

[별첨]

---

# iReport Designer 테스트 케이스

---

2013. 06

# Stack A

순번	대분류	중분류	소분류	시나리오명	시나리오 개요	시나리오 흐름	케이스 번호	케이스	입력데이터	예상결과	결과	오류증상	비고	
1	시작/종료			iReport Designer 실행/종료	iReport Designer 실행/종료	1. iReport Designer 시작한다. 2. iReport Designer를 종료한다.	1	명령 프롬프트에서 명령어를 입력한다.	iReport.sh	명령 프롬프트가 실행되고 UI가 활성화 된다.	pass			
							2	iReport Designer를 시작한다.						
2	Monitoring			모니터링	검색 정보를 모니터링 한다.	모니터링을 시작한다.	1	iReport Designer 실행		실행 중인 정보가 확인된다.	pass			
							2	iReport Designer output을 확인한다.						
3	HBase Connector			Hbase와 연동	Hbase의 데이터를 가져온다.	Hbase와 연동하여 데이터를 가져온다.	1	HbasePlugin-0.5.1.nbm 플러그인을 설치한다.	http://121.162.249.18 8080	Hbase 연동 플러그인이 설치된다	pass			
							2	플러그인 설치 후 iReport Designer 재부팅		설정이 적용되고 재부팅 된다.	pass			
							3	Report Datasource에서 Hbase와 연동 확인		Datasource에 Hbase Connector가 생성된다.	pass			
							4	Hbase와 연결한다.		Hbase Rest 통신으로 접속한다.	pass			
4	Report Inspector			Report Inspector를 사용한다	Report Inspector의 Parameters를 사용하여 다이나믹 쿼리를 생성한다.	Hbase와 연동하여 다이나믹 데이터 검색을 한다.	1	Parameters에서 변수를 생성한다.	{parameter1}	공백 파라미터1이 생성된다.	pass			
							2	생성한 변수명을 Hbase 검색 쿼리에 넣어준다.	"SubstringComparator": { "substr": "\${parameter1}" }	입력값에 따라 결과가 변경되는 쿼리가 생성된다.	pass			
							3	검색을 시작한다.	Seoul	검색시 파라미터1에 들어갈 값을 물어본다.	pass			
							4	검색 결과를 확인한다.		Seoul에 해당하는 정보를 확인 할 수 있다.	pass			
					1	Report Inspector의 Fileds를 사용하여 추가 컬럼을 사용한다.	Report Inspector의 Fileds를 사용하여 추가 컬럼을 사용 한다.	1	Fileds 에서 추가 필드를 생성한다.	\${Filed1}	공백 필드1이 생성된다.	pass		
								2	검색 결과를 Filed에 부여한다.	\${Filed1}	생성된 필드명으로 결과가 정렬된다.	pass		
					1	Report Inspector의 Variables를 사용하여 함수 를 사용한다.	Report Inspector의 Variables를 사용하여 함수를 사용한 다.	1	Variables에서 새 함수를 생성한다.	\$(variable1)	함수가 생성된다.	pass		
								2	variable 함수를 편집한다.	\$(Filed).size() * \$(PAGE COUNT)	검색된 결과 총 카운트 값을 구하는 함수를 작성한 다.	pass		
								3	함수의 결과를 확인한다.		검색값의 최종 카운트가 나온다.	pass		

## Stack B

순번	대분류	중분류	소분류	시나리오명	시나리오 개요	시나리오 흐름	케이스 번호	케이스	입력데이터	예상결과	결과	오류증상	비고
1	시작/종료			iReport Designer 실행/종료	iReport Designer 실행/종료	1. iReport Designer 시작한다. 2. iReport Designer를 종료한다.	1	명령 프롬프트에서 명령어를 입력한다.	iReport_w.exe	명령 프롬프트가 실행되고 UI가 활성화 된다.	pass		
							2	iReport Designer를 시작한다.					
2	Monitoring			모니터링	검색 정보를 모니터링 한다.	모니터링을 시작한다.	1	iReport Designer 실행		실행 중인 정보가 확인된다.	pass		
							2	iReport Designer output을 확인한다.					
3	HBase Connector			Hbase와 연동	Hbase의 데이터를 가져온다.	Hbase와 연동하여 데이터를 가져온다.	1	HbasePlugin-0.5.1.nbm 플러그인을 설치한다.	http://121.162.249.18 8080	Hbase 연동 플러그인이 설치된다	pass		
							2	플러그인 설치 후 iReport Designer 재부팅		설정이 적용되고 재부팅 된다.	pass		
							3	Report Datasource에서 Hbase와 연동 확인		Datasource에 Hbase Connector가 생성된다.	pass		
							4	Hbase와 연결한다.		Hbase Rest 통신으로 접속한다.	pass		
4				Report Inspector를 사용한다	Report Inspector의 Parameters를 사용하여 다이나믹 쿼리를 생성한다.	Hbase와 연동하여 다이나믹 데이터 검색을 한다.	1	Parameters에서 변수를 생성한다.	\$P{parameter1}	공백 파라미터1이 생성된다.	pass		
							2	생성한 변수명을 Hbase 검색 쿼리에 넣어준다.	"SubstringComparator": { "substr": "\$P{parameter1}" }	입력값에 따라 결과가 변경되는 쿼리가 생성된다.	pass		
							3	검색을 시작한다.	Seoul	검색시 파라미터1에 들어갈 값을 물어본다.	pass		
							4	검색 결과를 확인한다.		Seoul에 해당하는 정보를 확인 할 수 있다.	pass		
					Report Inspector의 Fileds를 사용하여 추가 컬럼을 사용한다.	Report Inspector의 Fileds를 사용하여 추가 컬럼을 사용한다.	1	Fileds 에서 추가 필드를 생성한다.	\$F{Filed1}	공백 필드1이 생성된다.	pass		
							2	검색 결과를 Filed에 부여한다.	\$F{Filed1}	생성된 필드명으로 결과가 정렬된다.	pass		
							1	Variables에서 새 함수를 생성한다.	\$V{variable1}	함수가 생성된다.	pass		
							2	variable 함수를 편집한다.	\$F{Filed}.size() * \$V{PAGE COUNT}	검색된 결과 총 카운트 값을 구하는 함수를 작성한다.	pass		
				Report Inspector의 Variables를 사용하여 함수를 사용한다.	Report Inspector의 Variables를 사용하여 함수를 사용한다.	3	함수의 결과를 확인한다.		검색값의 최종 카운트가 나온다.	pass			