在eclipse里面开发junit真的很方便，只要在相应的类文件中点右键，"new"-->"junit test case"就可以生成一个测试类，里面大都代码都有了。

　　但要测试的方法的访问权限要设为"public"

　　下面简单介绍一下上边用到的静态类junit.framework.Assert。该类主要包含7个方法：

1．assertEquals()方法，用来查看对象中存的值是否是期待的值，与字符串比较中使用的equals()方法类似；

2．assertFalse()和assertTrue()方法，用来查看变量是是否为false或true，如果assertFalse()查看的变量的值是false则测试成功，如果是true则失败，assertTrue()与之相反。

3．assertSame()和assertNotSame()方法，用来比较两个对象的引用是否相等和不相等，类似于通过“==”和“!=”比较两个对象；

4．assertNull()和assertNotNull()方法，用来查看对象是否为空和不为空。

部分代码

　　　　public void testId() {

　　　　　　book.setId("001"); //设置id属性的值为001

　　　　　　//使用Assert查看id属性的值是否为001

　　　　　　Assert.assertEquals("001", book.getId());

　　　　　　System.out.println("id属性被测试！");

　　　　}

　　　　public void testName() {

　　　　　　book.setName("ASP"); //设置name属性的值为ASP

　　　　　　//使用Assert查看name属性的值是否为JSP，这是个必然出现错误的测试

　　　　　　Assert.assertEquals("JSP", book.getName());

　　　　　　System.out.println("name属性被测试！");

　　　　}

　　测试成功会显示绿色进度条

　　测试不成功会显示红色进度条

public static void main(String[] args) {

　　　　　　//使用TestSuite组装测试类

　　　　　　TestSuite ts = new TestSuite();

　　　　　　ts.addTestSuite(BookTest.class);

　　　　　　//textui，命令行方式

　　　　　　junit.textui.TestRunner.run(ts);

　　　　　　//swingui，Swing方式

　　　　　　//junit.swingui.TestRunner.run(ts.getClass());

　　　　　　//awtui，AWT方式

　　　　　　//junit.awtui.TestRunner.run(ts.getClass());

　　　　}

　　其中TestSuite是用来组织测试类的，通过其addTestSuite()方法将测试类添加到ts对象中，凡是添加到ts中的测试类都将在junit.textui.TestRunner.run()时被执行。此外，您也可以在定义ts对象时利用其构造函数将测试类添加给ts对象，例如：

TestSuite ts = new TestSuite(BookTest.class);

　　这并不会影响到后续使用addTestSuite()方法。再有，JUnit提供了三种运行方式，junit.textui.TestRunner是命令行方式的，junit.swingui.TestRunner和junit.awtui.TestRunner是可视化方式的，后两个这里已经注释掉了，如果您想查看效果，可以将注释去掉。