

JUnit A Cook's Tour

: JUnit 3.8.x

1.

```

    (Test Infected : Programmers Love Writing Tests, Java Report, July
    1998, Volume 3, Number 7 )
        가
            가
                .
                    JUnit
                        .
                            , 가 가
                                .
                                    .
                                        .
                                            .
                                                .
                                                    .
                                                        .
                                                            .
                                                                .
                                                                    .
                                                                        .
                                                                            .
                                                                                .
                                                                                    .
                                                                                        .
                                                                                            .
                                                                                                .
                                                                                                    .
                                                                                                        .
                                                                                                            .
                                                                                                                .
                                                                                                                    .
                                                                                                                        .
                                                                                                                            .
                                                                                                                                .
                                                                                                                                    .
                                                                                                                    가
                                                                                                                        ( , )
                                                                                                                            .
                                                                                                                                .
                                                                                                                                    .
                                                                                                                    가

```

2.

```

JUnit
    ?
    , 가 가 가
    가 가 가
    가 가
    가 가
    가 가 , ,
    , PM !! ?
    가 가
    .
    .
    가
    1
    .
    5
    .
    가
    가

```

Setup

Fixture

Fixture

3. JUnit

JUnit ("Patterns Generate Architectures", Kent Beck Ralph Johnson, ECOOP 94) 가 JUnit

3.1 - TestCase

TestCase

-
-
-

가

()

Command (Gamma, E., et al. Design Patterns : Elements of Reusable Object-Oriented Software, Addison-Wesley, Reading, MA, 1995)

"execute"

TestCase

```
public abstract class TestCase implements Test
{
    ...
}
```

Test "public abstract" TestCase class 가 TestCase

```
public abstract class TestCase implements Test
{
    private final String fName;

    public TestCase (String name) {
        fName = name;
    }
}
```

```

public abstract void run ();
...
}

```

JUnit
가

가

가

(Gamma, E., Applying Design Patterns in Java, in Java Gems, SIGS Reference Library, 1997)

1 TestCase



3.2 Blanks to fill in - run()

```

"
TestCase abstract
가
가
"
"
fixture
가 subclassing TestCase
super class
가
test fixture
fixture
가 fixture
가
가
"
가
가 fixture(setup and teardown)
fixture 가 가
"

```

```

public void run()

```


Collecting Parameter

가

TestResult

```
public class TestResult extends Object
{
    protected int fRunTests;
    public TestResult ()
    {
        fRunTests = 0;
    }
}
```

TestResult

TestCase.run()

가

TestResult

가

```
public void run (TestResult result) {
    result.startTest (this);
    ();
    runTest ();
    tearDown ();
}
```

TestResult

```
public synchronized void startTest (Test test)
{
    fRunTests ++;
}
```

가

TestResult가

TestResult

startTest

synchronized

TestCase

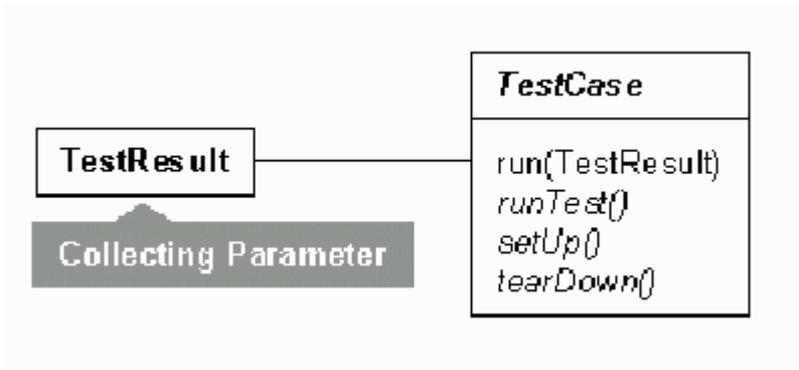
TestResult

가 run()

```
public TestResult run () {
    TestResult = createResult ();
    ();
    ;
}
```

```
protected TestResult createResult () {
    return new TestResult ();
}
```

3



가

가

가

JUnit

가

assertions

ArrayIndexOutOfBoundsException

AssertionFailedError

가 catch (1)

(2)

```

public void run(TestResult result)
{
    result.startTest(this);
    setUp();
    try{
        runTest();
    }
    catch(AssertionFailedError e){// 1
        result.addFailure(this, e);
    }
    catch(Throwable e){// 2
        result.addError(this, e);
    }
    finally{
        tearDown();
    }
}
  
```

AssertionFailedError TestCase assert
 JUnit 가 assert
 가

```

protected void assertTrue (boolean condition)
{
    if(!condition)
        throw new AssertionFailedError();
}
  
```

```
}
```

AssertionFailedError (TestCase)가
 TestCase.run() AssertionFailedError

```
public class AssertionFailedError extends Error
{
    public AssertionFailedError () {}
}
```

TestResult

```
public synchronized void addError (Test test, Throwable t)
{
    fErrors.addElement (new TestFailure (test, t));
}
```

```
public synchronized void addFailure (Test test, Throwable t)
{
    fFailures.addElement (new TestFailure (test, t));
}
```

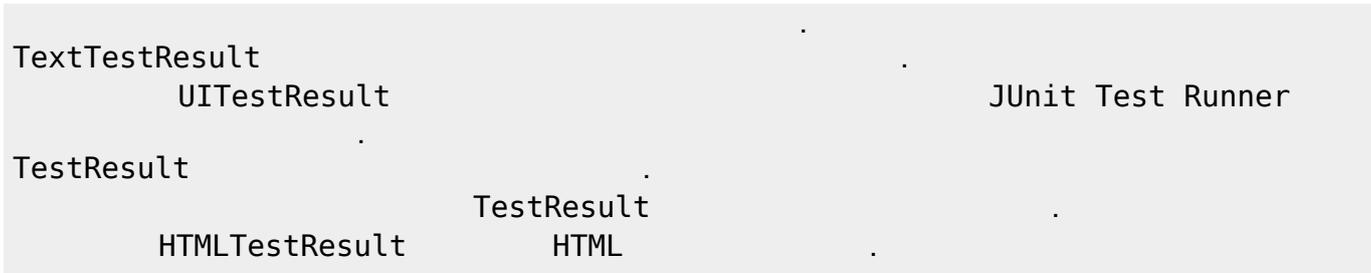
TestFailure
 helper class

```
public class TestFailure extends Object
{
    protected Test fFailedTest;
    protected Throwable fThrownException;
}
```

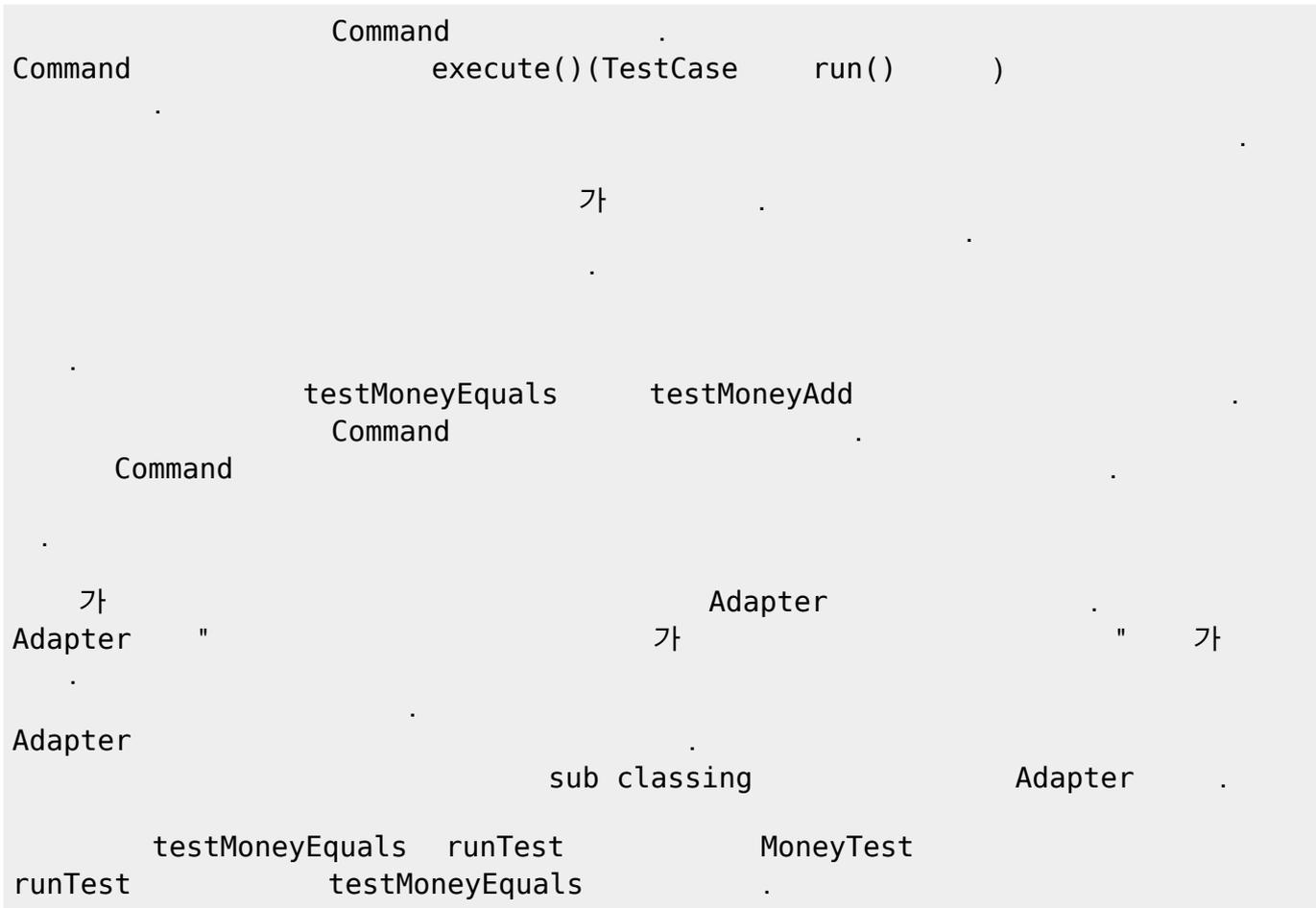
collecting parameter collecting parameter
 " " TestResult 가
 throw MoneyTest TestResult 가

```
public void testMoneyEquals()
{
    assertTrue(!f12CHF.equals(null));
    assertEquals(f12CHF, f12CHF);
    assertEquals(f12CHF, new Money(12, "CHF"));
    assertTrue(!f12CHF.equals(f14CHF));
}
```

JUnit TestResult

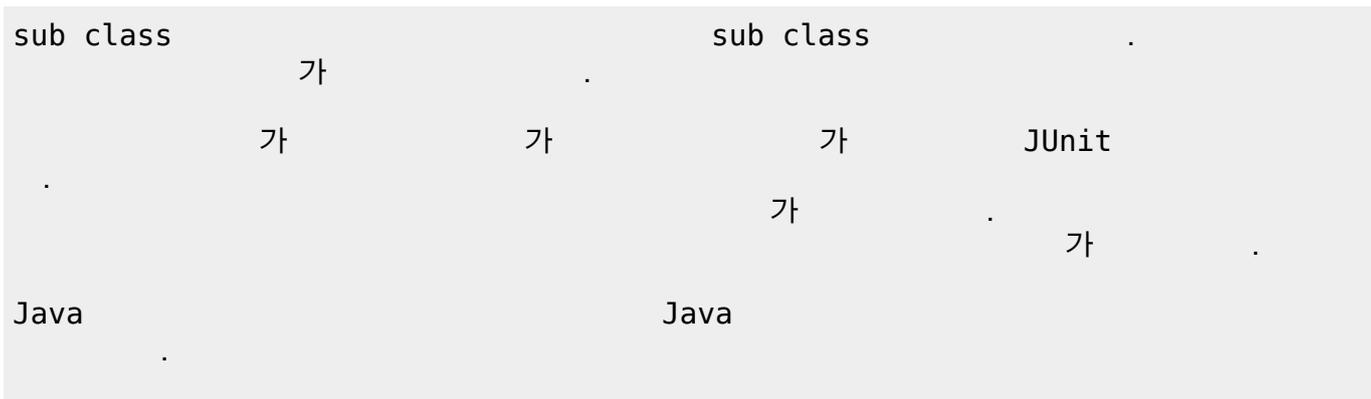


3.4 No stupid subclasses - TestCase again



```

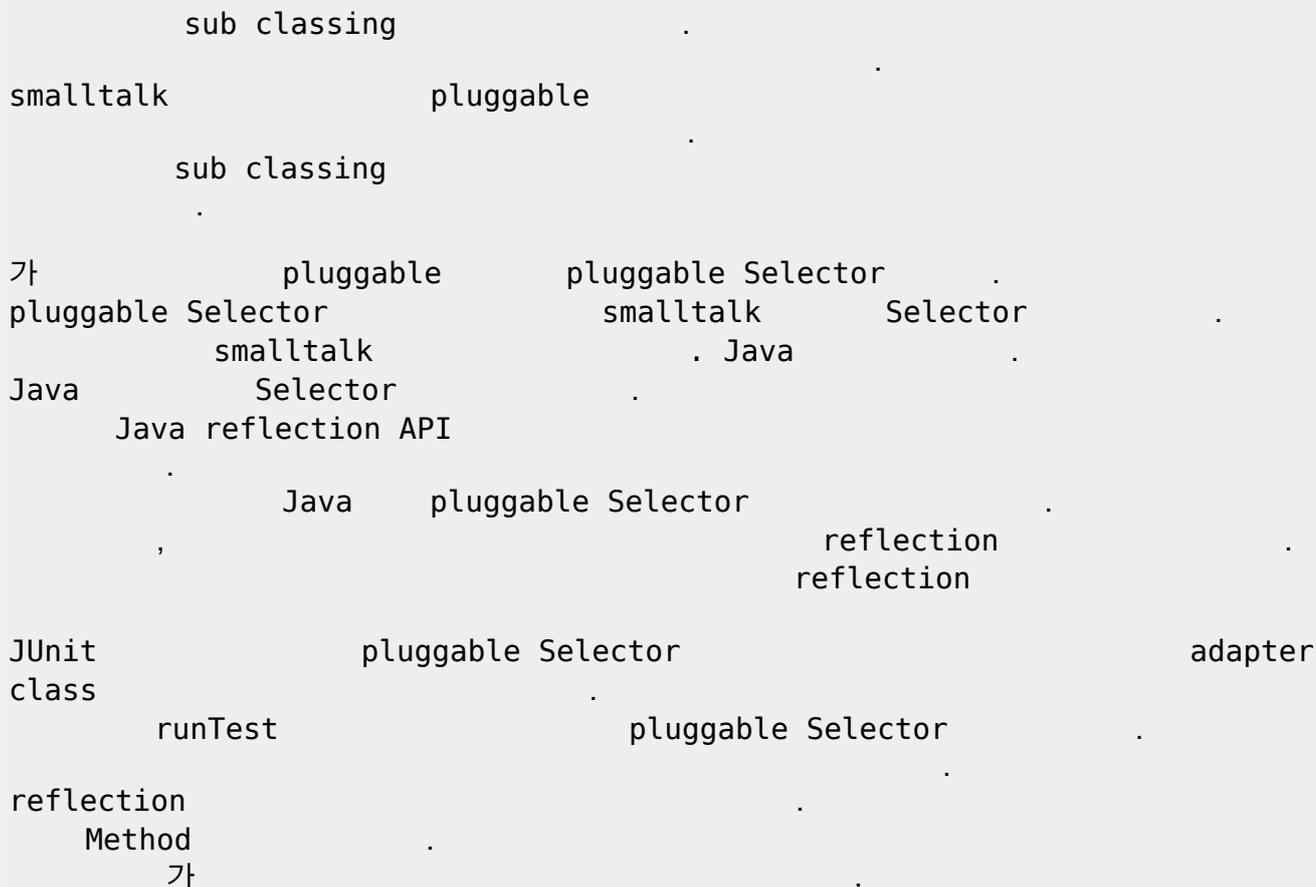
public class TestMoneyEquals extends MoneyTest
{
    public TestMoneyEquals() {super("testMoneyEquals"); }
    protected void runTest() {testMoneyEquals(); }
}
    
```



```

TestCase test = new MoneyTest ( "testMoneyEquals" )
{
    protected void runTest () {testMoneyEquals (); }
};

```



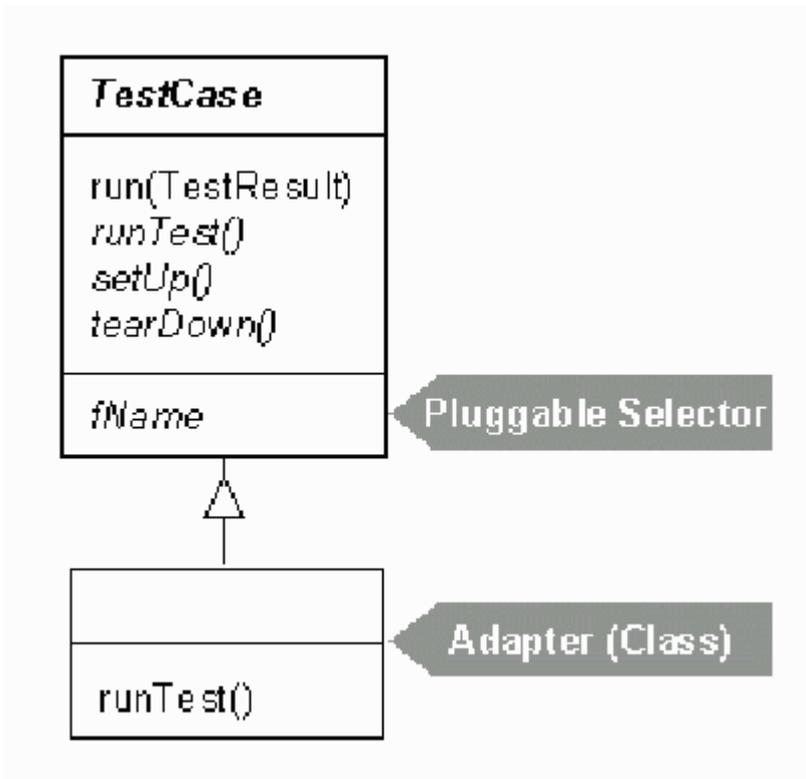
```

protected void runTest() throws Throwable
{
    Method runMethod = null;
    try
    {
        runMethod = getClass().getMethod(fName, new Class [0]);
    }
    catch (NoSuchMethodException e)
    {
        assertTrue ("Method \" " + fName + "\" ", false);
    }
    try
    {
        runMethod.invoke (this, new Class [0]);
    }
    // InvocationTargetException IllegalAccessException
}

```

JDK 1.1 reflection API public

public
 NoSuchMethodException
 Adapter pluggable selector가 가



3.5 Don't care about one or many - TestSuite

JUnit 가 가 Composite 가
 " Composite 가 Composite ."
 suites of suites of suites of tests
 Composite 가
 Component :
 Composite :
 Leaf : Component composition
 Java Composite 가
 JUnit 가

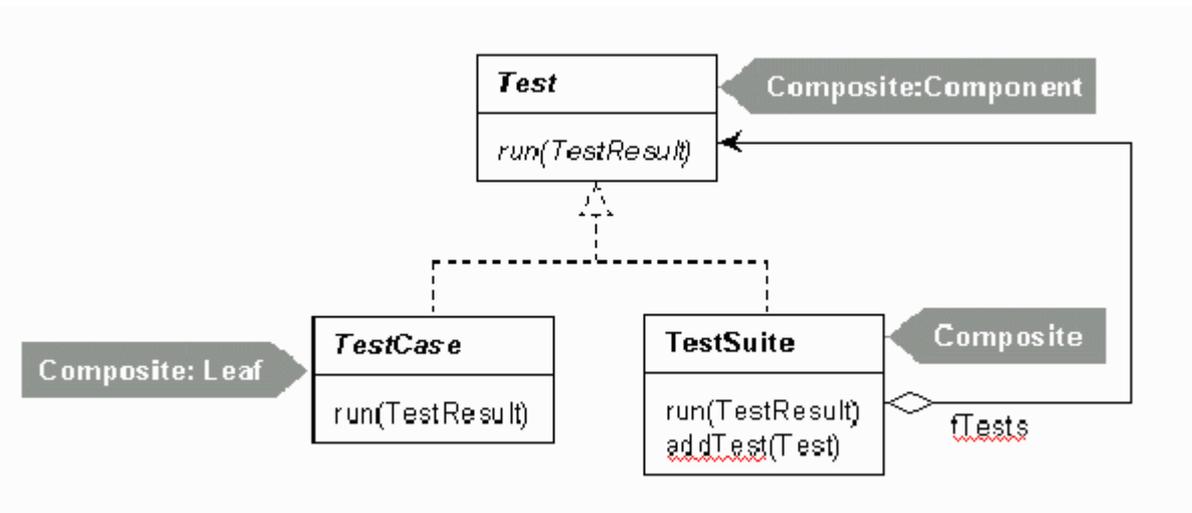
```
public interface Test
{
    public abstract void run(TestResult result);
}
```

TestCase Composite Leaf
 Composite 가
 TestSuite
 TestSuite Vector

```
public class TestSuite implements Test {
    private Vector fTests = new Vector ();
}
```

run()

```
public void run(TestResult result)
{
    for (Enumeration e = fTests.elements (); e.hasMoreElements ();) {
        Test test = (Test)e.nextElement ();
        test.run (result);
    }
}
```



suite 가 addTest

```
public void addTest (Test test)
{
    fTests.addElement (test);
}
```

가
 TestCase TestSuite Test test suites
 TestSuite

TestSuite
TestSuite

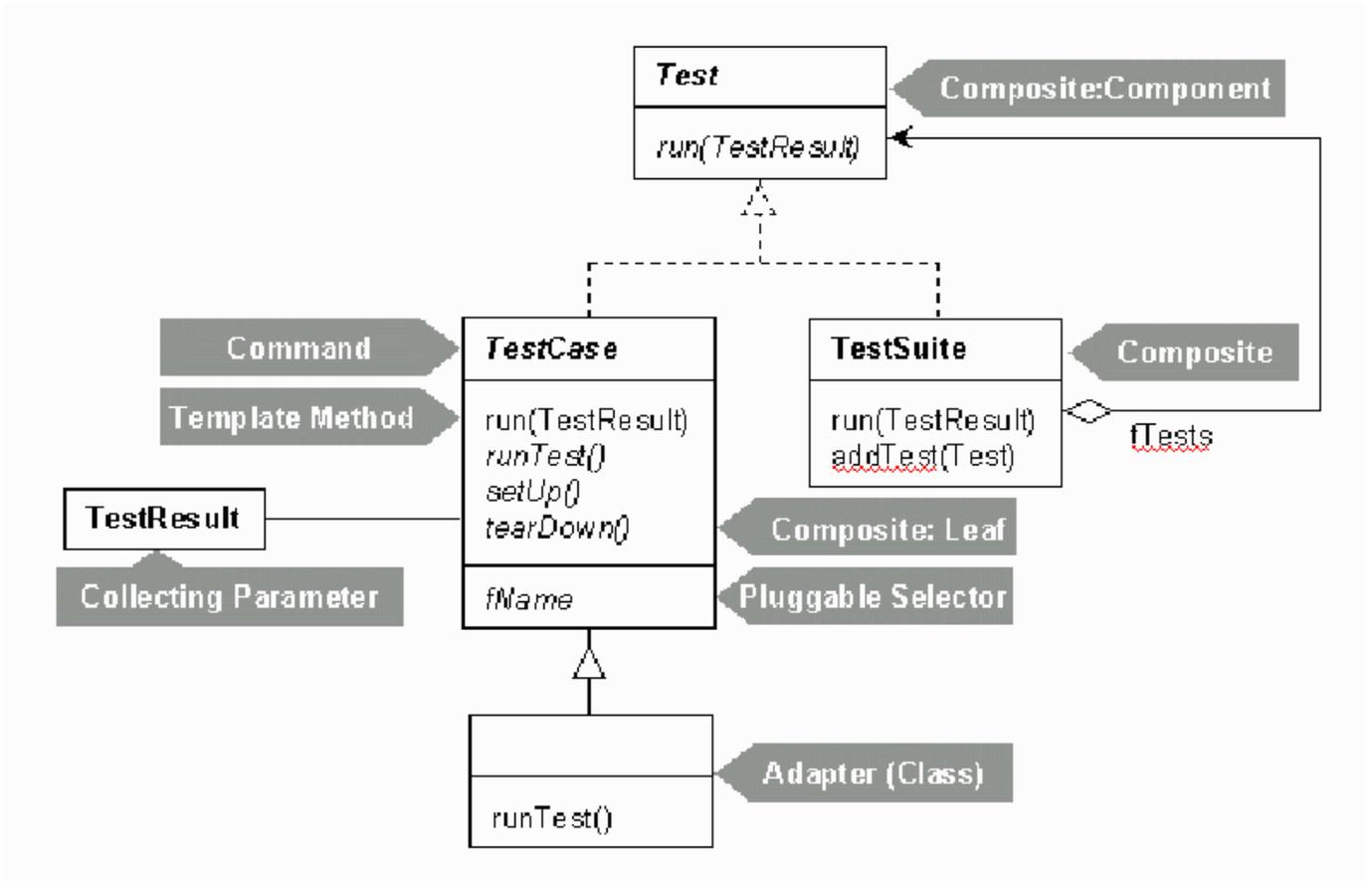
```
public static Test suite()  
{  
    TestSuite suite= new TestSuite ();  
    suite.addTest(new MoneyTest ( "testMoneyEquals"));  
    suite.addTest(new MoneyTest ( "testSimpleAdd"));  
}
```

JUnit
가
static suite() 가
TestSuite 가
"test"
reflection
가

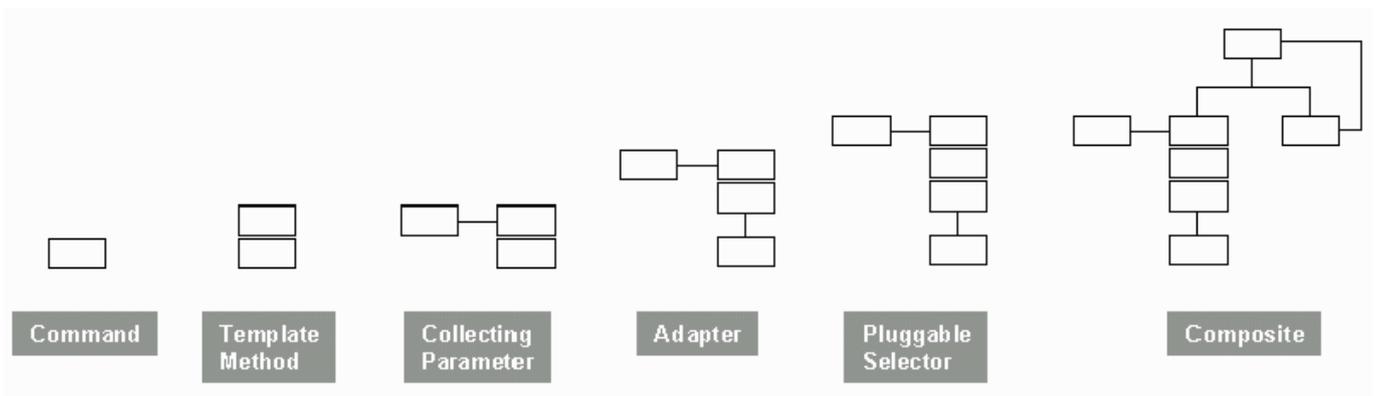
```
public static Test suite ()  
{  
    return new TestSuite (MoneyTest.class);  
}
```

3.6

JUnit A Cook's Tour
JUnit



TestCase가 4가
 " " :
 JUnit
 Command TestCase Template Method run
 (가 6 .)



Composite
 Composite가 " "

John Vlissides, Ralph Johnson, Nick Edgar .

: <http://junit.sourceforge.net/doc/cookstour/cookstour.htm>

junit

From:

<https://125.132.25.164/dokuwiki/> -
2023.12

Permanent link:

https://125.132.25.164/dokuwiki/doku.php?id=wiki:java:junit:junit_a_cook_s_tour

Last update: **2023/01/13 18:44**

