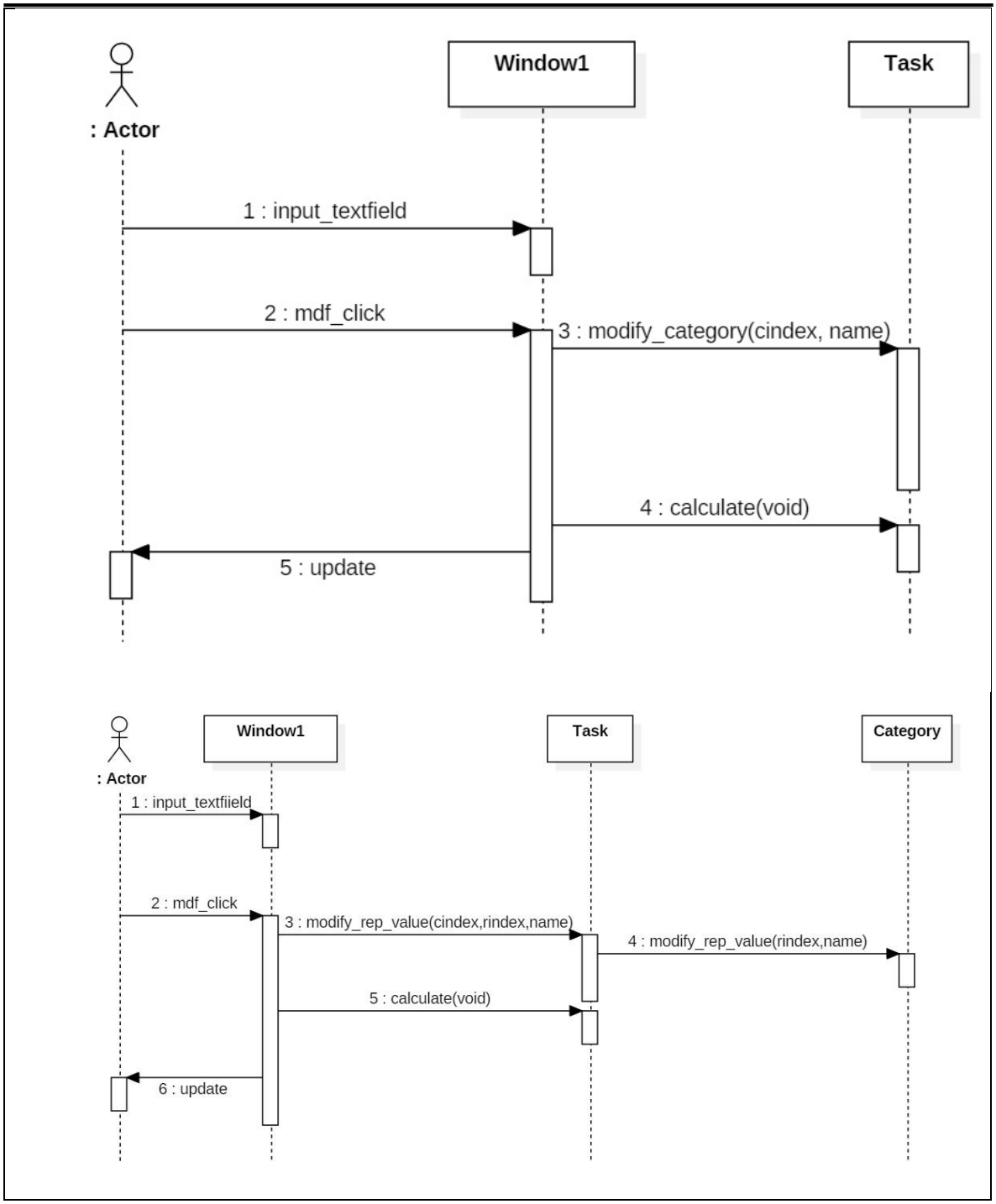
	변경된다면 이에 대한 의사확인 후 제거된다.

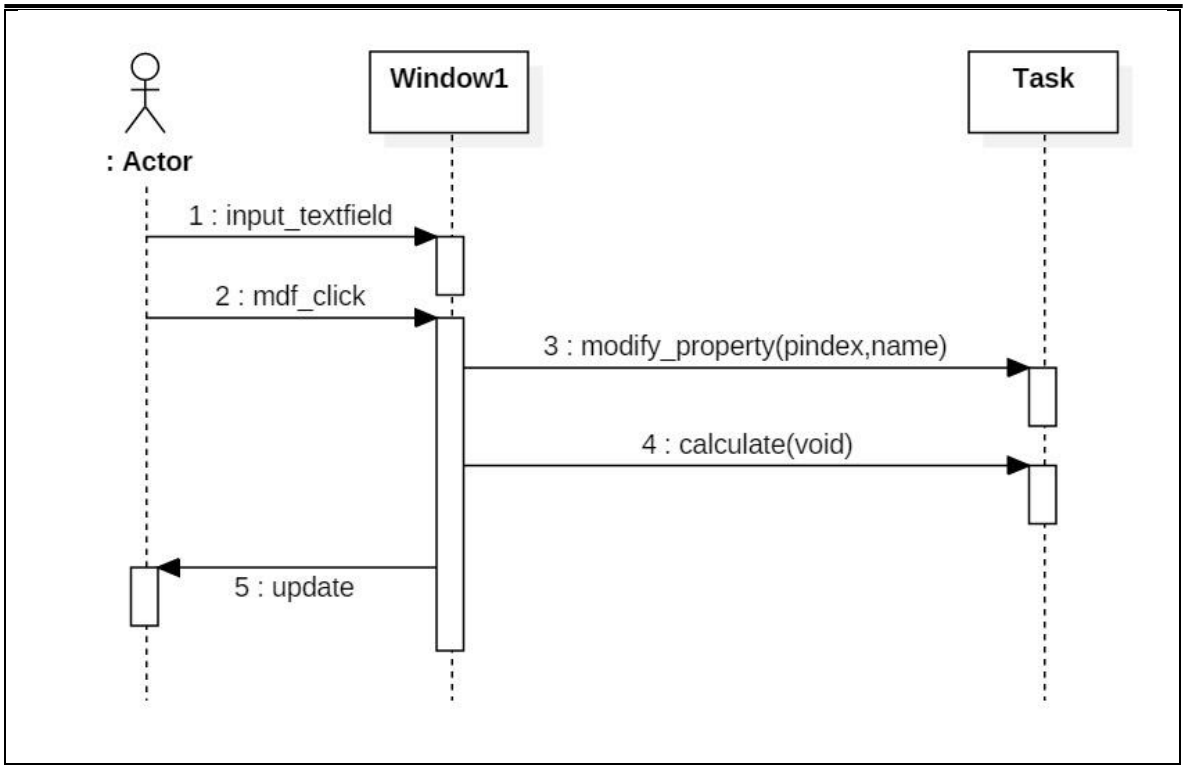






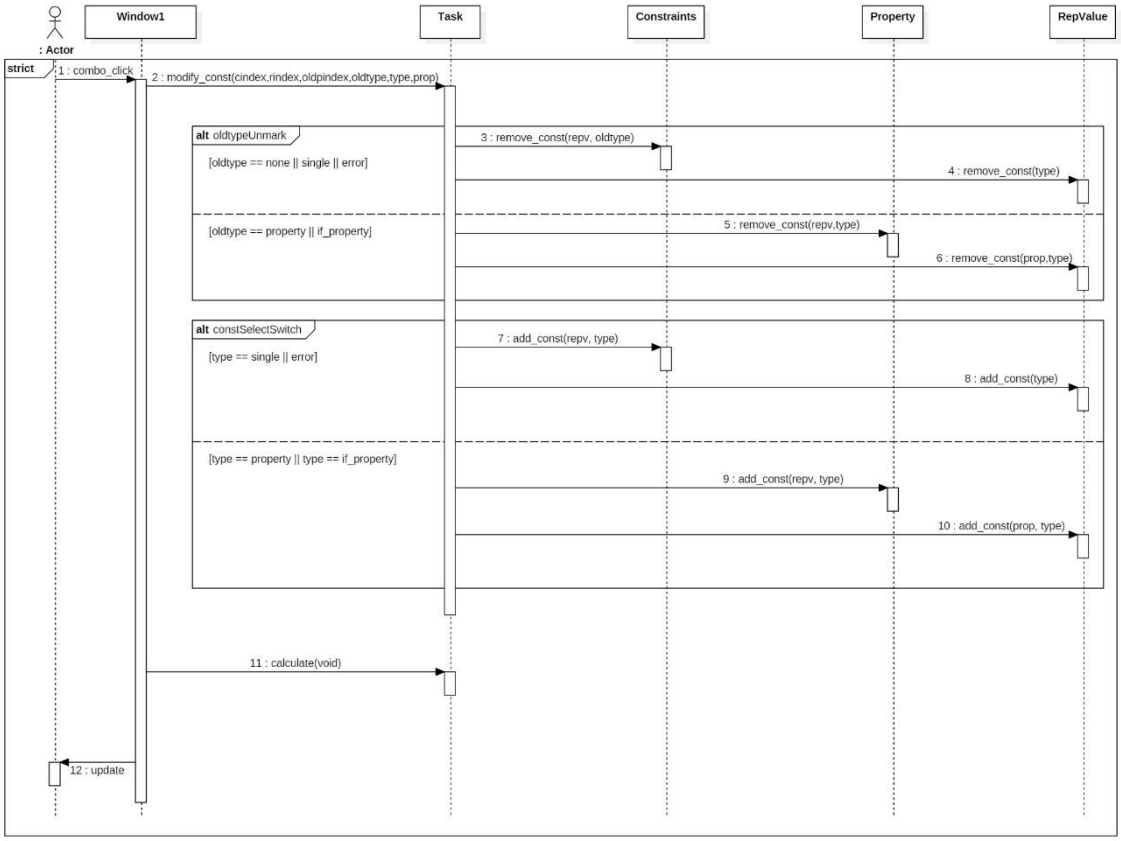
Name	Mdf_click
Responsibilities	객체 옆의 렌치 모양 버튼을 누른다.
Type	GUI
Cross References	Functions : R 2.3, R 3.3, R 5.3 Use Cases : Modify Category, Modify Rep. Value, Modify Property
Notes	객체의 이름을 수정한다.
Pre-Conditions	-
Post-Conditions	현재 textField에 입력된 이름으로 객체의 이름을 수정한다.


: Actor
Window1
Task
1 : input_textfield
2 : mdf_click
3 : modify_category(cindex, name)
4 : calculate(void)
5 : update
: Actor
Window1
Task
Category
1 : input_textfiield
2 : mdf_click
3 : modify_rep_value(cindex,rindex,name)
4 : modify_rep_value(rindex,name)
5 : calculate(void)
6 : update



Name	combo_click
Responsibilities	제약관계 이름 옆의 콤보 버튼을 조작한다.
Type	GUI
Cross References	Functions : R 4.3 Use Cases : Modify Constraints
Notes	제약 관계를 결정한다.
Pre-Conditions	-
Post-Conditions	기존의 제약 관계가 제거되고 선택한 제약 관계가

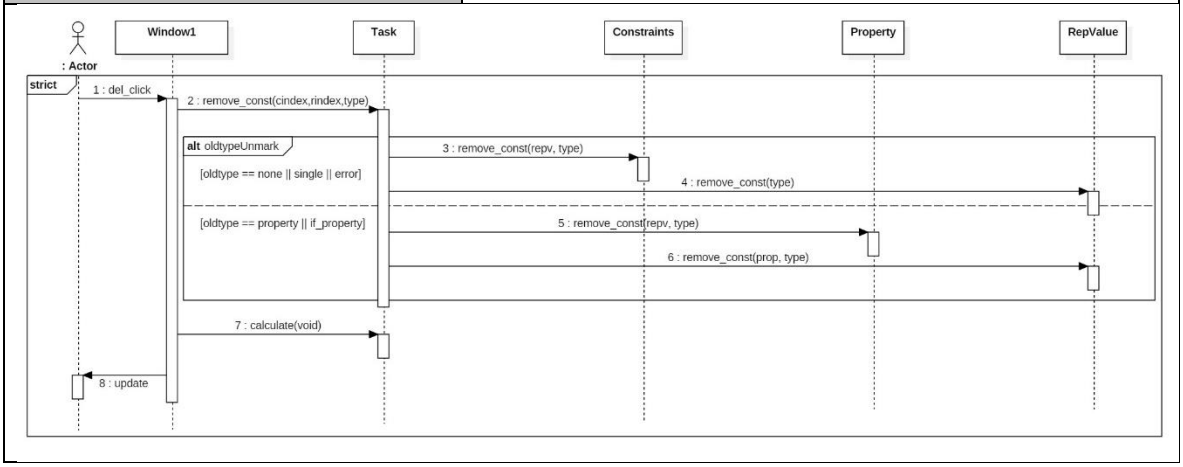
	새로 생성된다.



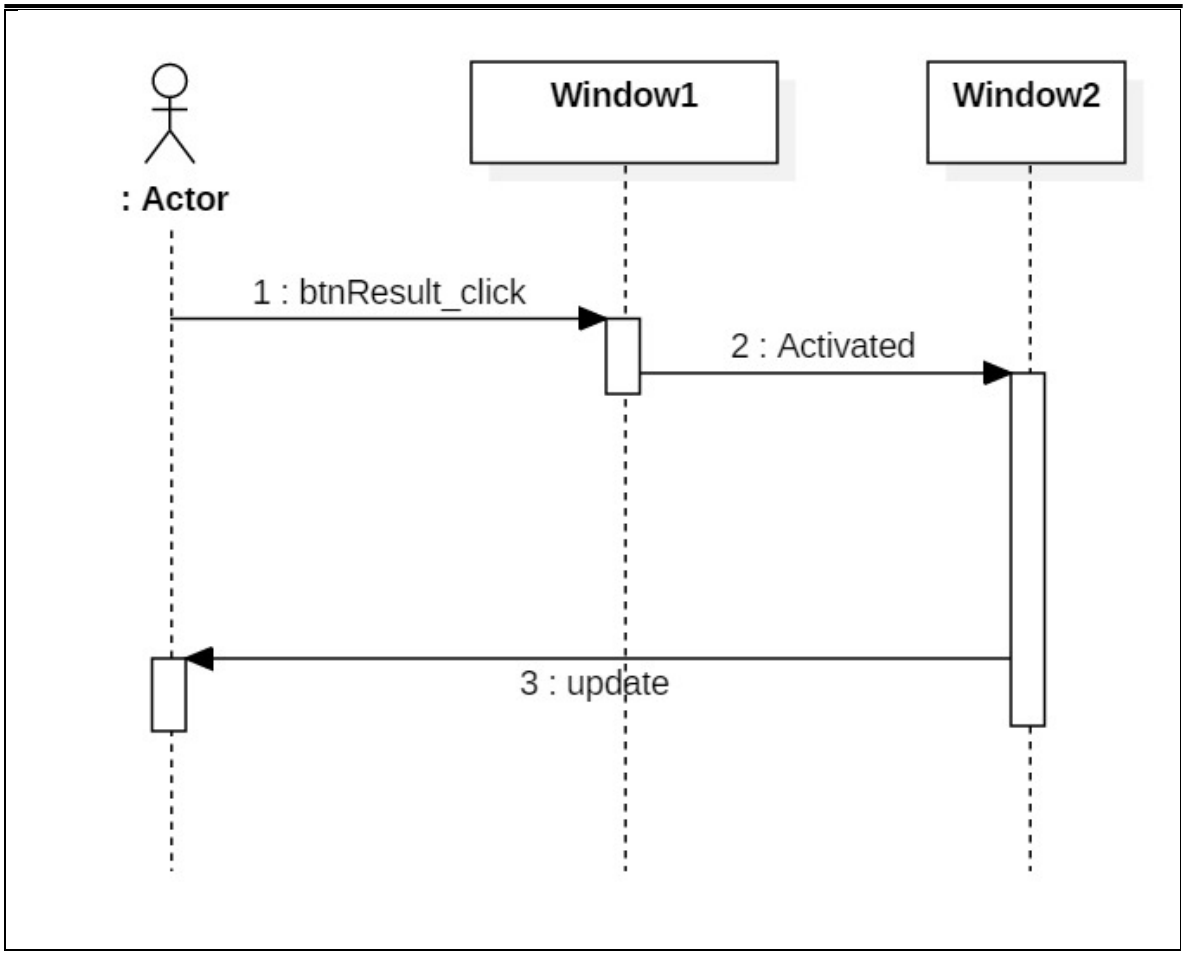
Name	Sel
Responsibilities	제약 관계 이름 옆의 체크 모양 버튼을 누른다.
Type	GUI
Cross References	Functions : - Use Cases : -
Notes	구체적으로 어떤 Property와 연결되는지 설정할 제약관계를 선택한다.
Pre-Conditions	제약 관계가 Property, IF_Property이다.

Post-Conditions	제약 관계가 선택된다.

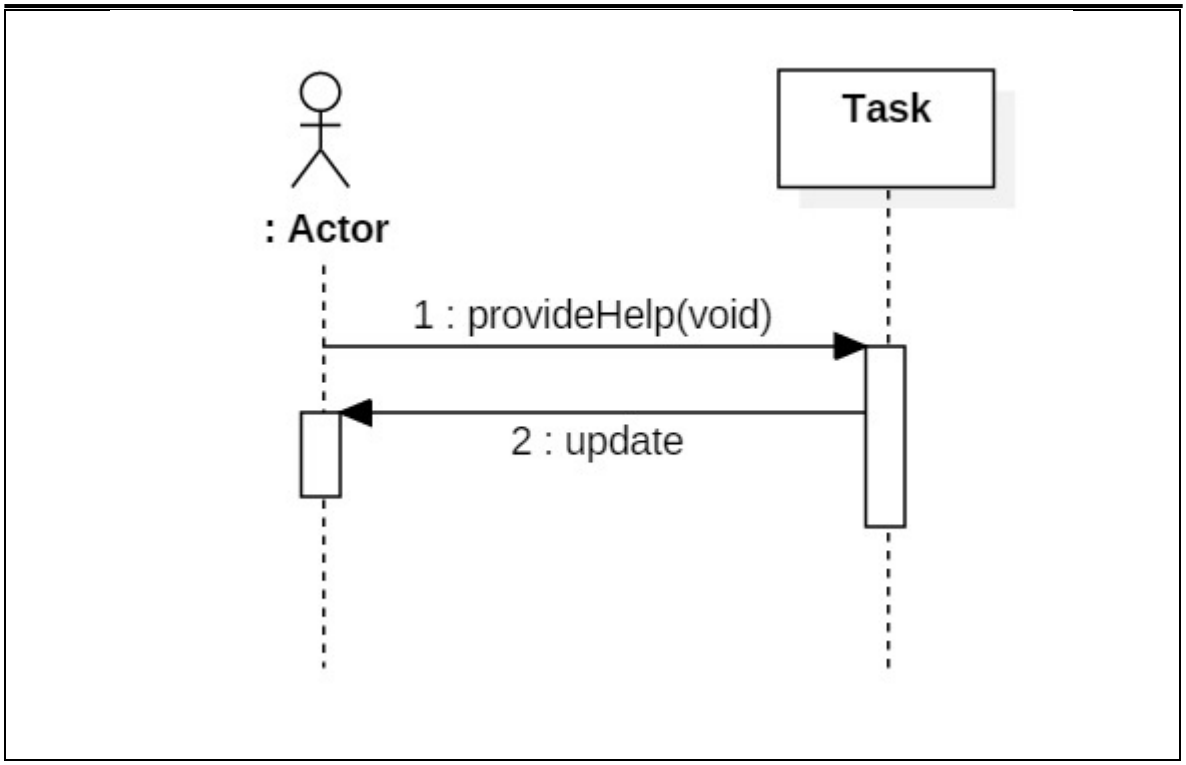
Name	Del
Responsibilities	제약 관계 이름 옆의 휴지통 모양 버튼을 누른다.
Type	GUI
Cross References	Functions : R 4.2 Use Cases : Remove Constraints
Notes	제약 관계를 제거한다.
Pre-Conditions	-
Post-Conditions	제약 관계가 제거된다.



Name	btnResult_click
Responsibilities	RESULT라 적힌 버튼을 누른다.
Type	GUI
Cross References	Functions : R 6.2 Use Cases : Show Result
Notes	결과창으로 넘어간다.
Pre-Conditions	-
Post-Conditions	작업창이 비활성화되고 결과창이 보인다.



Name	provideHelp_click
Responsibilities	HELP라 적힌 버튼을 누른다.
Type	GUI
Cross References	Functions : R 6.4 Use Cases : Provide Help
Notes	결과창으로 넘어간다.
Pre-Conditions	-
Post-Conditions	작업창이 비활성화되고 결과창이 보인다.


Task
: Actor
1 : provideHelp(void)
2 : update

Activity 2055. Write Unit Test Code

- Category_Test

```
@Test
public void testNew_rep_value() {
        Category category = new Category("test",0);
        for(int i=0;i<100;i++)
        {
                category.new_rep_value();
                assertEquals(i+1,category.getRpvListSize());
        }
        //fail("Not yet implemented");
}

@Test
public void testRemove_rep_value() {
        Category category = new Category("test",0);
        for(int i=0;i<100;i++)
                category.new_rep_value();

        for(int i=0;i<100;i++)
        {
                category.remove_rep_value(0);
        }
        assertEquals(0,category.getRpvListSize());
        //fail("Not yet implemented");
}

@Test
public void testModify_rep_value() {
        Category category = new Category("test",0);
        for(int i=0;i<100;i++)
                category.new_rep_value();
        for(int i=0;i<100;i++)
                category.modify_rep_value(i, "test" + i);
        for(int i=0;i<100;i++)
```

```
                    assertEquals("test" + i,category.getRepValue(i).getName());
    }
```

- **Property_Test**

```
    @Test
    public void testAdd_const() {
            Property ptemp = new Property("test");
            RepValue rtemp = new RepValue(0,0);
            ptemp.add_const(rtemp, "Property");
            ptemp.add_const(rtemp, "IFProperty");
            assertEquals(rtemp,ptemp.getConst(0, "Property"));
            assertEquals(rtemp,ptemp.getConst(0, "IFProperty"));
            //fail("Not yet implemented");
    }

    @Test
    public void testRemove_const() {
            //fail("Not yet implemented");
            Property ptemp = new Property("test");
            RepValue rtemp = new RepValue(0,0);
            ptemp.add_const(rtemp, "Property");
            ptemp.add_const(rtemp, "IFProperty");
            ptemp.remove_const(rtemp, "Property");
            ptemp.remove_const(rtemp, "IFProperty");
            assertEquals(0,ptemp.getCstSize("Property"));
            assertEquals(0,ptemp.getCstSize("IFProperty"));
    }

    @Test
    public void testMark_filter() {
            //fail("Not yet implemented");
            Property ptemp = new Property("test");
            ptemp.mark_filter();
            assertEquals(true,ptemp.isFilter());
            ptemp.mark_filter();
            assertEquals(false,ptemp.isFilter());
    }
```

- RepValue_Test

```
@Test
public void testAdd_const1() {
        RepValue rtest = new RepValue(0,0);
        rtest.add_const("None");
        rtest.add_const("Error");
        rtest.add_const("Single");

        assertEquals(true,rtest.getConst("None"));

        //fail("Not yet implemented");
}

@Test
public void testAdd_const2() {
        RepValue rtest = new RepValue(0,0);
        Property ptest = new Property("test");
        rtest.add_const(ptest, "Property");
        rtest.add_const(ptest,"IFProperty");
        assertEquals(ptest,rtest.getConst(0, "Property"));
        assertEquals(ptest,rtest.getConst(0, "IFProperty"));

        //fail("Not yet implemented");
}

@Test
public void testRemove_const1() {
        RepValue rtest = new RepValue(0,0);
        rtest.add_const("None");
        rtest.add_const("Error");
        rtest.add_const("Single");
        rtest.remove_const("None");
        rtest.remove_const("Error");
        rtest.remove_const("Single");
        assertEquals(false,rtest.getConst("None"));
        assertEquals(false,rtest.getConst("Error"));
```

```
		assertEquals(false,rtest.getConst("Single"));
		//fail("Not yet implemented");
	}

	@Test
	public void testRemove_const2() {
		RepValue rtest = new RepValue(0,0);
		Property ptest = new Property("test");
		rtest.add_const(ptest, "Property");
		rtest.add_const(ptest,"IFProperty");
		rtest.remove_const(ptest, "Property");
		rtest.remove_const(ptest, "IFProperty");
		assertEquals(0,rtest.getCstSize("Property"));
		assertEquals(0,rtest.getCstSize("IFProperty"));
		//fail("Not yet implemented");
	}

	@Test
	public void testMark_filter() {
		RepValue rtest = new RepValue(0,0);
		rtest.mark_filter();
		assertEquals(true,rtest.isFilter());
		rtest.mark_filter();
		assertEquals(false,rtest.isFilter());
		//fail("Not yet implemented");
	}
```

- Task_Test

```
	@Test
	public void testNew_category() {
		Task task = new Task();
		for(int i=0;i<100;i++)
		{
			task.new_category("test"+i);
			assertEquals(i+1,task.getCgrListSize());
		}
		//fail("Not yet implemented");
```

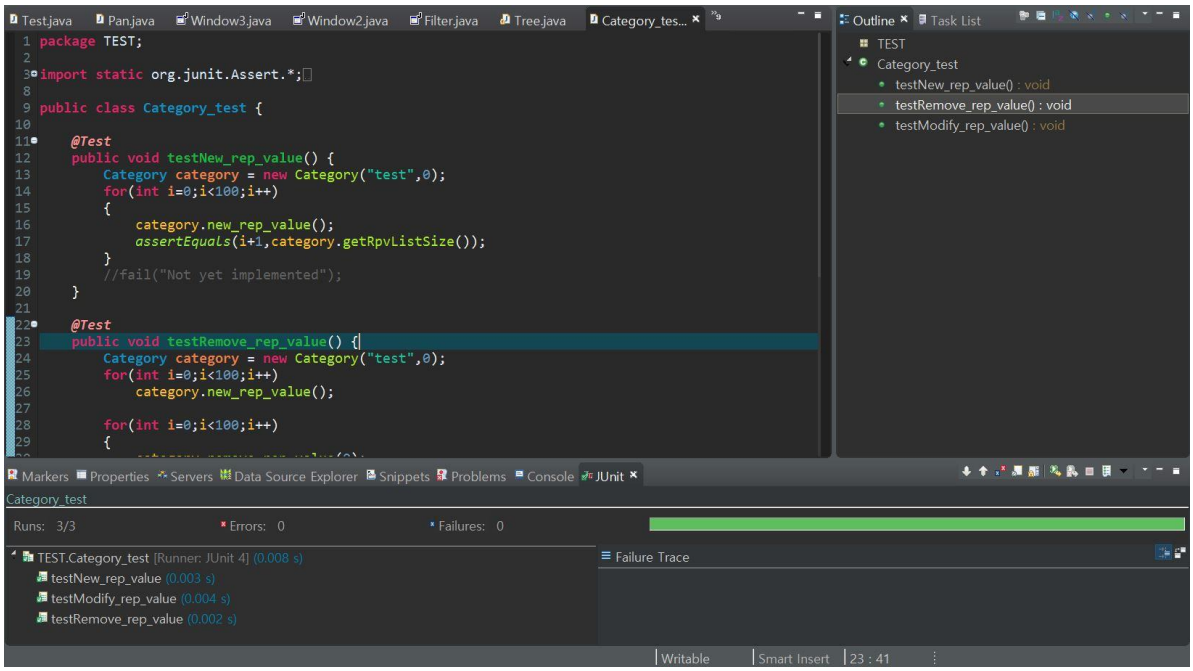
```
}

@Test
public void testRemove_category() {
	//fail("Not yet implemented");
	Task task = new Task();
	for(int i=0;i<100;i++)
	{
		task.new_category("test"+i);
	}
	for(int i=0;i<100;i++)
	{
		task.remove_category(0);
	}
	assertEquals(0,task.getCgrListSize());
}

@Test
public void testModify_category() {
	//fail("Not yet implemented");
	Task task = new Task();

	for(int i=0;i<100;i++)
		task.new_category("test"+i);
	for(int i=0;i<100;i++)
		task.modify_category(i, "itest"+i);
	for(int i=0;i<100;i++)
		assertEquals("itest" + i,task.getCategory(i).getName());
}

@Test
public void testNew_property() {
	//fail("Not yet implemented");
	Task task = new Task();
	for(int i=0;i<100;i++)
	{
```

```
                task.new_property("test"+i);
                assertEquals(i+1,task.getPropListSize());
        }
}

@Test
public void testRemove_property() {
        //fail("Not yet implemented");
        Task task = new Task();
        for(int i=0;i<100;i++)
        {
                task.new_property("test"+i);
        }
        for(int i=0;i<100;i++)
        {
                task.remove_property(0);
        }
        assertEquals(0,task.getPropListSize());
}

@Test
public void testModify_property() {
        //fail("Not yet implemented");
        Task task = new Task();

        for(int i=0;i<100;i++)
                task.new_property("test"+i);
        for(int i=0;i<100;i++)
                task.modify_property(i, "itest"+i);
        for(int i=0;i<100;i++)
                assertEquals("itest" + i,task.getProperty(i).getName());
}
```

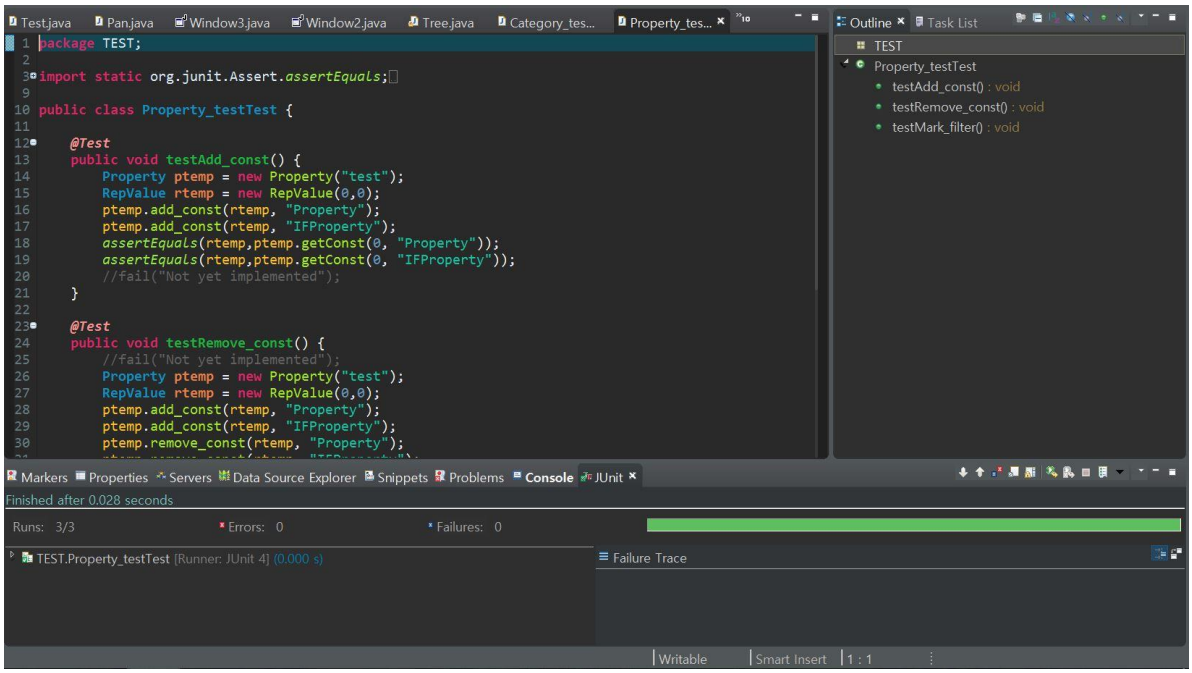
Activity 2060. Test

Activity 2061. Unit Testing

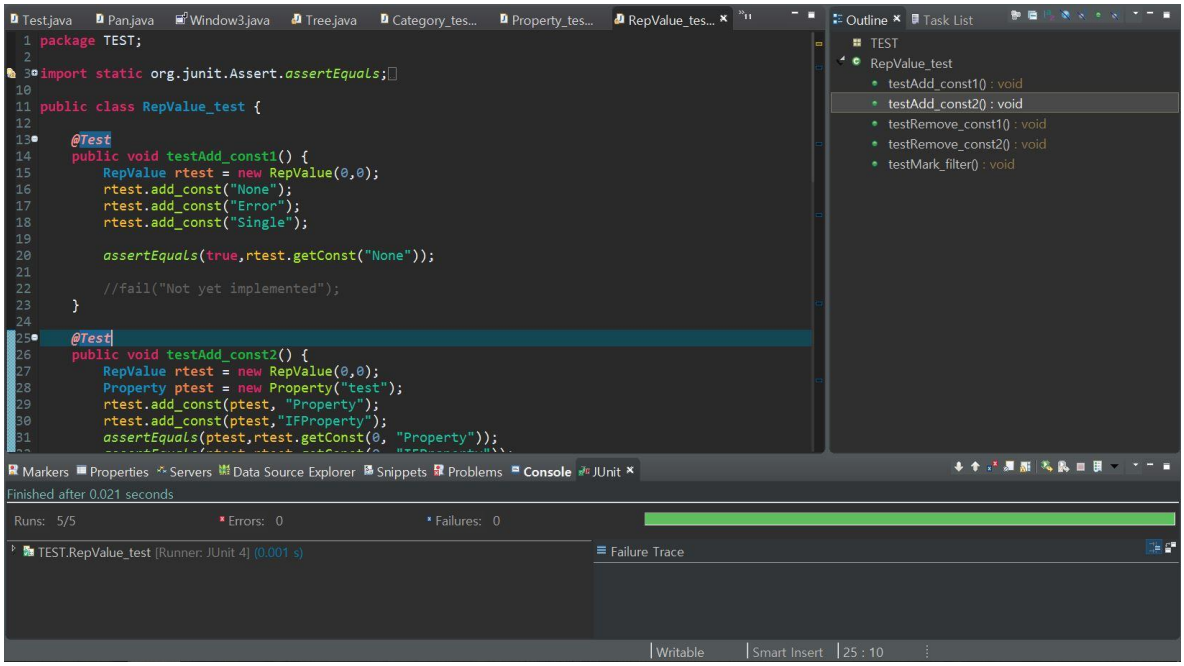
- Category_Test



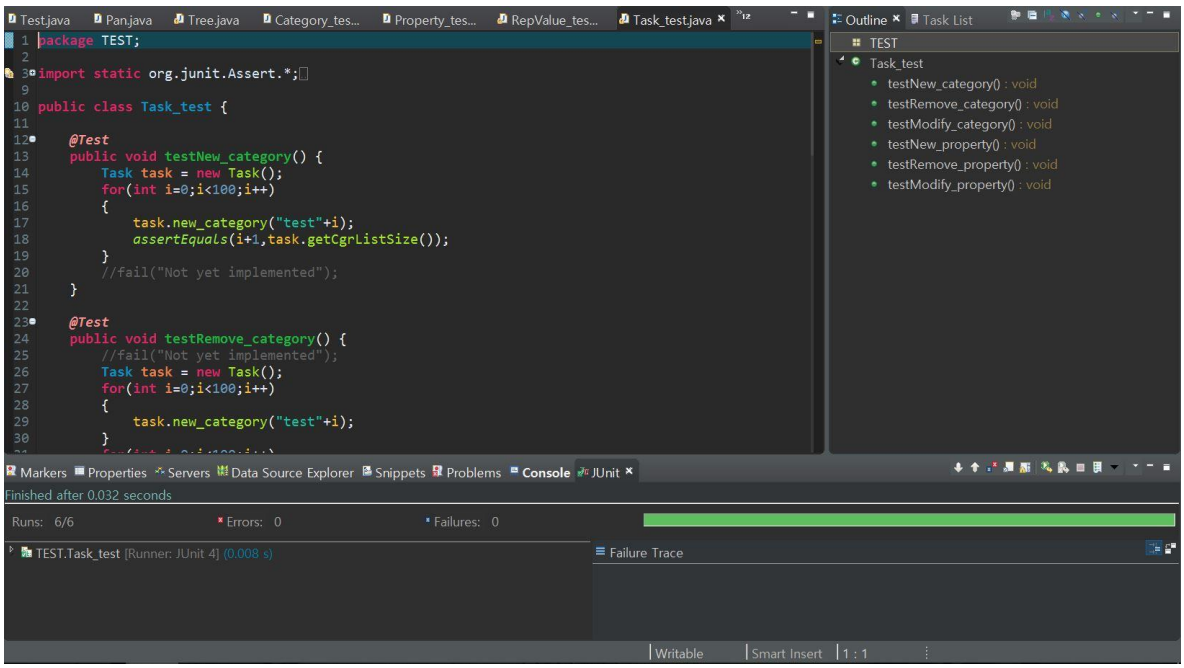
- Property_Test



- RepValue_Test



- Task_Test



Activity 2063. System Testing

Test Number	Test 항목	Description	Use Case	System Function	Pass / Fail
1.1	작업 파일 불러오기 시험	작업이 로드 되지 않은 상태에서 파일이 로드되는 지 확인한다.	Load File	R1.1	P
1.2	작업 파일 불러오기 시험	작업이 로드 된 상태에서 파일이 로드되는 지 확인한다.	Load File	R1.1	P
1.3	작업 파일 불러오기 시험	존재하지 않는 파일에 대해 로드한 경우 에러를 표시하는 지 확인한다.	Load File	R1.1	P
2.1	작업중이던 파일 저장 시험	작업이 진행 되지 않은 상태에서 파일이 저장되는 지 확인한다.	Save File	R1.2	P
2.2	작업중이던 파일 저장 시험	작업이 로드 된 상태에서 파일이 저장되는 지 확인한다.	Save File	R1.2	P
2.3	작업중이던 파일 저장 시험	유효하지 않은 경로에 대해 파일을 저장하는 경우 에러를 표시하는 지 확인한다.	Save File	R1.2	P
3.1	새로운 카테고리 생성 시험	이름을 입력한 상태로 카테고리를 생성하는 지 확인한다.	New Category	R2.1	P
3.2	새로운 카테고리 생성 시험	아무 이름도 입력하지 않은 상태에서 카테고리를 생성하는 지 확인한다.	New Category	R2.1	P
4.1	카테고리 삭제 시험	카테고리 삭제가 정상적으로 수행되는 지 확인한다.	Remove Category	R2.2	P
5.1	카테고리 이름 수정 시험	카테고리 이름을 수정	Modify Category	R2.3	P
6.1	새로운 Rep. Value 생성 시험	Rep. Value 생성이 되는 지 확인한다.	New Representative Value	R3.1	P
7.1	Rep. Value 삭제 시험	Rep. Value 삭제가 되는 지 확인한다.	Remove Representative	R3.2	P

			Value		
8.1	Rep. Value 이름 수정 시험	Rep. Value 의 이름이 수정되는 지 확인한다.	Modify Representative Value	R3.3	P
9.1	새로운 제약조건 생성 시험	새로운 Constraints 가 생성되어 None 값을 갖는 지 확인한다.	New Constraints	R 4.1	P
10.1	제약조건 삭제 시험	Constraints가 정상적으로 삭제되는 지 확인한다.	Remove Constraints	R 4.2	P
11.1	제약조건 항목 수정 시험	Constraints의 항목이 수정되는 지 확인한다.	Modify Constraints	R 4.3	P
12.1	새로운 Property 생성 시험	새로운 Property 가 정상적으로 생성되는 지 확인한다.	New Property	R 5.1	P
13.1	Property 삭제 시험	Property가 정상적으로 삭제되는 지 확인한다.	Remove Property	R 5.2	P
14.1	Property 이름 수정 시험	Property 의 이름이 수정되는 지 확인한다.	Modify Property	R 5.3	P
15.1	Calculate 시험	입력된 정보를 바탕으로 Test Case 계산이 정상적으로 수행되는 지 확인한다.	Calculate	R 6.1	F
16.1	결과 화면 보여주기 시험	수행중인 작업이 있는 상태에서 Test Case 생성 확인	Show Result	R 6.2	P
16.2	결과 화면 보여주기 시험	수행중인 작업이 없는 상태에서 Test Case 생성 확인	Show Result	R 6.2	P
17.1	필터링할 대푯값 설정 시험	필터링할 Representative Value 선택했을 때 결과 화면이 정상적으로 변하는 지 확인한다.	Filter Representative Value	R 6.2.1	P
17.2	필터링할 대푯값 설정 시험	필터링할 Representative Value 선택을 해제했을 때 결과 화면이 정상적으로 변하는 지 확인한다.	Filter Representative Value	R 6.2.1	P
18.1	필터링할	필터링할 Property 선택했을	Filter Property	R 6.2.2	P

	Property 설정 시험	때 결과 화면이 정상적으로 변하는 지 확인한다.			
18.2	필터링할 Property 설정 시험	필터링할 Property 선택을 해제했을 때 결과 화면이 정상적으로 변하는 지 확인한다.	Filter Property	R 6.2.2	P
19.1	도움말 제공 기능 시험	작업 화면에서 도움말 버튼을 클릭했을 때 도움말 화면이 정상적으로 표시되는 지 확인한다.	Provide Help	R 6.3	P
20.1	Test Case 수행 결과 입력 시험	Show Result 화면에서 Test Case 수행 결과가 입력되는 지 확인한다.	Check True/False	R 7.1	P

Activity 2067. Testing Traceability Analysis