

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 가 ?
, PM !!
가 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
(Gamma, E., et al. Design Patterns : Elements of Reusable Object-Oriented Software, Addison-Wesley, Reading, MA, 1995)
" " " ..."
TestCase

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

TestCase TestCase 가

```
public abstract class TestCase implements Test  
{  
    private final String fName;  
    public TestCase (String name) {  
        fName = name;  
    }  
}
```

```

public abstract void run ();
...
}

```

JUnit
가

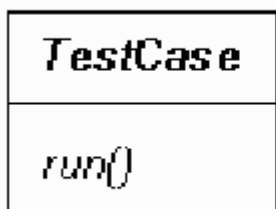
가

가

(Gamma, E., Applying

Design Patterns in Java, in Java Gems, SIGS Reference Library, 1997).

1 TestCase



3.2 in-run ()

TestCase

가

TestCase

가

가

가

가

가

"

가

가

"

가

가

```

public void run()
{
    setUp ();
    runTest ();
    tearDown ();
}

```

```
}
```

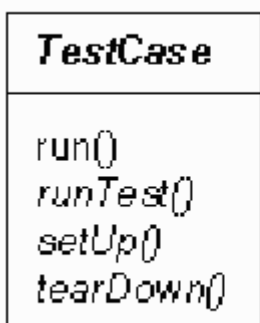
```
protected void runTest ()  
{  
    ...  
}
```

```
protected void setUp ()  
{  
    ...  
}
```

```
protected void tearDown ()  
{  
    ...  
}
```

```
setUp    tearDown
```

2



Template Method

3.3

TestCase가

가

?

가

가

가

TestCase

Smalltalk

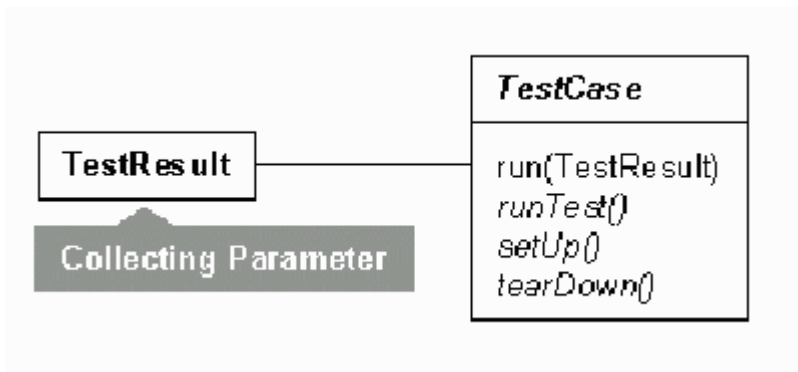
(Beck, K. Smalltalk

, Prentice Hall, 1996

) 가

가

TestResult



가

가

가

JUnit
가

ArrayIndexOutOfBoundsException
AssertionFailedError

가 catch (1)

(2)

```

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

AssertionFailedError TestCase assert
JUnit assert
가

```

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

```

AssertionFailedError (TestCase)가
TestCase.run () AssertionFailedError

```

```

AssertionFailedError {
public AssertionFailedError () {}
}

```

```
. TestResult
```

```

addError (, Throwable t) {
fErrors.addElement (new TestFailure (test, t));
}

```

```

addFailure (, Throwable t) {
fFailures.addElement (new TestFailure (test, t));
}

```

TestFailure

```

TestFailure Object {
protected Test fFailedTest
Throwable fThrownException;
}

```

```

" " TestResult 가
TestResult
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
TextTestResult
UITestResult JUnit Test Runner
TestResult
HTMLTestResult TestResult HTML

```

3.4 -TestCase

```
Command
execute () (TestCase run () )

가

testMoneyEquals testMoneyAdd

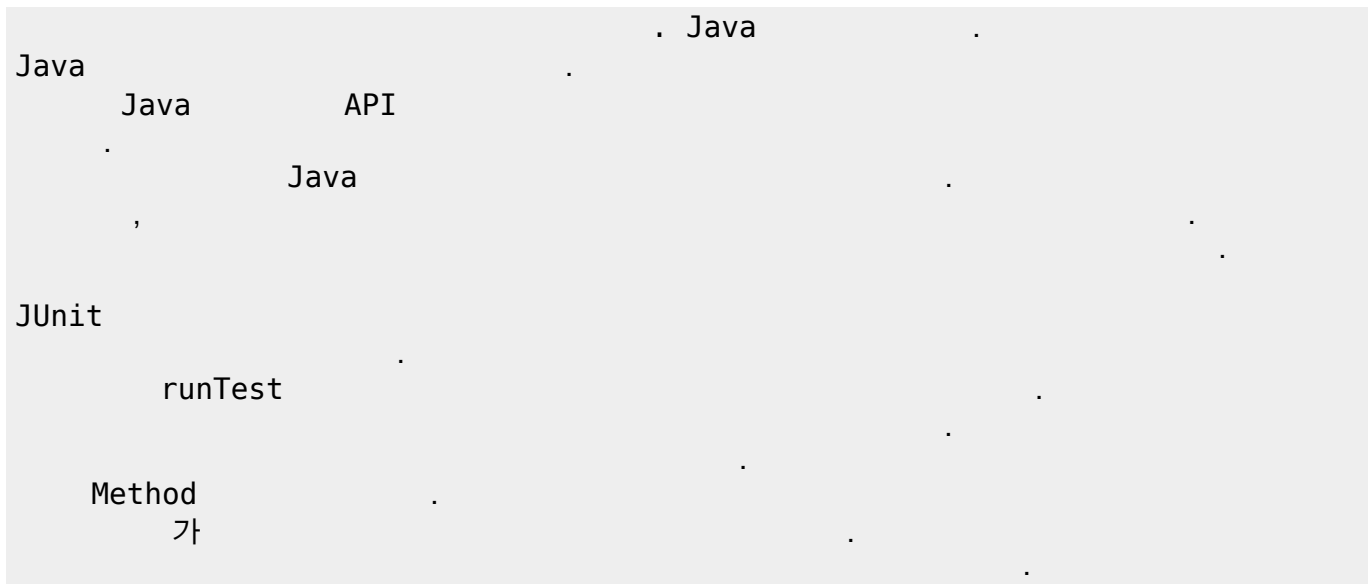
Command
가 " 가 " 가
runTest testMoneyEquals runTest MoneyTest
runTest testMoneyEquals
```

```
TestMoneyEquals MoneyTest {
public TestMoneyEquals () {super ( "testMoneyEquals"); }
runTest () {testMoneyEquals (); }
}
```

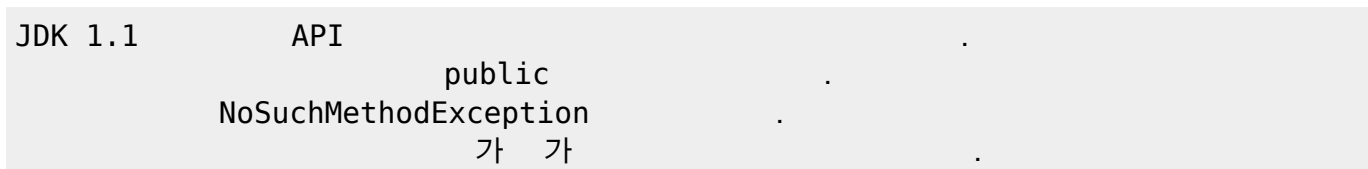
```
가
가 가 가 JUnit
가
Java Java 가
```

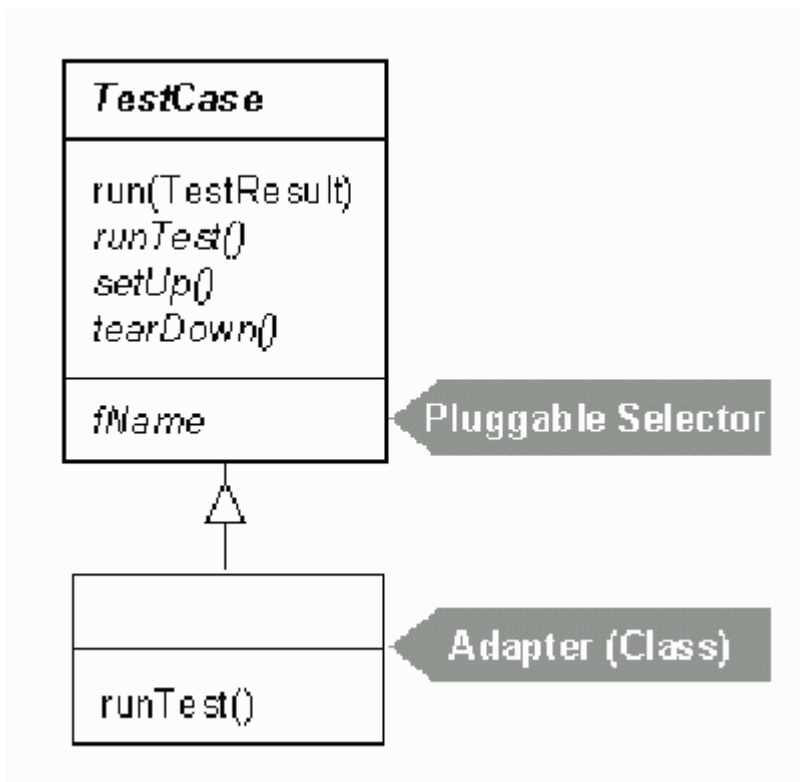
```
TestCase test = new MoneyTest ( "testMoneyEquals" ) {
protected void runTest () {testMoneyEquals (); }
};
```

```
가
Pluggable Selector Smalltalk
```

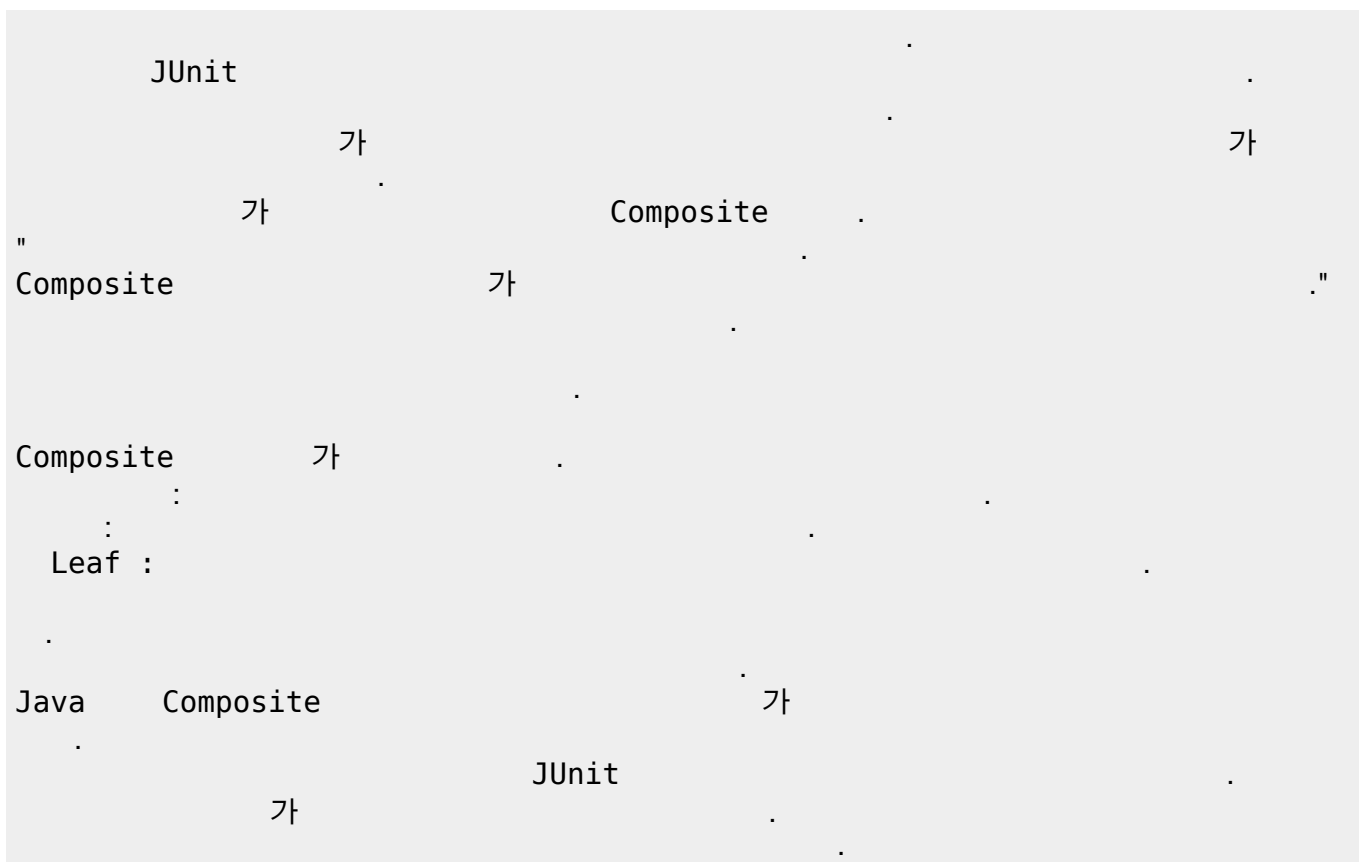


```
protected void runTest () throws Throwable {
    Method runMethod = null;
    try {
        runMethod = getClass (). getMethod ( fName, new Class [0]);
    } catch (NoSuchMethodException e) {
        assertTrue ( "Method \" " + fName + "\" " " ", false);
    }
    try {
        runMethod.invoke (this,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





3.5 -TestSuite



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

```

TestCase Composite Leaf
가
TestSuite
TestSuite

```

```

TestSuite Test {
private Vector fTests = new Vector ();
}

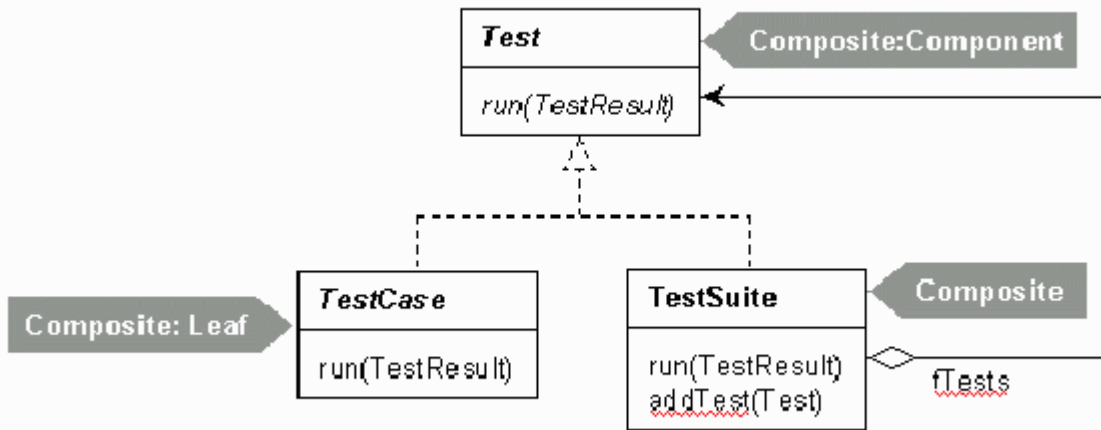
```

```
run ()
```

```

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

```



```
가 addTest
```

```

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

```

```

가
TestCase TestSuite Test
TestSuite
TestSuite
TestSuite

```

```

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

```

```

suite.addTest (new MoneyTest ( "testSimpleAdd"));
}
    
```

```

JUnit
    static suite ()
        TestSuite
            "test"
    
```

```

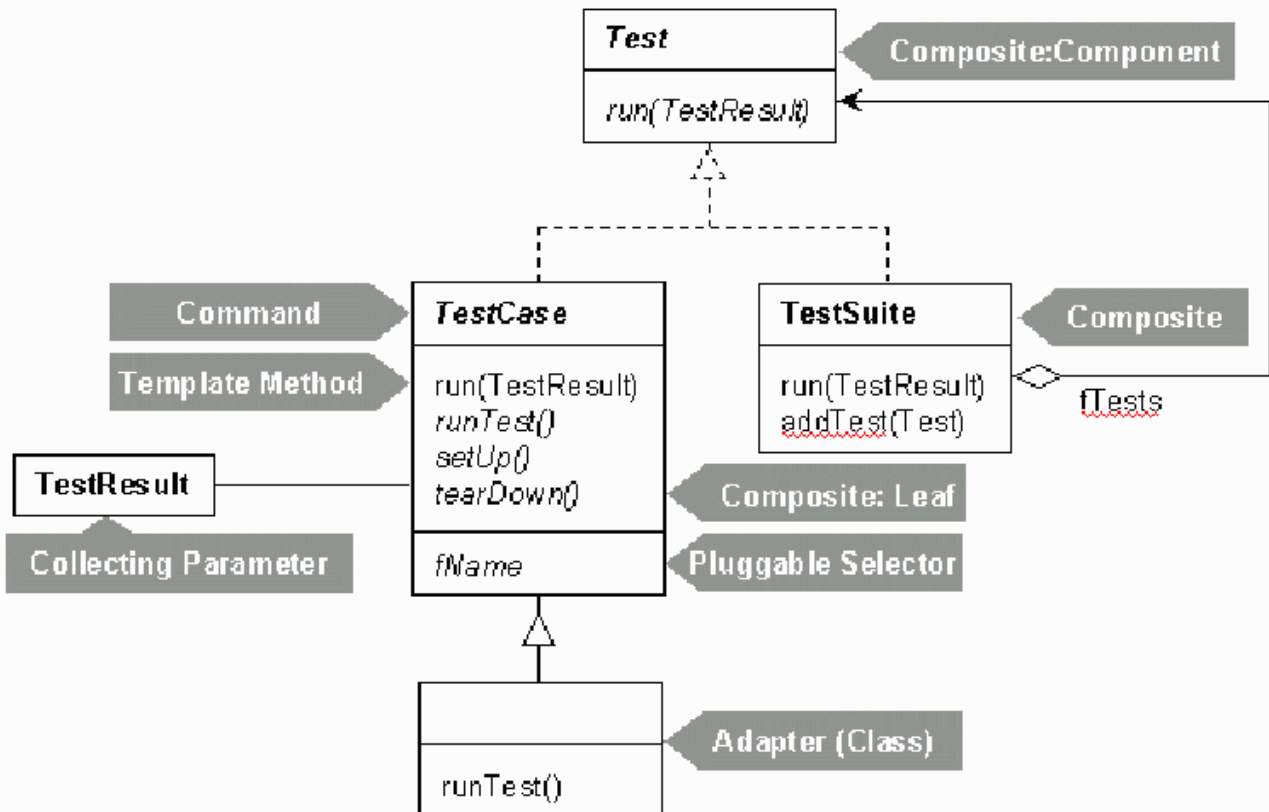
public static Test suite ()
{
    return new TestSuite (MoneyTest.class);
}
    
```

The original way is still useful when you want to run a subset of the test cases only.

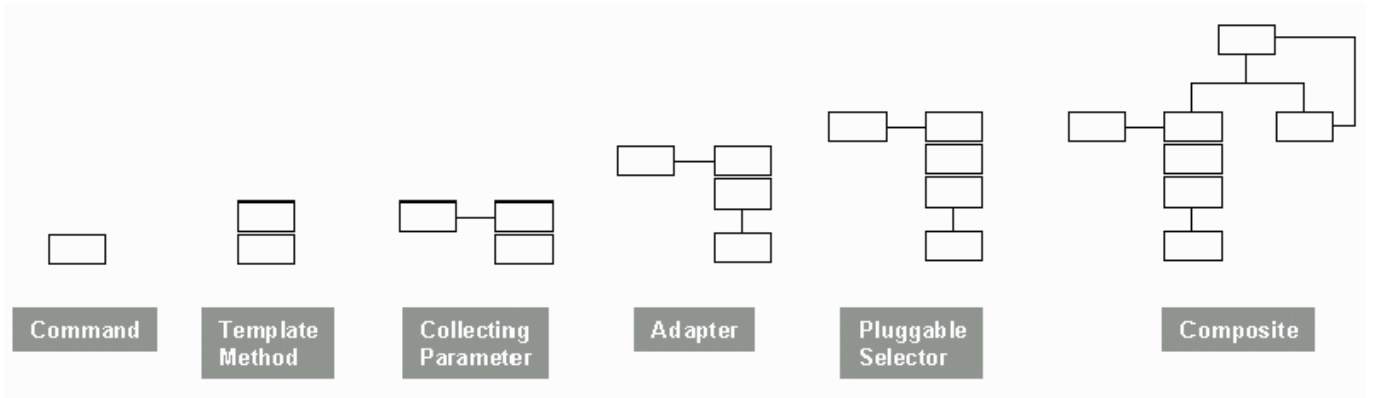
3.6

JUnit A Cook's Tour

JUnit



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



Composite
 Composite가 " "

4.

가 .

Patterns

Pattern density

JUnit TestCase " "가 .
 가 가 가 .
 가 " "

Eat your own dog food

TestTest 가 , .

