Unit Test Report

T6

201011311 김도희

201111353 박수민

+ 김태현

1. Test Environment

Date : 2015.05.21

Version : 1.0

OS : Windows 8.1 64bits

2. Test Result

2.1 Game - init()

1	Name="testInitGame"	input	output	result
1-1	assertArrayEquals(voidanswer,game.word) init()실행후 단어 목록이 초기화되었는지 확인한다.	voidans	word[]	PASS
1-2	assertArrayEquals(voidloc,game.loc[0]) init()실행후 단어 x위치 정보가 초기화되었는지 확인한다.	voidloc	loc[][0]	PASS
1-3	assertArrayEquals(voidloc,game.loc[1]) init()실행후 단어 y위치 정보가 초기화되었는지 확인한다.	voidloc	loc[][1]	PASS
1-4	assertArrayEquals(voidloc,game.loc[2])	voidloc	loc[][2]	PASS

2.2 Game - setGame()

2	Name="testSetGame"	input	output	result
2-1	assertEquals(5,game.getTitle()) game을 초기화하고 설정되어 있는 title이 설정한 주제와 일치하는지 확인	5	int	PASS
2-2	assertArrayEquals(testanswer5,game.word) game에 설정되어 있는 word가 title에 맞는 단어 리스트인지 확인	String[]	String[]	PASS
2-3	assertNotNull(game.hint(index)); hint의 이미지 주소를 리턴하는지 확인	index	String	PASS
2-4	assertEquals(loc[index][2],game.word[index].length()) game에 설정되어 있는 임의의 word의 길이와 단어 길이 정보가 일치하는지 확인	index	int	PASS
2-5	assertNotNull(game.answer[(game.answerNum()-1)])); answer이 정해진 개수만큼 세팅되어 있는지 확인	index	String	PASS

2.3 Game - checkAnswer()

3	Name="testRunGame"	input	output	result
3-1	assertTrue(checkAnswer(index,game.word[index])) 게임의 정답과 입력한 단어가 일치할 때 true 리턴	index	TRUE	PASS
3-2	assertTrue(game.checkAnswer(index,game.word[index].toUpperCase())) 게임의 정답과 대문자로 입력한 단어가 일치할 때 true 리턴	index	TRUE	FAIL
3-3	assertFalse(checkAnswer(0,"f low er")) 게임의 정답과 입력한 단어에서 공백을 제외한 단어가 일치할 때 true 리턴	index	TRUE	PASS
3-4	assertFalse(checkAnswer(index,"wrongAns")) 게임의 정답과 입력한 단어가 다를 때 false 리턴	index	FALSE	PASS

2.4 FindWord()

4	Name="testFindWord"	input	output	result
4-1	assertTrue(find.isAlpha("a")) a-z의 대/소문자 알파벳 입력시 true 리턴	a	TRUE	PASS
4-2	assertFalse(find.isAlpha("abc")) 길이 2 이상의 문자열 입력시 false 리턴	abc	FALSE	PASS
4-3	assertFalse(find.isAlpha("")) 아무것도 입력하지 않았을 시 false 리턴		FALSE	PASS
4-4	assertFalse(find.isAlpha("!")) 특수문자 입력시 false 리턴	!	FALSE	PASS
4-5	assertFalse(find.isalpha(inputWord)); 공백 문자 입력시 false 리턴		FALSE	PASS
4-6	assertTrue(find.isWord("apple")) a-z의 알파벳 조합으로 이루어진 문자열 입력시 true 리턴	apple	TRUE	PASS
4-7	assertFalse(find.isWord("ds!@3")) 특수문자, 숫자 등이 포함된 문자열 입력시 false 리턴	ds!@3	FALSE	PASS
4-8	assertFalse(find.isWord("")) 아무것도 입력하지 않았을 시 false 리턴		FALSE	PASS
4-9	assertFalse(find.isWord("apple11")); a-z의 알파벳과 숫자 조합으로 이루어진 문자열 입력시 false 리턴	apple11	FALSE	PASS
4-10	assertFalse(find.isWord(" ")); 공백 문자 입력시 false 리턴		FALSE	PASS
4-11	assertEquals("사과",find.searchWord("apple").getKor())	사과	사과	PASS

	데이터베이스에 저장되어 있는 단어 입력시 해당 단어의 getkor()이 입력한 단어의 한글 뜻과 일치하는지 확인한다.			
4-12	assertNull(find.searchWord("ffs")); 데이터베이스에 저장되어 있지 않은 단어 입력시 null을 리턴하는지 확인한다.	ffs	null	PASS
4-13	assertTrue(find.isWord(searchAlpha('a').getEng())) 알파벳 입력시 랜덤하게 리턴되는 단어가 영어단어인지 확인한다.	a	TRUE	PASS
4-14	assertEquals('a',find.searchAlpha('a').getEng().charAt(0)) 알파벳 입력시 랜덤하게 리턴되는 단어가 입력된 알파벳으로 시작하는지 확인한다.	a	a	PASS

2.5 WordDic()

5	Name="testWordDic"	input	output	result
5-1	assertEquals("사과",dic.returnWord("apple").getKor()) 데이터베이스에 저장되어 있는 단어 입력시 해당 단어의 getkor()이 입력한 단어의 한글 뜻과 일치하는지 확인한다.	사과	사과	PASS
5-2	assertEquals("apple",dic.returnWord("apple").getEng()) 데이터베이스에 저장되어 있는 단어 입력시 해당 단어의 getEng()이 입력한 단어와 일치하는지 확인한다.	apple	apple	PASS
5-3	assertNotNull(dic.returnWord("apple").getImg()) 데이터베이스에 저장되어 있는 단어 입력시 해당 단어의 getImg()이 null이 아닌지 확인한다.	apple	String	PASS
5-4	assertNotNull(dic.returnWord("apple").getSound()) 데이터베이스에 저장되어 있는 단어 입력시 해당 단어의 getSound()이 null이 아닌지 확인한다.	apple	String	PASS
5-5	assertNull(dic.returnWord("ffs")); 데이터베이스에 저장되어 있지 않은 단어 입력시 null을 리턴하는지 확인한다.	ffs	null	PASS
5-6	assertNotNull(dic.getImgpath("apple")) 데이터베이스에 저장되어 있는 단어 입력시 해당 단어의 이미지 주소가 Null이 아닌지 확인한다.	apple	String	PASS
5-7	assertNull(dic.getImgpath("ffs")) 데이터베이스에 저장되어 있지 않은 단어 입력시 해당 단어의 이미지 주소가 null인지 확인한다.	ffs	null	FAIL

2.6 Rank - getTitle()

6	Name="testRank"	input	output	result
6-1	assertEqual("색깔",rank.getTitle()) title에 맞는 문자열을 리턴하는지 확인	색깔	색깔	PASS
6-2	assertNull(rank.getTitle()); title 범위에 맞지 않는 숫자가 입력되었을 때 null을 리턴하는지 확인	7	null	PASS

2.7 Rank - compareTo()

7	Name="testRankSort"	input	output	result
7-1	assertTrue(ranklist(0).getScore()>ranklist(1).getScore()) 임의의 ranklist가 내림차순 정렬되는지 확인	ArrayList	TRUE	PASS