

Testautomatisierung ohne Assertions

Dr. Jeremias Rößler

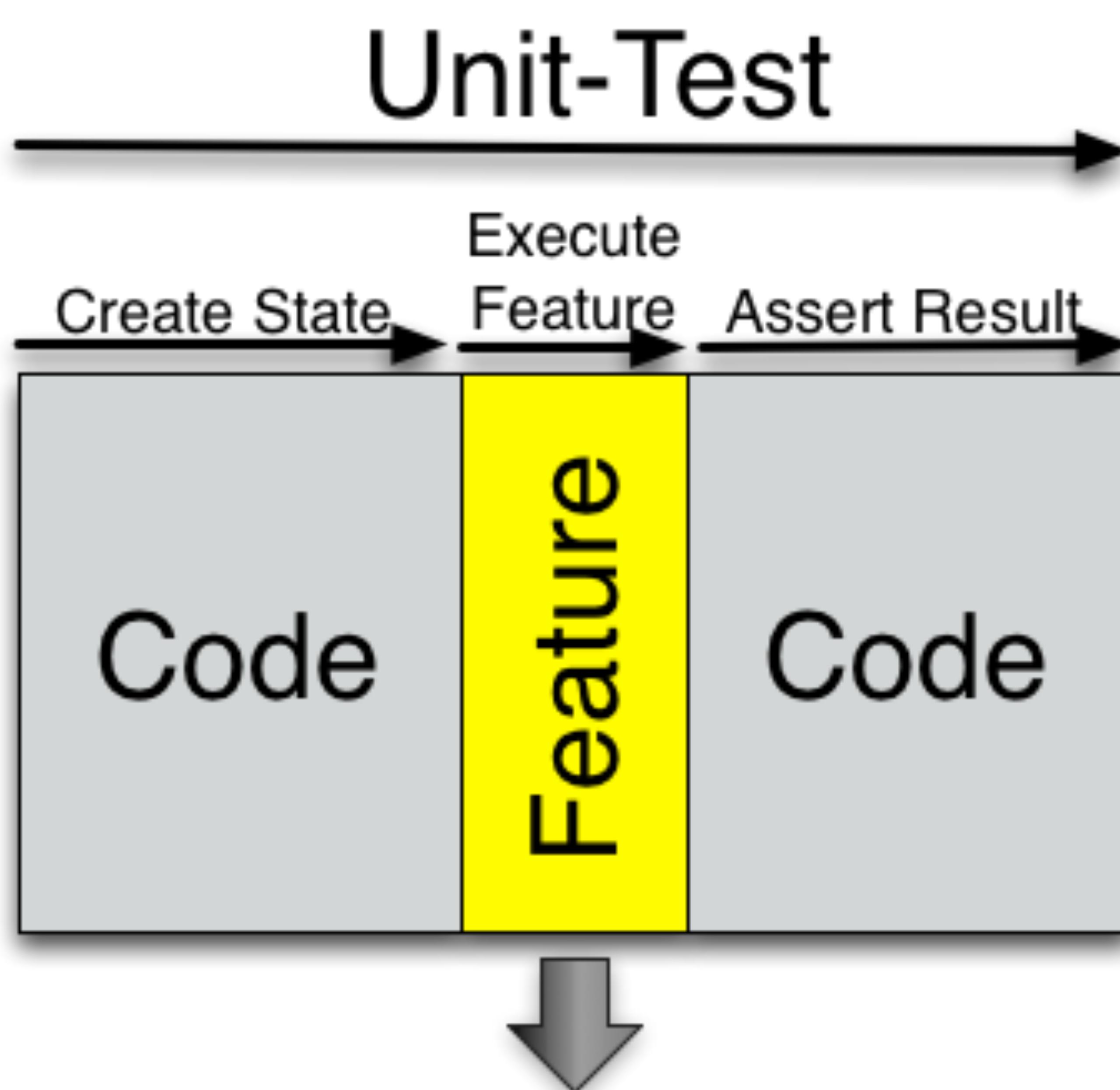
<https://retest.de>

What is an assertion?

```
@Test
public void extended_xml_should_be_removed_correctly() throws Exception {
    // Prepare state
    final String xml = "<?xml version=\"1.0\" encoding=\"UTF-8\" ?>\n" //
        + "<root>\n" //
        + "\t<parent>\n" //
        + "\t\t<remove>should be removed</remove>\n" //
        + "\t\t<keep>should not be removed</keep>\n" //
        + "\t</parent>\n" //
        + "</root>";
    final RemoveElementTransformer transformer = new RemoveElementTransformer( "parent", "remove" );

    // Execute feature under test
    final InputStream transform = transformer.transform( new ByteArrayInputStream( xml.getBytes() ) );

    // Assert result
    final String result = IOUtils.toString( transform, "UTF-8" );
    Assert.assertFalse( result.contains( "remove" ) );
    Assert.assertTrue( result.contains( "keep" ) );
}
```



4
DB-Mock

```
@Test
public void extended_xml_should_be_removed_correctly() throws Exception {
    // Prepare state
    final String xml = "<?xml version=\"1.0\" encoding=\"UTF-8\" ?>\n" //
        + "<root>\n" //
        + "\t<parent>\n" //
        + "\t\t<remove>should be removed</remove>\n" //
        + "\t\t<keep>should not be removed</keep>\n" //
        + "\t</parent>\n" //
        + "</root>";
    final RemoveElementTransformer transformer = new RemoveElementTransformer( "parent", "remove" );

    // Execute feature under test
    final InputStream transform = transformer.transform( new ByteArrayInputStream( xml.getBytes() ) );

    // Assert result
    final String result = IOUtils.toString( transform, "UTF-8" );
    Assert.assertFalse( result.contains( "remove" ) );
    Assert.assertTrue( result.contains( "keep" ) );
}
```

```
private WebDriver driver;

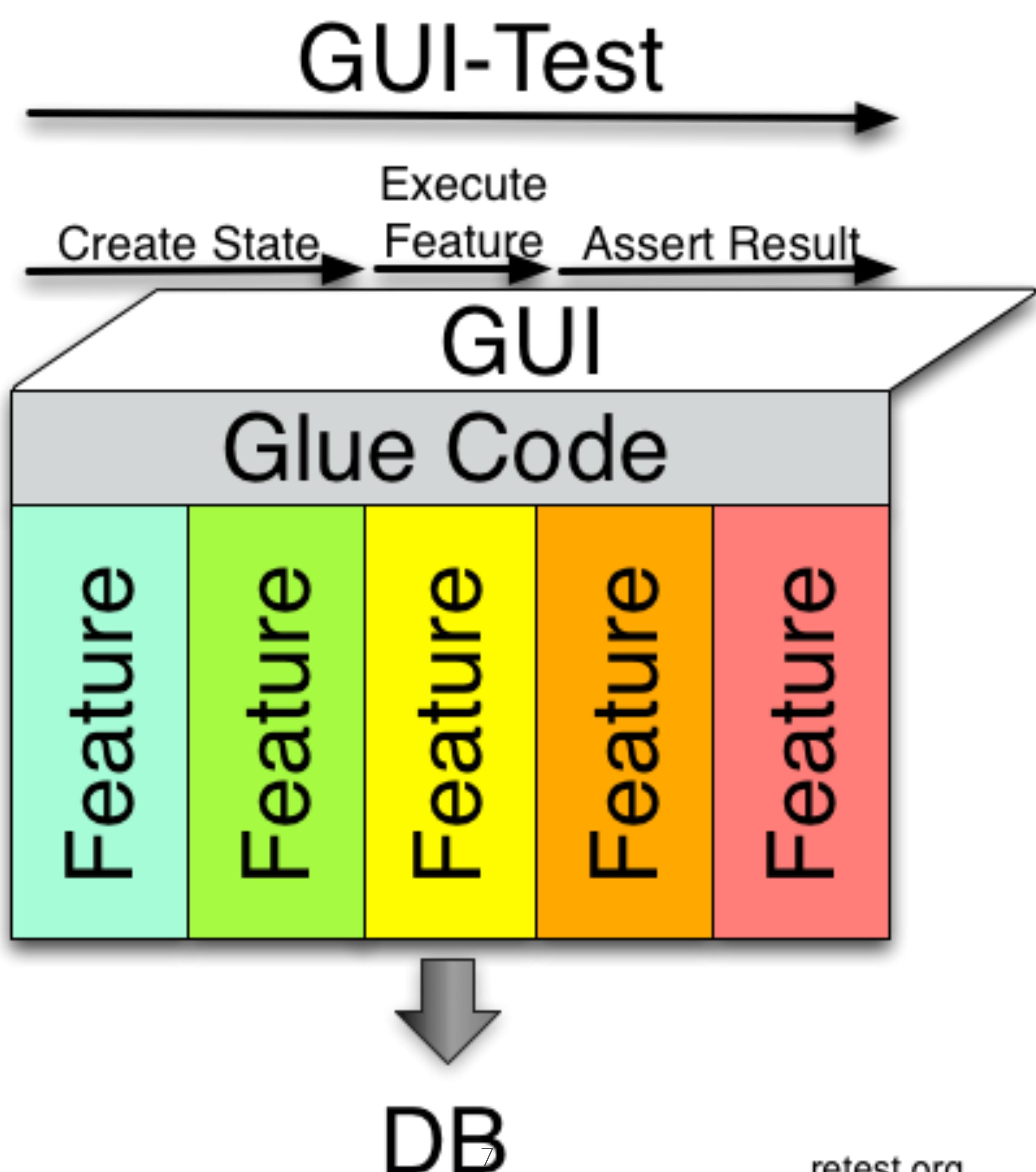
@Before
public void setup() {
    System.setProperty("webdriver.chrome.driver", "src/test/resources/chromedriver");
    driver = new ChromeDriver();
    driver.manage().timeouts().pageLoadTimeout(-1, TimeUnit.MINUTES);
}

@Test
public void check_add() throws Exception {
    // driver.get("http://localhost:8080/login");
    // login
    driver.findElement(By.name("username")).sendKeys("Max");
    driver.findElement(By.name("password")).sendKeys("retest");
    driver.findElement(By.name("login")).click();

    // test
    driver.findElement(By.id("1")).click();
    driver.findElement(By.id("plus")).click();
    driver.findElement(By.id("1")).click();
    driver.findElement(By.id("equals")).click();

    Assert.assertEquals(driver.findElement(By.name("result")).getAttribute("value"), "2.0");
}

@After
public void tearDown() {
    driver.quit();
}
```



```
private WebDriver driver;

@Before
public void setup() {
    System.setProperty("webdriver.chrome.driver", "src/test/resources/chromedriver");
    driver = new ChromeDriver();
    driver.manage().timeouts().pageLoadTimeout(-1, TimeUnit.MINUTES);
}

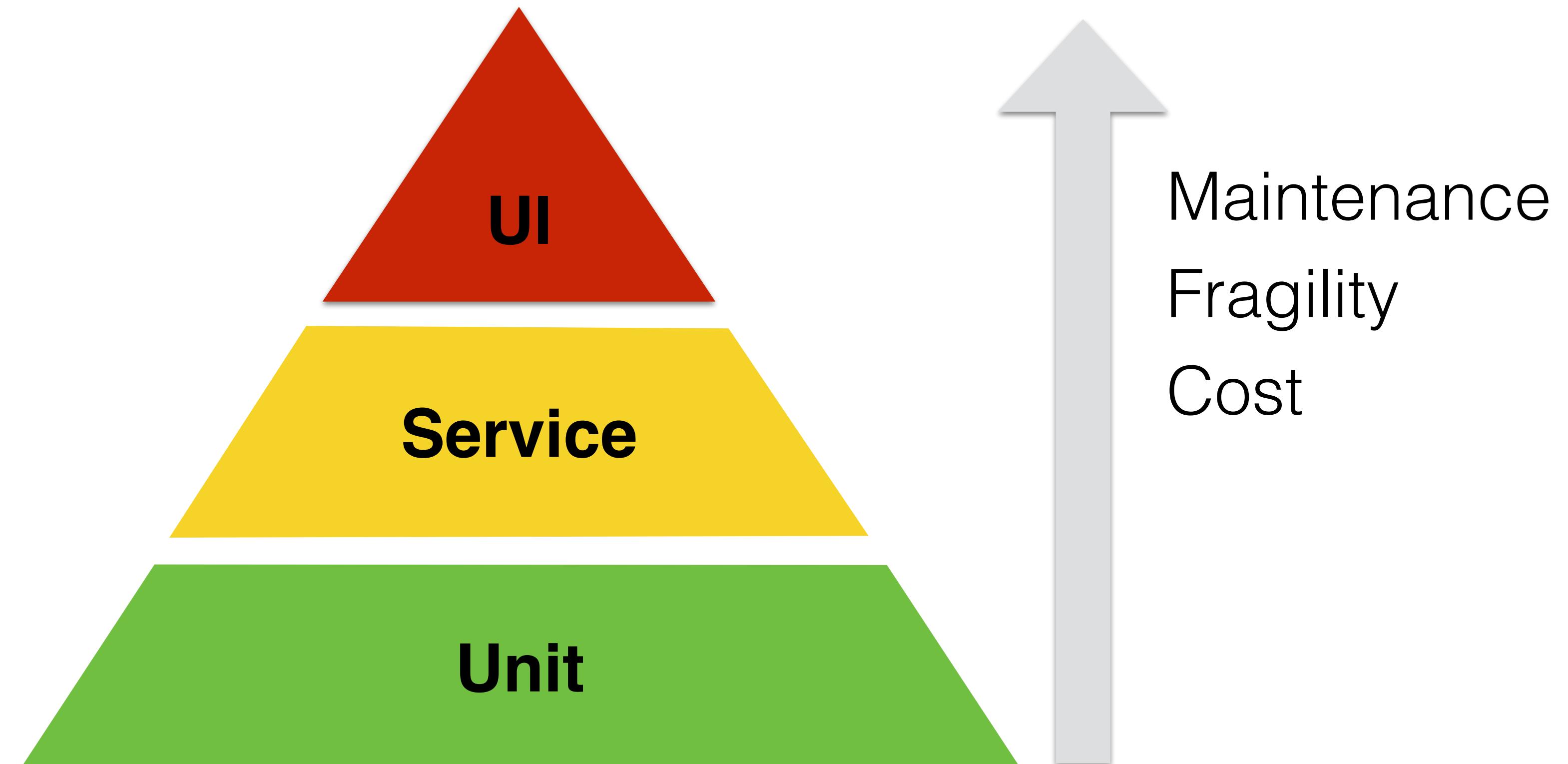
@Test
public void check_add() throws Exception {
    driver.get("http://localhost:8080/login");
    // login
    driver.findElement(By.name("username")).sendKeys("Max");
    driver.findElement(By.name("password")).sendKeys("retest");
    driver.findElement(By.name("login")).click();

    // test
    driver.findElement(By.id("1")).click();
    driver.findElement(By.id("plus")).click();
    driver.findElement(By.id("1")).click();
    driver.findElement(By.id("equals")).click();

    Assert.assertEquals(driver.findElement(By.name("result")).getAttribute("value"), "2.0");
}

@After
public void tearDown() {
    driver.quit();
}
```

Test Pyramid



[Show Tables](#) [Show Views](#) [Show Column](#) [Create View](#) [Drop Table](#) [Create Index](#)

Query time (in millisec) = 47

Number of rows retrieved from database = 22

[Reformat](#) [CSV](#) [Plot \(VOPlot\)](#)

haloID	subHaloID	lastProgenitorID	treeID	snapNum	redshift	firstProgenitorID	nextProgenitorID	descendantID	firstMolteForwards	lastMolteForwards
526000013	50000000001445	526000033	526000000	50	0.40769944	526000014	-1	526000012	526000013	-1
874000013	50000000001486	874000051	874000000	50	0.40769944	874000014	-1	874000012	874000013	-1
663000013	50000000001461	663000043	663000000	50	0.40769944	663000014	-1	663000012	663000013	-1
623000013	50000000001403	623000042	623000000	50	0.40769944	623000014	-1	623000012	623000013	-1
518000013	50000000001349	518000041	518000000	50	0.40769944	518000014	-1	518000012	518000013	-1
642000013	50000000001354	642000046	642000000	50	0.40769944	642000014	-1	642000012	642000013	-1
602000013	50000000001345	602000043	602000000	50	0.40769944	602000014	-1	602000012	602000013	-1
522000013	50000000001338	522000045	522000000	50	0.40769944	522000014	-1	522000012	522000013	-1
534000013	50000000001324	534000044	534000000	50	0.40769944	534000014	-1	534000012	534000013	-1
589000013	50000000001248	589000041	589000000	50	0.40769944	589000014	-1	589000012	589000013	-1
585000013	50000000001245	585000038	585000000	50	0.40769944	585000014	-1	585000012	585000013	-1
427000013	50000000001204	427000043	427000000	50	0.40769944	427000014	-1	427000012	427000013	-1
476000013	50000000001181	476000059	476000000	50	0.40769944	476000014	-1	476000012	476000013	-1
112000094	50000000001116	112000126	112000000	50	0.40769944	112000095	-1	112000093	112000094	-1
457000013	50000000001084	457000054	457000000	50	0.40769944	457000014	-1	457000012	457000013	-1
307000013	50000000000892	307000041	307000000	50	0.40769944	307000014	-1	307000012	307000013	307000055
375000013	50000000000981	375000054	375000000	50	0.40769944	375000014	-1	375000012	375000013	-1
18000444	50000000001001	18000498	180000000	50	0.40769944	18000445	-1	18000443	18000444	-1

```
Assert.assertEquals(driver.findElement(By.xpath("table['results']/tr[1]/td[1]")), "5260000D13");
Assert.assertEquals(driver.findElement(By.xpath("table['results']/tr[1]/td[2]")), "50000000001445");
Assert.assertEquals(driver.findElement(By.xpath("table['results']/tr[1]/td[3]")), "526000033");
Assert.assertEquals(driver.findElement(By.xpath("table['results']/tr[1]/td[4]")), "526000000");
Assert.assertEquals(driver.findElement(By.xpath("table['results']/tr[1]/td[5]")), "50");
Assert.assertEquals(driver.findElement(By.xpath("table['results']/tr[1]/td[6]")), "0.40789944");
Assert.assertEquals(driver.findElement(By.xpath("table['results']/tr[1]/td[7]")), "5260000014");
Assert.assertEquals(driver.findElement(By.xpath("table['results']/tr[1]/td[8]")), "-1");
Assert.assertEquals(driver.findElement(By.xpath("table['results']/tr[1]/td[9]")), "5260000012");
Assert.assertEquals(driver.findElement(By.xpath("table['results']/tr[1]/td[10]")), "5260000013");
Assert.assertEquals(driver.findElement(By.xpath("table['results']/tr[1]/td[11]")), "-1");
```

Binding test to test-data!

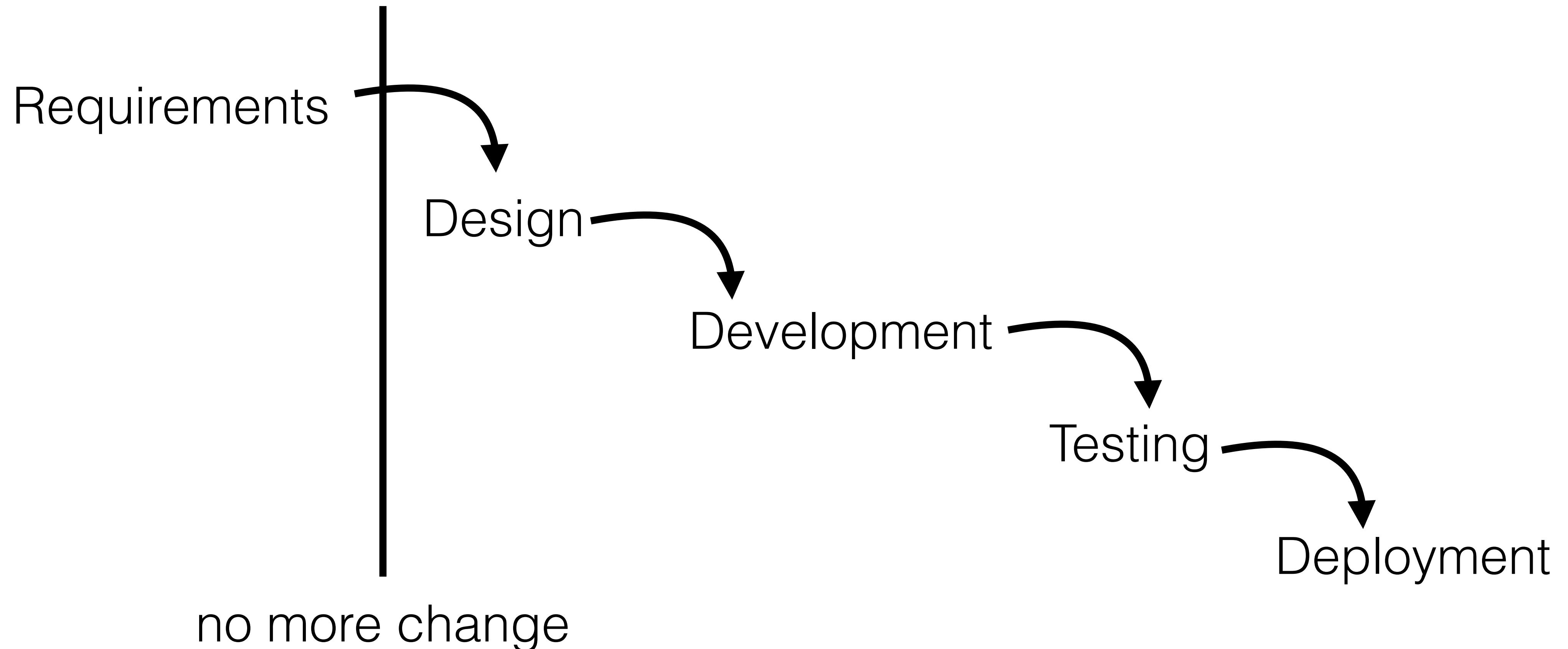
What does an assertion do?

Assert correct behavior

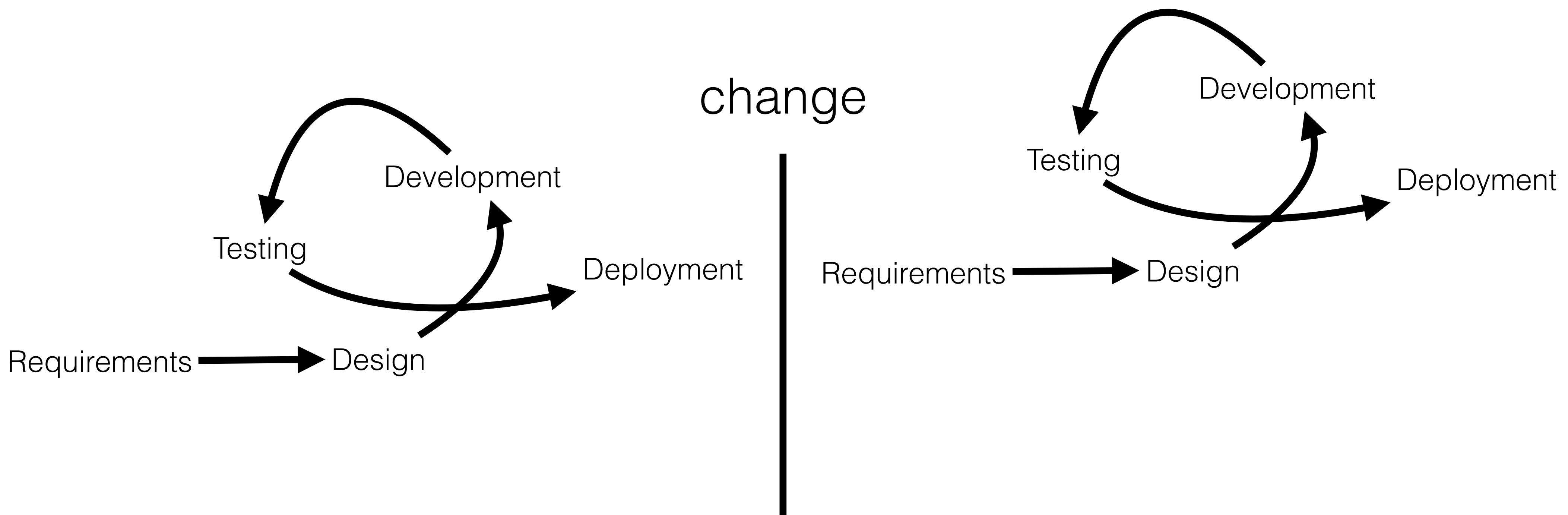
$$1 - 2 = ?$$

- -1 is mathematically correct
- 0 if I don't want to payback gift certificates to costumers
- 23 if I'm working with hours
- 59 if I'm working with minutes or seconds
- Something between 27 and 30 if I'm working with days
- 11 if I am working with months
- exception if I am working with degrees Calvin
- ...

Waterfall



Agile



“correct” changes

When is a bug a bug?



When it's not a feature!

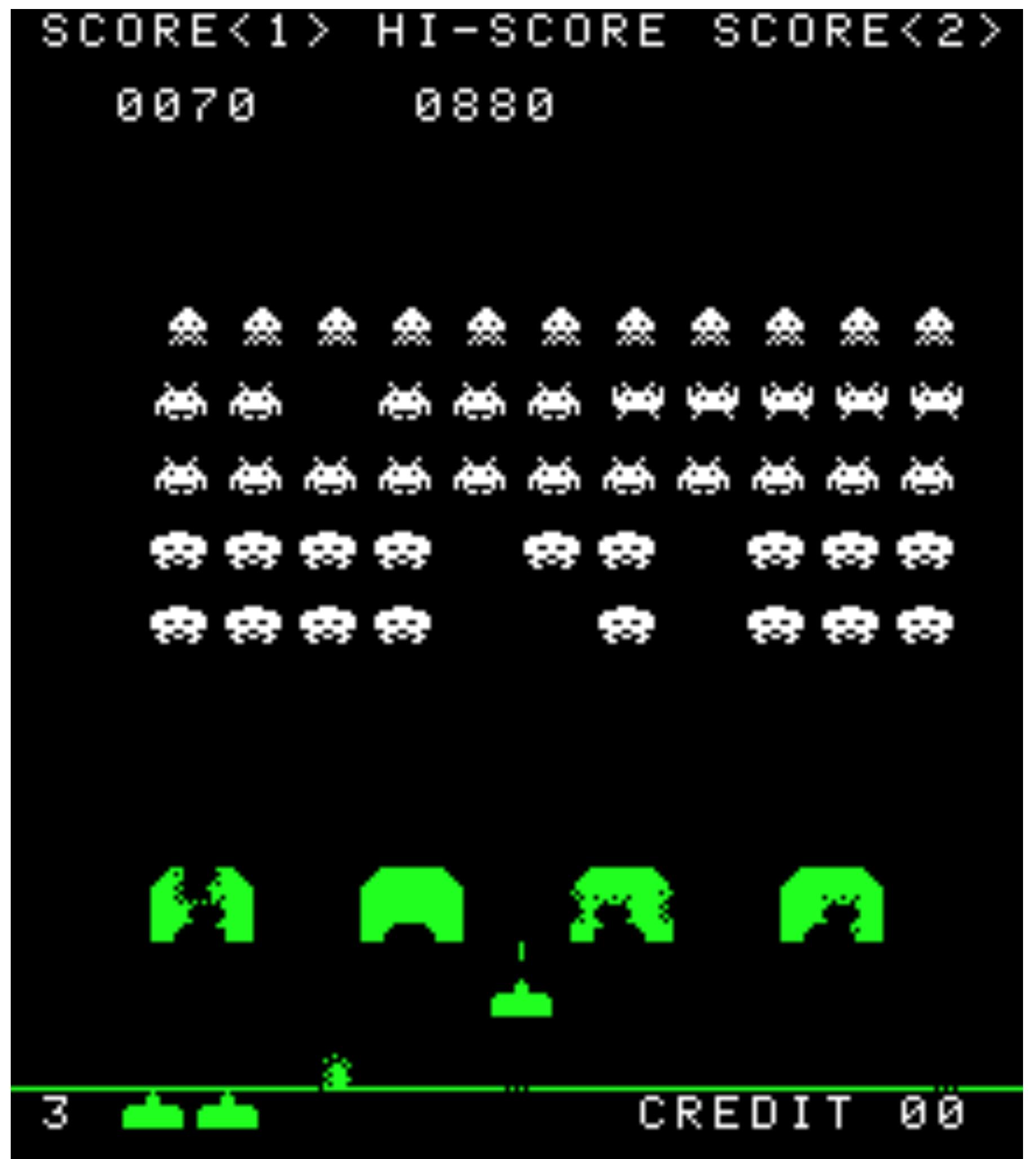
Is it a bug?

```
..  
..  
text.c  
other.files
```

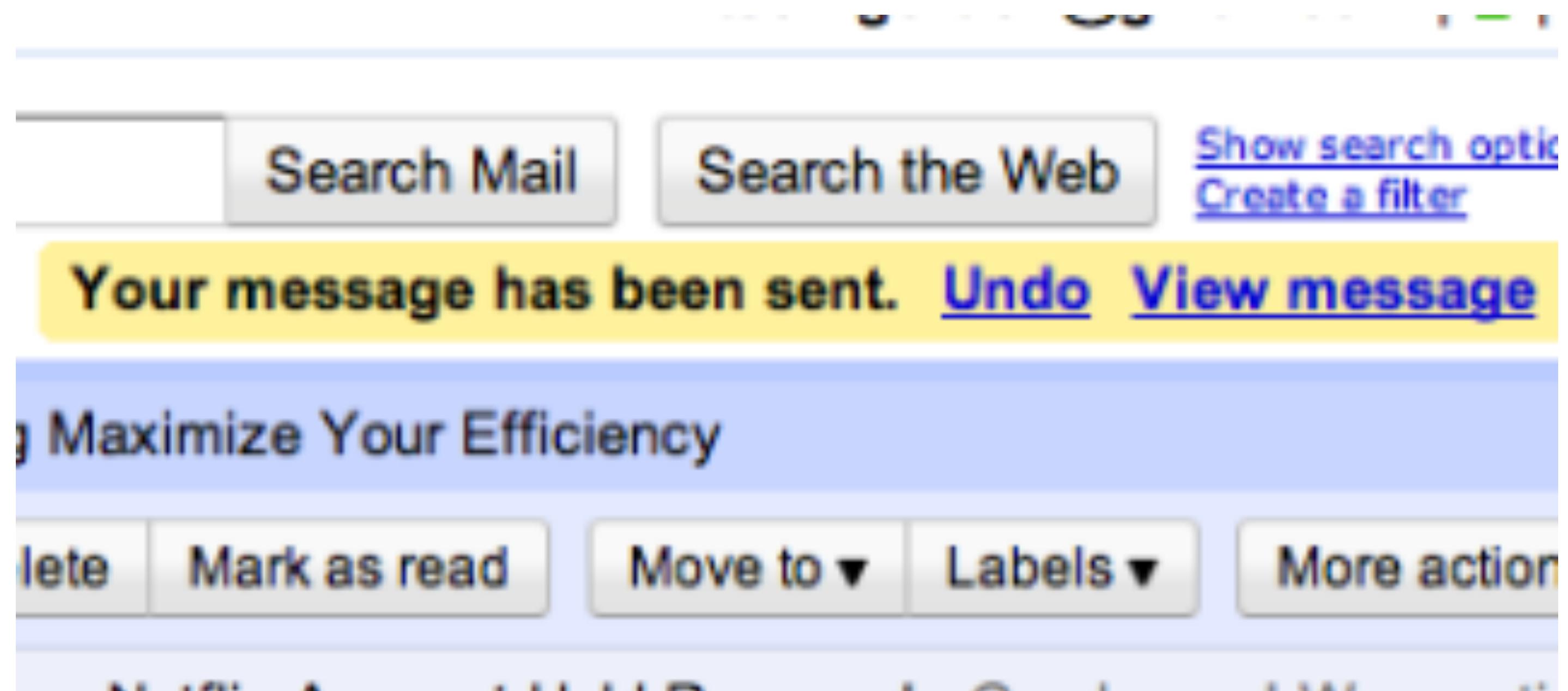
ls

```
for (. . .) {  
    if (name[0] == '.') continue;  
    . . .  
}
```

Is it a bug?



Is it a bug?



What is a bug?

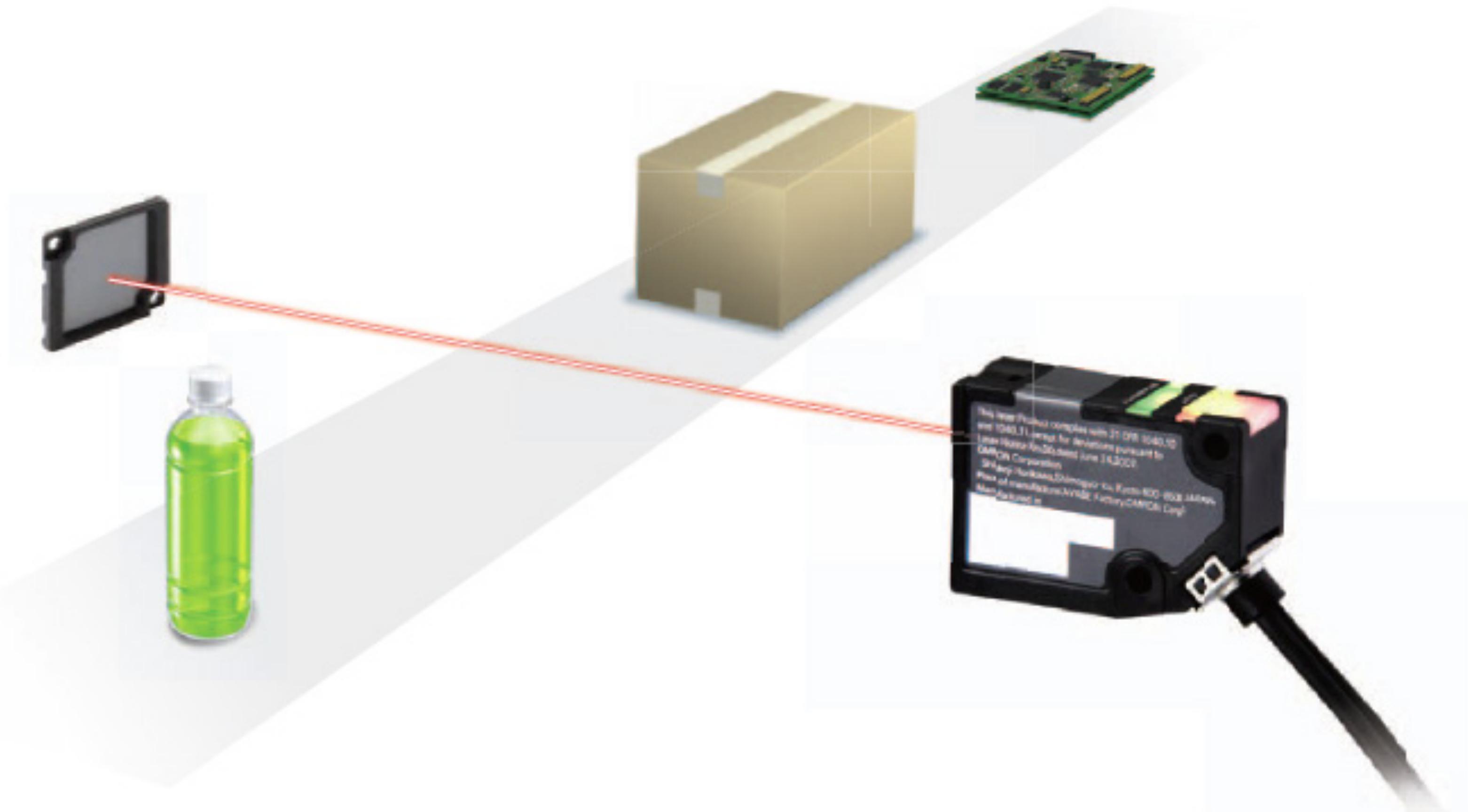
“

Without specification, there are no bugs
— only surprises.

Brian Kernighan

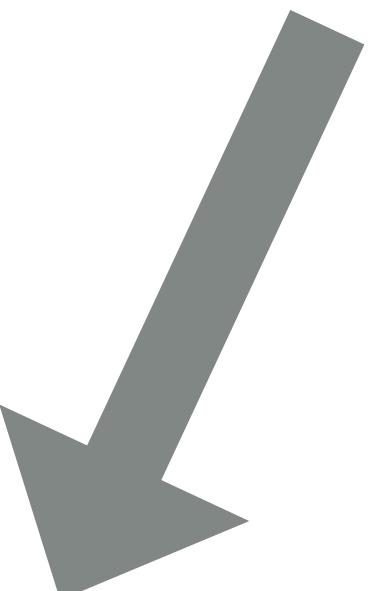
Detect change

Assert correct behavior



Assertions detect change

Why do we test?

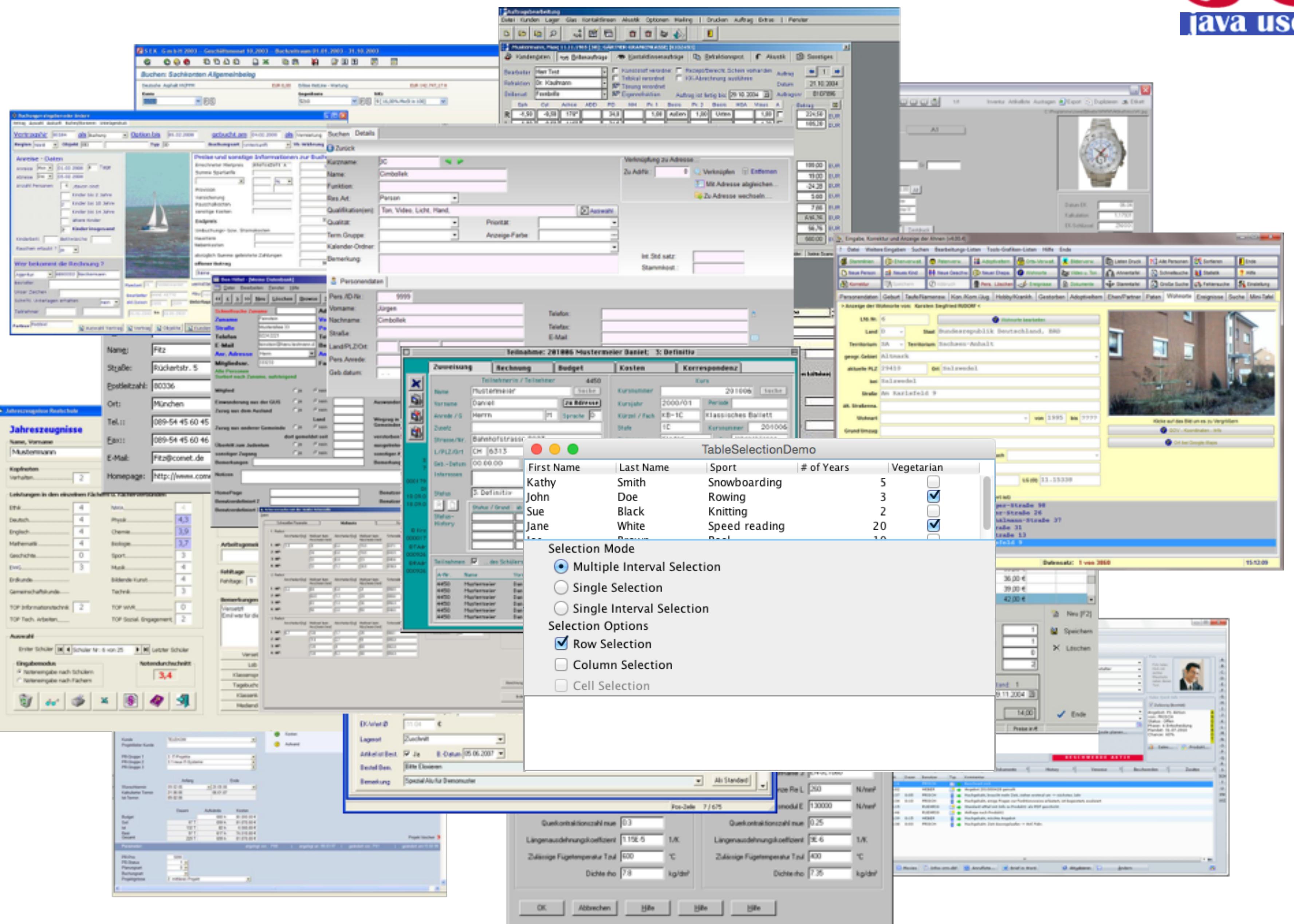


After
implementation?

$$1 + 1 = 3 \quad \text{X}$$
$$2 + 2 = 4 \quad \checkmark$$

The collage illustrates a variety of Java-based applications:

- Top Left:** A screenshot of a financial application showing a ledger entry for "Deutsche Asphalt HV/FFM" with a balance of EUR 142,747,17 H.
- Top Center:** An "Auftragbearbeitung" (Order Processing) window showing a customer record for "Mustermann, Max 11.11.1955 [30]; GÄRTNER-KRANKENKASSE [K102493]".
- Top Right:** A screenshot of a watch inventory application showing a list of watches and their details.
- Middle Left:** A "Buchungen eingeben oder ändern" (Enter or Change Entries) window for a contract (Vertrag Nr. 00184).
- Middle Center:** A "Personendaten" (Person Data) window showing contact information for "Jürgen Cimballek".
- Middle Right:** A "TableSelectionDemo" window displaying a table of student data with selection options.
- Bottom Left:** A "Jahreszeugnisse Realschule" (Report Card) window for "Mustermann" showing grades and subjects like Math, English, and Science.
- Bottom Center:** A screenshot of a project management application showing tasks and budgets.
- Bottom Right:** A "BESCHWERDE AKTIV" (Complaint Active) window showing a list of complaints and their status.



TableSelectionDemo

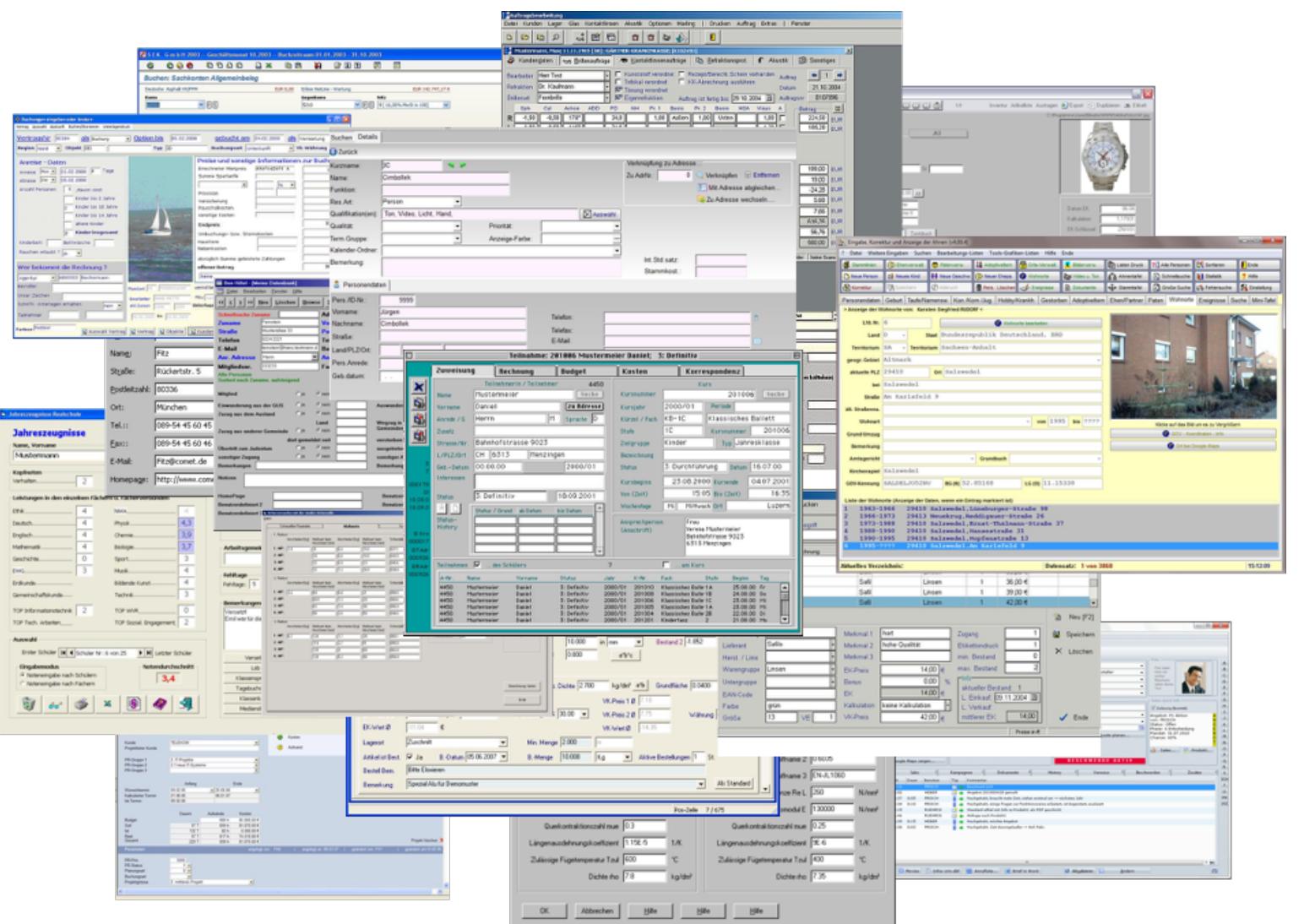
First Name	Last Name	Sport	# of Years	Vegetarian
Kathy	Smith	Snowboarding	5	<input type="checkbox"/>
John	Doe	Rowing	3	<input checked="" type="checkbox"/>
Sue	Black	Knitting	2	<input type="checkbox"/>
Jane	White	Speed reading	20	<input checked="" type="checkbox"/>
Total: 10				

Selection Mode

- Multiple Interval Selection
- Single Selection
- Single Interval Selection

Selection Options

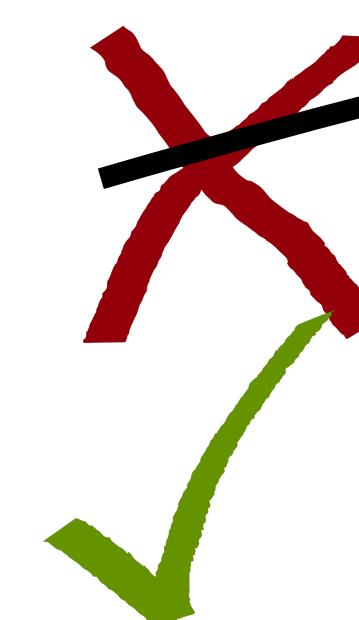
- Row Selection
- Column Selection
- Cell Selection



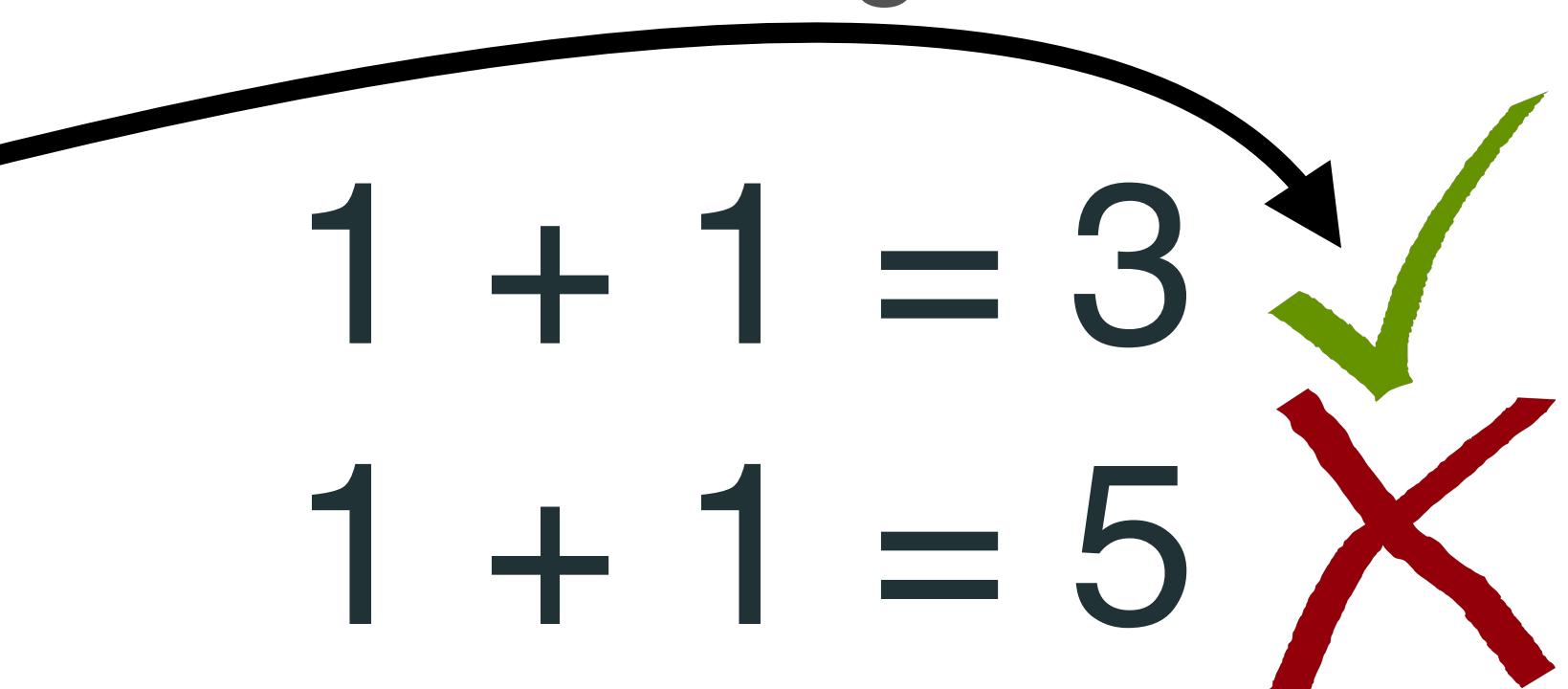
Why do we test?



After
implementation?

$$\begin{aligned}1 + 1 &= 3 \\2 + 2 &= 4\end{aligned}$$


After
change?


$$\begin{aligned}1 + 1 &= 3 \\1 + 1 &= 5\end{aligned}$$


automated
regression test

≠

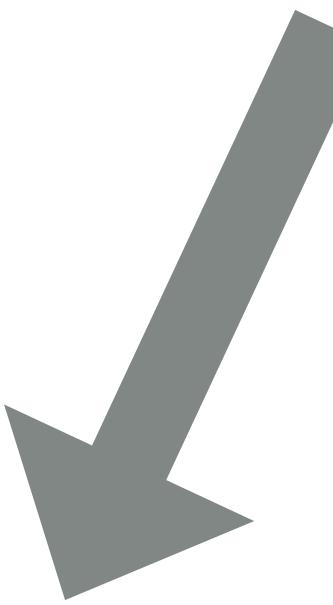
test

automated
regression test

=

version control

Why do we test?



After
implementation?

$$1 + 1 = 3$$

$$2 + 2 = 4$$

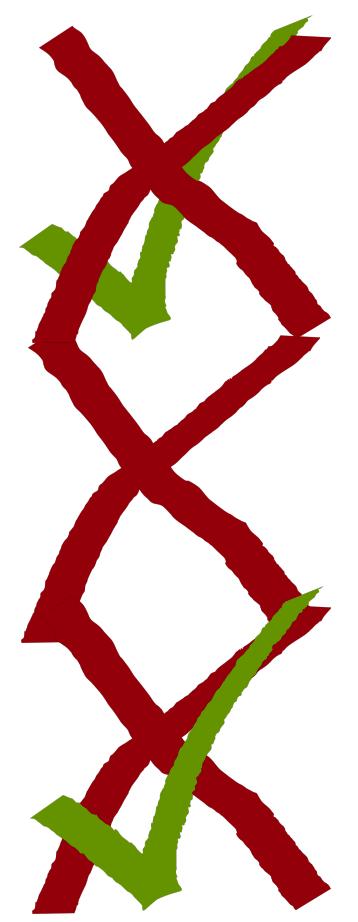


After
change?

$$1 + 1 = 3$$

$$1 + 1 = 5$$

$$1 + 1 = 2$$



The screenshot shows a Java application window titled "Viewing: apstring.cpp". The window has tabs for "File", "Difference", "Settings", and "Help". Below the tabs are buttons for "Previous File", "Next File", and "Files". The main area is divided into two panes: the left pane displays the contents of "apstring.cpp" and the right pane displays the contents of "apstring.h". Both files are color-coded with syntax highlighting. The "apstring.cpp" file contains code for string manipulation, including operators like + and substr. The "apstring.h" file contains declarations for these operators and defines a private class structure. A large, stylized logo of the letters "JUGS" is overlaid on the bottom left of the viewer window.

```

/home/joshua/devel/cpp/cps111/bigcalc/apstring.cpp
234 apstring operator + ( char ch, const apstring &
235 // postcondition: returns concatenation of ch +
236 {
237     apstring result; // make string equivalent
238     result = ch;
239     result += str;
240     return result;
241 }
242
243 apstring operator + ( const apstring & str, char
244 // postcondition: returns concatenation of str
245 {
246     apstring result(str);
247     result += ch;
248     return result;
249 }
250
251
252 apstring apstring::substr(int pos, int len) const
253 // description: extract and return the substr
254 // at index pos
255 // precondition: this string represents c0, c1,
256 // 0 <= pos <= pos + len - 1 < n.

```

```

/home/joshua/devel/cpp/cps111/bigcalc/apstring.h
43
44 // indexing
45
46     char operator[]( int k ) const;
47     char & operator[]( int k );
48
49 // modifiers
50
51     const apstring & operator += ( const apstring & str );
52     const apstring & operator += ( char ch );
53
54
55     private:
56         int myLength; // length
57         int myCapacity; // capacity
58         char * myCString; // string
59     };
60
61 // The following free (non-member) functions operate on apstring
62 //
63 // I/O functions
64

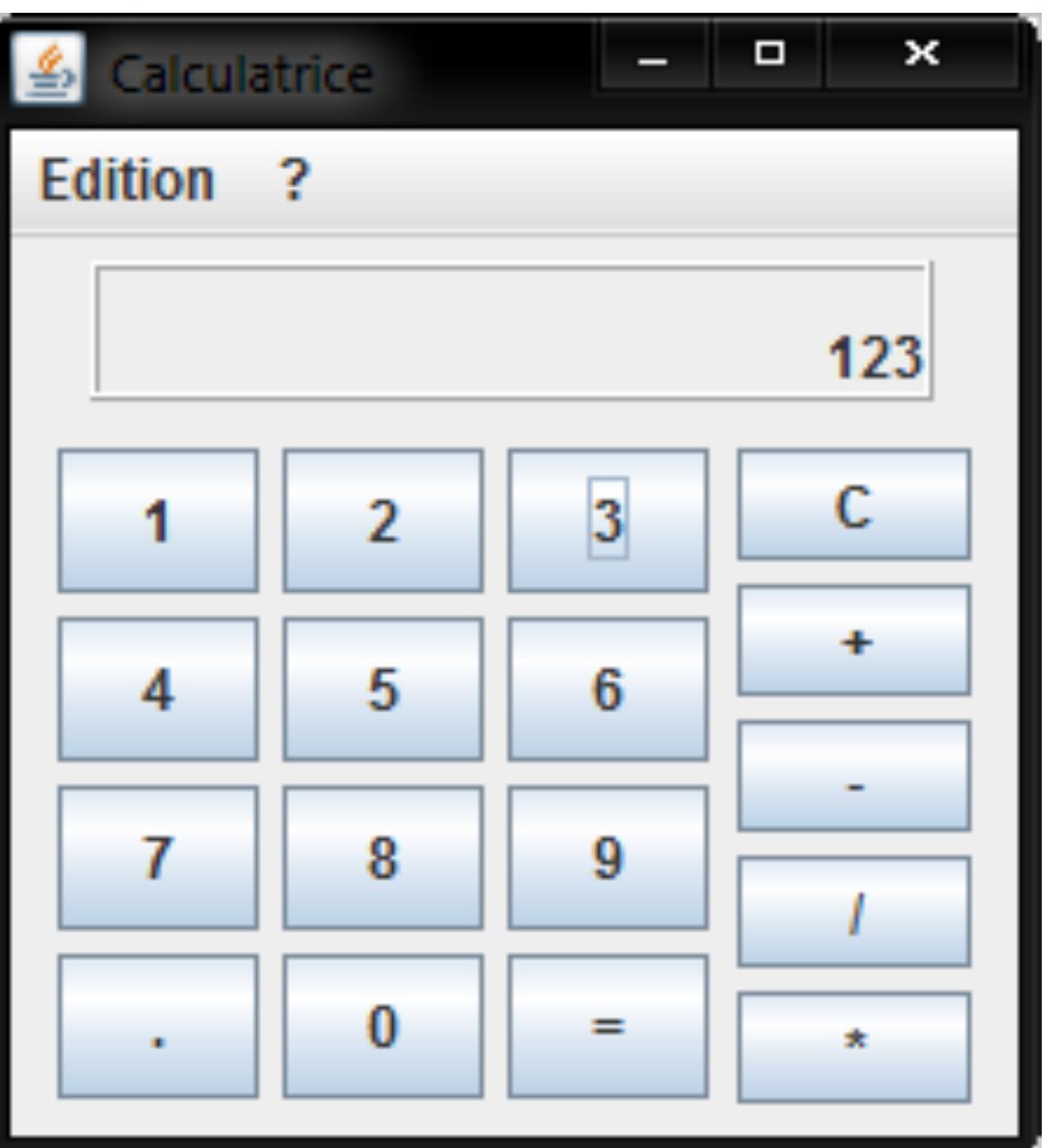
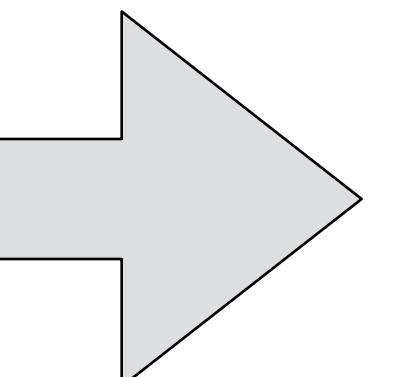
```

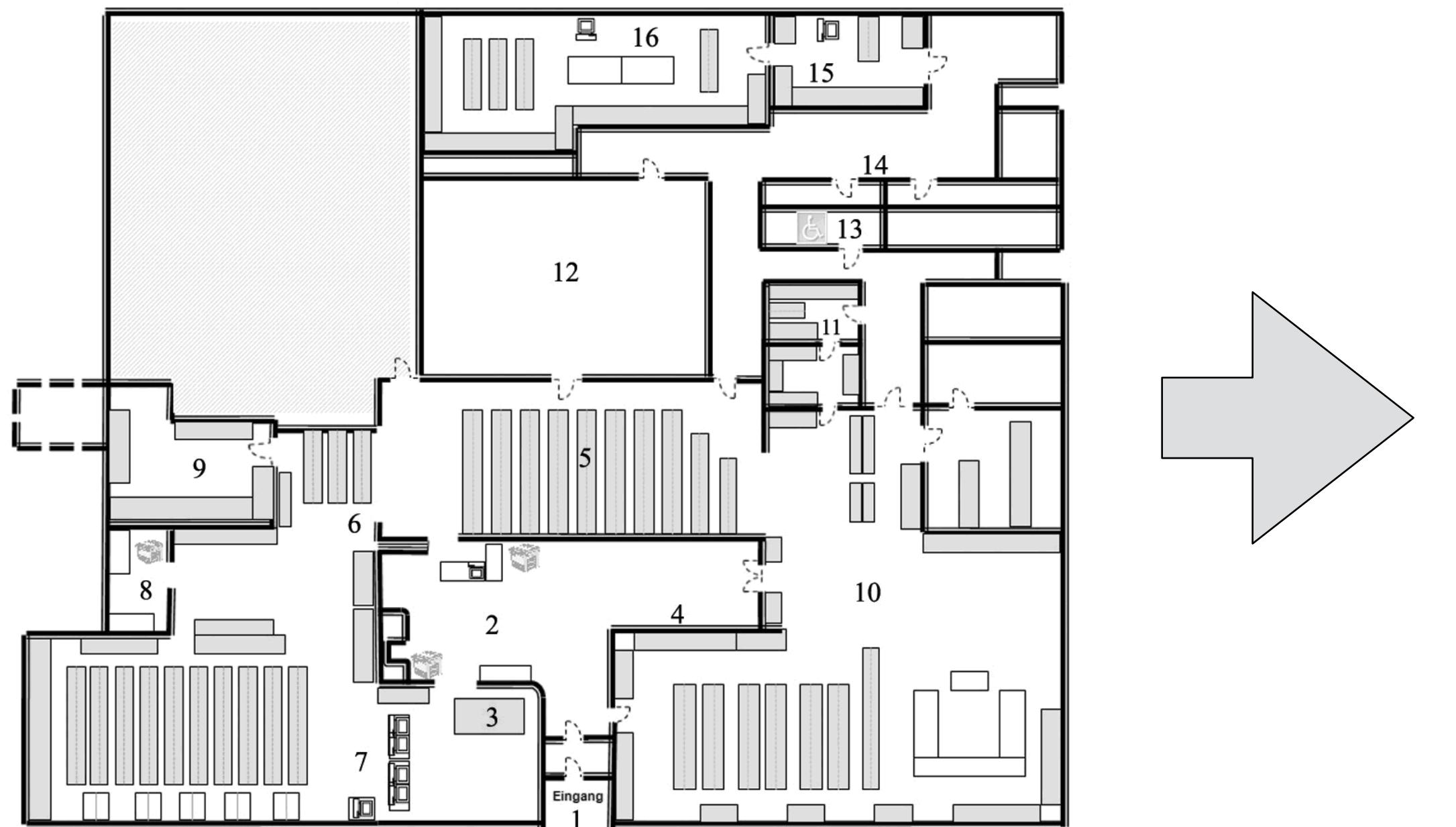
```
function date_mysql2ger($dateex) {
    $t = substr($dateex, 0, 4);
    $list = explode("-", $dateex);
    return sprintf("%02d.%02d.%04d", $list[2], $list[1], $list[0]);
}

function date_ger2mysql($dateex) {
    $list = explode("-", $dateex);
    return sprintf("%04d-%02d-%02d", $list[2], $list[1], $list[0]);
}

function timestamp_mysql2ger($t) {
    $t = substr($t, 6, 2);
    $t = substr($t, 4, 2);
    $t = substr($t, 0, 4);
    return sprintf("%02d.%02d.%04d", $t);
}

function date_mysql2ger($dateex) {
    $list = explode("-", $dateex);
    return sprintf("%02d.%02d.%04d", $list[2], $list[1], $list[0]);
}
```

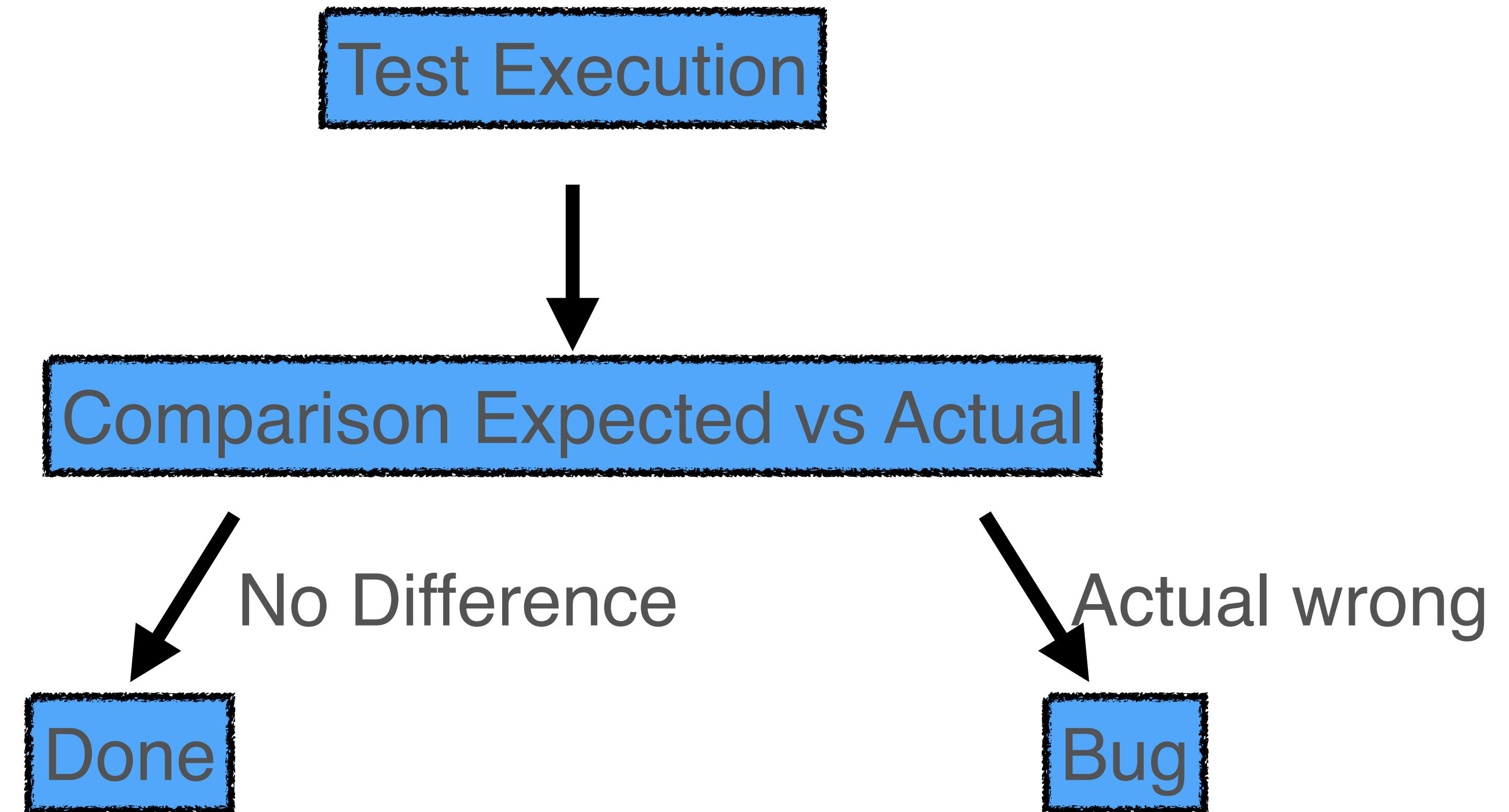


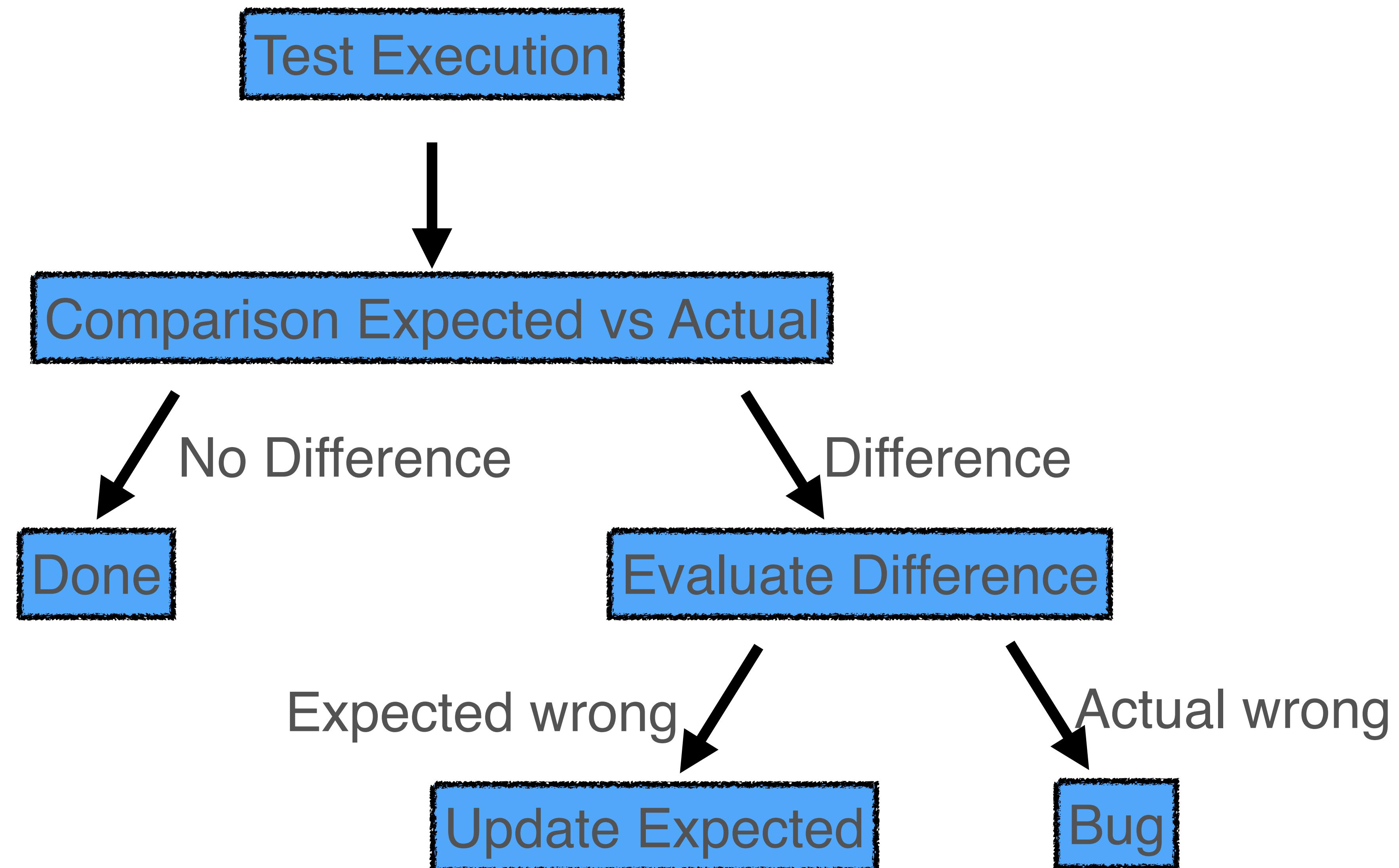


Die Lücke schließen: *automatische* Tests

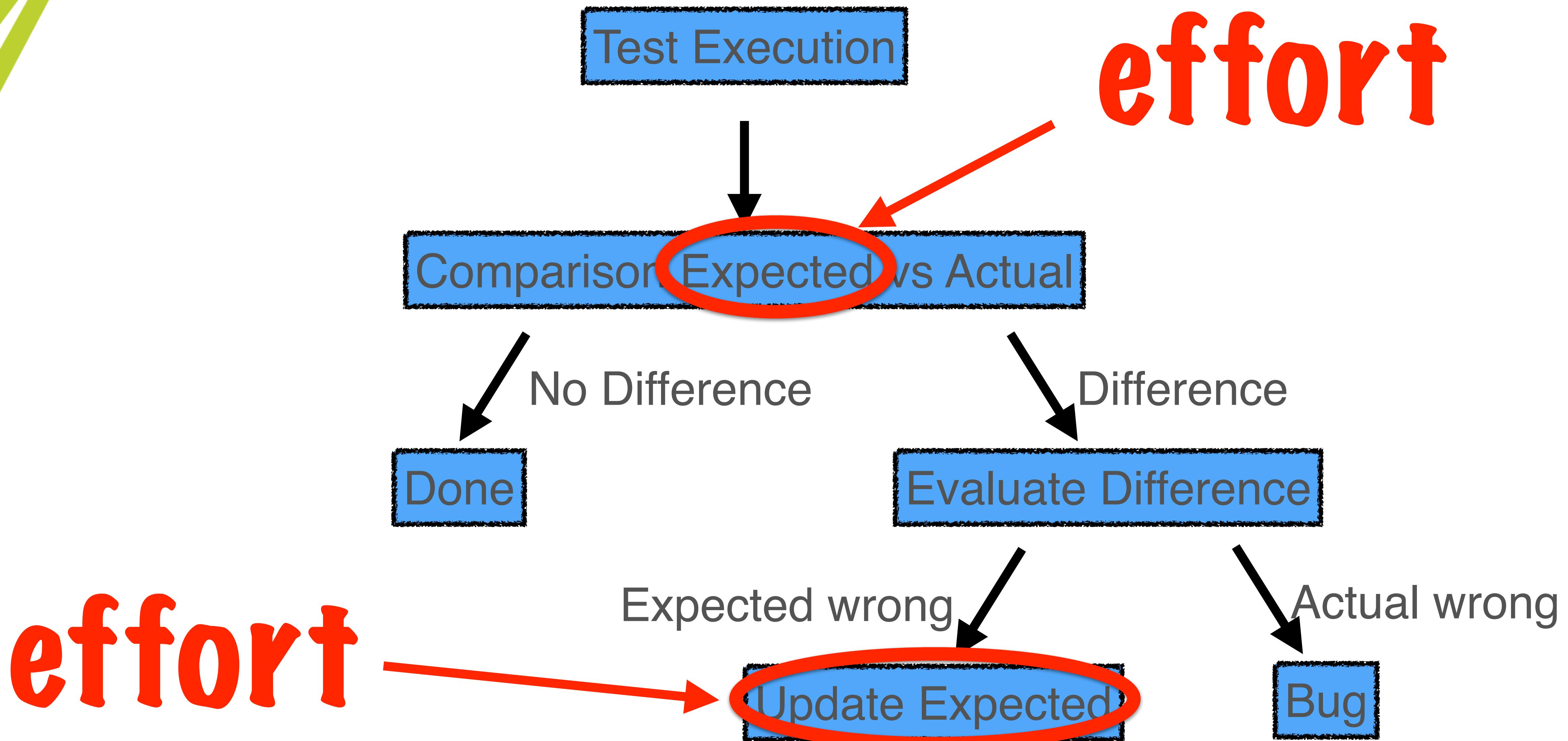
Drawbacks of assertions

Testing





Testing



[Show Tables](#) [Show Views](#) [Show Column](#) [Create View](#) [Drop Table](#) [Create Index](#)

Query time (in millisec) = 47

Number of rows retrieved from database = 22

Reformat	CSV	Plot (VOPlot)								
haloID	subHaloID	lastProgenitorID	treeID	snapNum	redshift	firstProgenitorID	nextProgenitorID	descendantID	firstMolteForwards	lastMolteForwards
526000013	50000000001445	526000033	526000000	50	0.40769944	526000014	-1	526000012	526000013	-1
874000013	50000000001486	874000051	874000000	50	0.40769944	874000014	-1	874000012	874000013	-1
663000013	50000000001461	663000043	663000000	50	0.40769944	663000014	-1	663000012	663000013	-1
623000013	50000000001403	623000042	623000000	50	0.40769944	623000014	-1	623000012	623000013	-1
518000013	50000000001349	518000041	518000000	50	0.40769944	518000014	-1	518000012	518000013	-1
642000013	50000000001354	642000046	642000000	50	0.40769944	642000014	-1	642000012	642000013	-1
602000013	50000000001345	602000043	602000000	50	0.40769944	602000014	-1	602000012	602000013	-1
522000013	50000000001338	522000045	522000000	50	0.40769944	522000014	-1	522000012	522000013	-1
534000013	50000000001324	534000044	534000000	50	0.40769944	534000014	-1	534000012	534000013	-1
589000013	50000000001248	589000041	589000000	50	0.40769944	589000014	-1	589000012	589000013	-1
585000013	50000000001245	585000038	585000000	50	0.40769944	585000014	-1	585000012	585000013	-1
427000013	50000000001204	427000043	427000000	50	0.40769944	427000014	-1	427000012	427000013	-1
476000013	50000000001181	476000059	476000000	50	0.40769944	476000014	-1	476000012	476000013	-1
112000094	50000000001116	112000126	112000000	50	0.40769944	112000095	-1	112000093	112000094	-1
457000013	50000000001084	457000054	457000000	50	0.40769944	457000014	-1	457000012	457000013	-1
307000013	50000000000892	307000041	307000000	50	0.40769944	307000014	-1	307000012	307000013	307000055
375000013	50000000000981	375000054	375000000	50	0.40769944	375000014	-1	375000012	375000013	-1
18000444	50000000001001	18000498	180000000	50	0.40769944	18000445	-1	18000443	18000444	-1

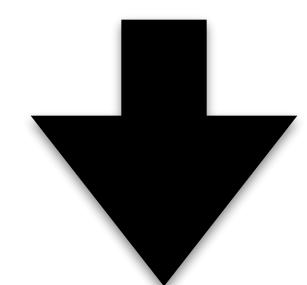
```
Assert.assertEquals(driver.findElement(By.xpath("table['results']/tr[1]/td[1]")), "5260000D13");
Assert.assertEquals(driver.findElement(By.xpath("table['results']/tr[1]/td[2]")), "50000000001445");
Assert.assertEquals(driver.findElement(By.xpath("table['results']/tr[1]/td[3]")), "526000033");
Assert.assertEquals(driver.findElement(By.xpath("table['results']/tr[1]/td[4]")), "52600000");
Assert.assertEquals(driver.findElement(By.xpath("table['results']/tr[1]/td[5]")), "50");
Assert.assertEquals(driver.findElement(By.xpath("table['results']/tr[1]/td[6]")), "0.40789944");
Assert.assertEquals(driver.findElement(By.xpath("table['results']/tr[1]/td[7]")), "526000014");
Assert.assertEquals(driver.findElement(By.xpath("table['results']/tr[1]/td[8]")), "-1");
Assert.assertEquals(driver.findElement(By.xpath("table['results']/tr[1]/td[9]")), "526000012");
Assert.assertEquals(driver.findElement(By.xpath("table['results']/tr[1]/td[10]")), "526000013");
Assert.assertEquals(driver.findElement(By.xpath("table['results']/tr[1]/td[11]")), "-1");
```

```
@Test
public void extended_xml_should_be_removed_correctly() throws Exception {
    // Prepare state
    final String xml = "<?xml version=\"1.0\" encoding=\"UTF-8\" ?>\n" //
        + "<root>\n" //
        + "\t<parent>\n" //
        + "\t\t<remove>should be removed</remove>\n" //
        + "\t\t<keep>should not be removed</keep>\n" //
        + "\t</parent>\n" //
        + "</root>";
    final RemoveElementTransformer transformer = new RemoveElementTransformer( "parent", "remove" );

    // Execute feature under test
    final InputStream transform = transformer.transform( new ByteArrayInputStream( xml.getBytes() ) );

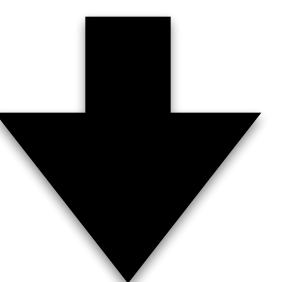
    // Assert result
    final String result = IOUtils.toString( transform, "UTF-8" );
    Assert.assertFalse( result.contains( "remove" ) );
    Assert.assertTrue( result.contains( "keep" ) );
}
```

```
<?xml version="1.0" encoding="UTF-8" ?>
<root>
    <parent>
        <remove>should be removed</remove>
        <keep>should not be removed</keep>
    </parent>
</root>
```



```
<?xml version=\"1.0\" encoding=\"UTF-8\" ?>
<root>
    <parent>
        <keep>should not be removed</keep>
    </parent>
</root>
```

```
<?xml version="1.0" encoding="UTF-8" ?>
<root>
    <parent>
        <remove>should be removed</remove>
        <keep>should not be removed</keep>
    </parent>
</root>
```

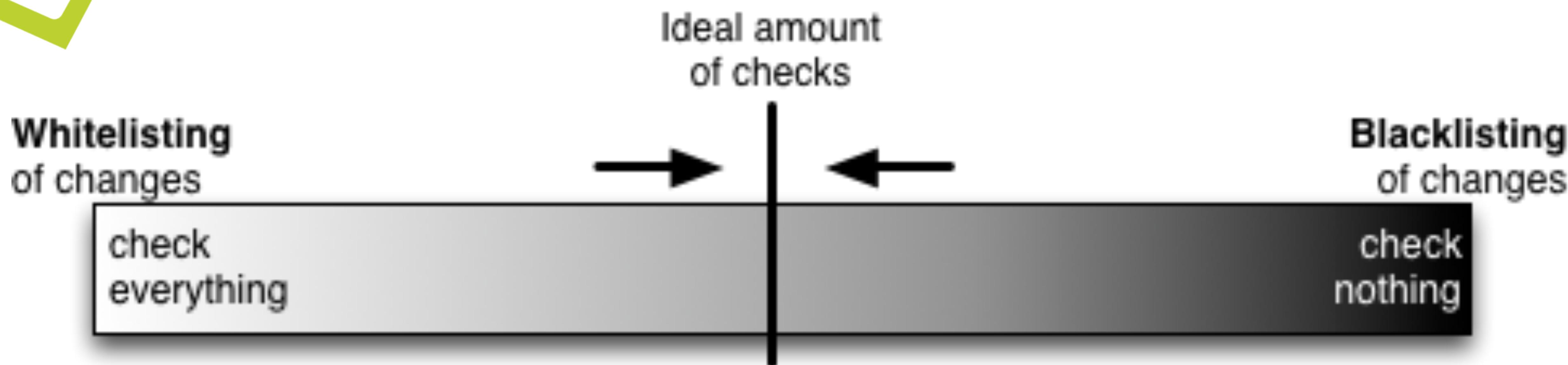


```
<?xml keep</keep>
    </parent>
</root>
```

✓ passes

Golden Master to the rescue





retest.org

1

Multiple choice:



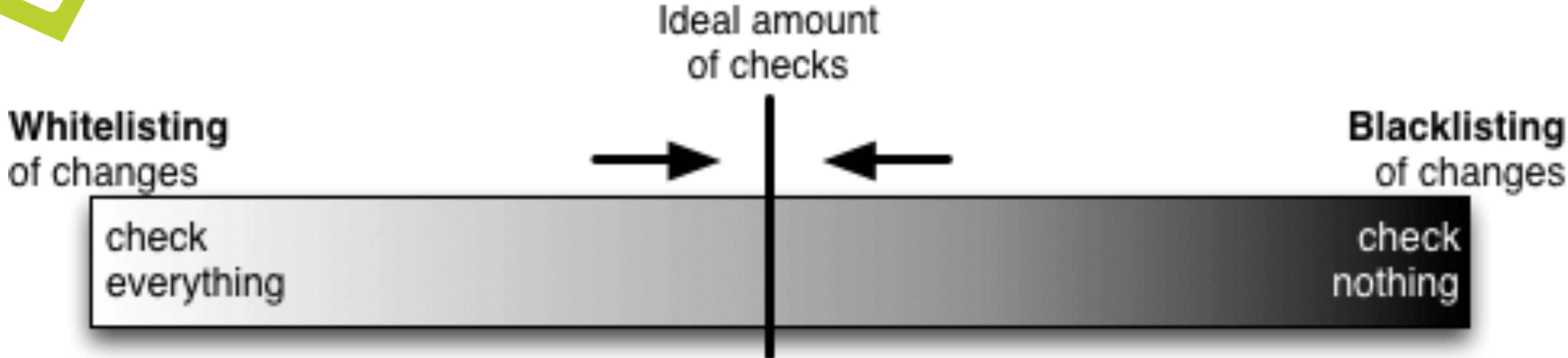
100 characters
remaining

Randomize answer order for each respondent

Randomize answer order for each respondent

enter answer text





retest.org



retest.org

[Show Tables](#) [Show Views](#) [Show Column](#) [Create View](#) [Drop Table](#) [Create Index](#)

Query time (in millisec) = 47

Number of rows retrieved from database = 22

Reformat	CSV	Plot (VOPlot)								
haloId	subHaloId	lastProgenitorId	treeId	snapNum	redshift	firstProgenitorId	nextProgenitorId	descendantId	firstMolteForwards	lastMolteForwards
526000013	50000000001445	526000033	526000000	50	0.40769944	526000014	-1	526000012	526000013	-1
874000013	50000000001486	874000051	874000000	50	0.40769944	874000014	-1	874000012	874000013	-1
663000013	50000000001461	663000043	663000000	50	0.40769944	663000014	-1	663000012	663000013	-1
623000013	50000000001403	623000042	623000000	50	0.40769944	623000014	-1	623000012	623000013	-1
518000013	50000000001349	518000041	518000000	50	0.40769944	518000014	-1	518000012	518000013	-1
642000013	50000000001354	642000046	642000000	50	0.40769944	642000014	-1	642000012	642000013	-1
602000013	50000000001345	602000043	602000000	50	0.40769944	602000014	-1	602000012	602000013	-1
522000013	50000000001338	522000045	522000000	50	0.40769944	522000014	-1	522000012	522000013	-1
534000013	50000000001324	534000044	534000000	50	0.40769944	534000014	-1	534000012	534000013	-1
589000013	50000000001248	589000041	589000000	50	0.40769944	589000014	-1	589000012	589000013	-1
585000013	50000000001245	585000038	585000000	50	0.40769944	585000014	-1	585000012	585000013	-1
427000013	50000000001204	427000043	427000000	50	0.40769944	427000014	-1	427000012	427000013	-1
476000013	50000000001181	476000059	476000000	50	0.40769944	476000014	-1	476000012	476000013	-1
112000094	50000000001116	112000126	112000000	50	0.40769944	112000095	-1	112000093	112000094	-1
457000013	50000000001084	457000054	457000000	50	0.40769944	457000014	-1	457000012	457000013	-1
307000013	50000000000892	307000041	307000000	50	0.40769944	307000014	-1	307000012	307000013	307000055
375000013	50000000000981	375000054	375000000	50	0.40769944	375000014	-1	375000012	375000013	-1
18000444	50000000001001	18000498	180000000	50	0.40769944	18000445	-1	18000443	18000444	-1

Even better...

```
private WebDriver driver;

@Before
public void setup() {
    System.setProperty("webdriver.chrome.driver", "src/test/resources/chromedriver");
    driver = new ChromeDriver();
    driver.manage().timeouts().pageLoadTimeout(-1, TimeUnit.MINUTES);
}

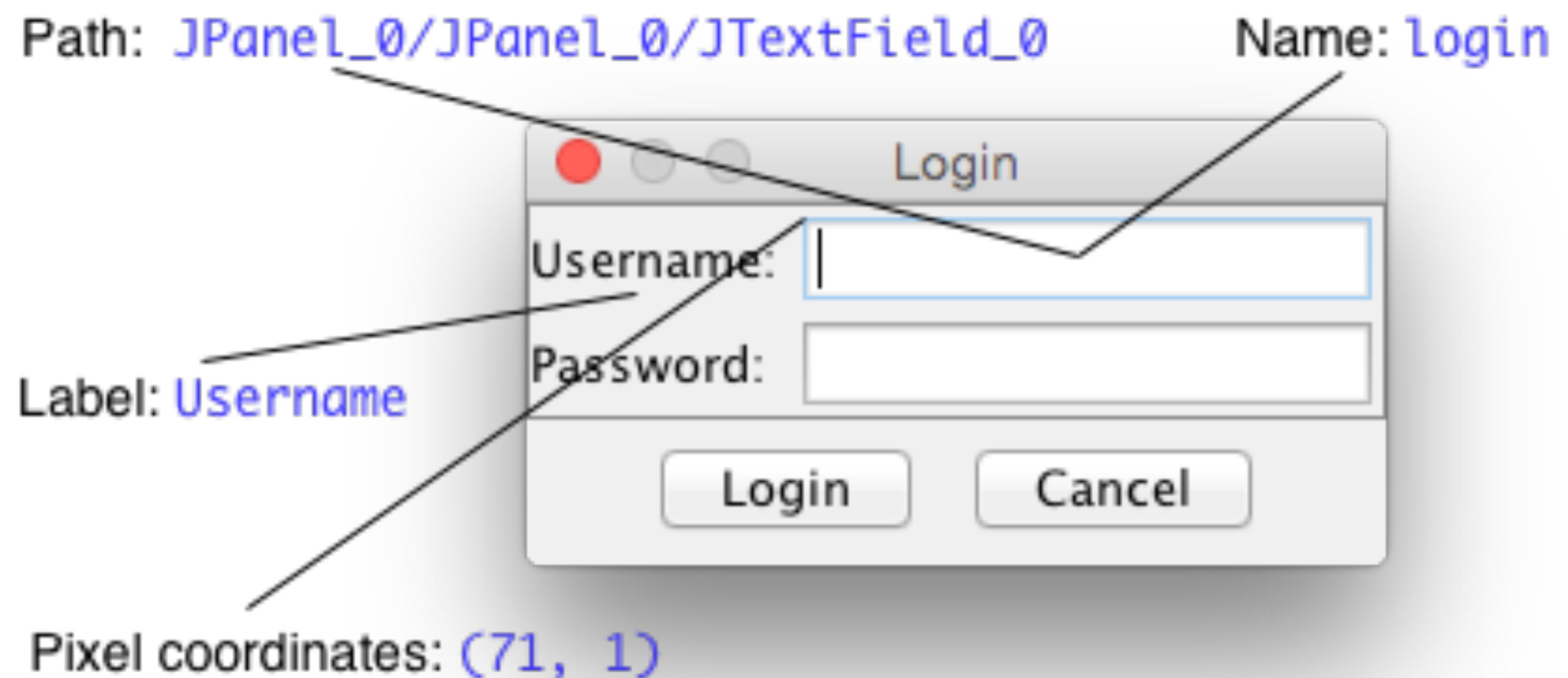
@Test
public void check_add() throws Exception {
    driver.get("http://localhost:8080/login");
    // login
    driver.findElement(By.name("username")).sendKeys("Max");
    driver.findElement(By.name("password")).sendKeys("retest");
    driver.findElement(By.name("login")).click();

    // test
    driver.findElement(By.id("1")).click();
    driver.findElement(By.id("plus")).click();
    driver.findElement(By.id("1")).click();
    driver.findElement(By.id("equals")).click();

    Assert.assertEquals(driver.findElement(By.name("result")).getAttribute("value"), "2.0");
}

@After
public void tearDown() {
    driver.quit();
}
```

Multilocators



Lastname	Mayer	Salutation	Mr.
Forename	Andrew	Lastname	Andrew
Street	350 5th Avenue	Forename	Mayer
City	New York	Street and M	350 5th Avenue
Postal Code	10118	City	New York
Accept	Cancel	Save	Cancel

```
graph LR; L1[Lastname Mayer] --> L2[Lastname Andrew]; L1 --> F1[Forename Andrew]; S1[Street 350 5th Avenue] --> S2[Street and M]; C1[City New York] --> P1[Postal Code 10118]; C1 --> C2[City New York]; A1[Accept] --- C1; C1 --- S3[Save]; C2 --- C4[Cancel]; S3 --- C4;
```

Tool support

wearefriday / spectre

CSS Critic



WRAITH



PhantomCSS

gemini-testing / gemini

Needle 0.1a1

DIFF.IO



backTrac

applitools

rprt / rspec-page-regression

ShooV

automated-screenshot-diff

bslatkin / dpxdt

Visual Review

Ghost Inspector

Pix-Diff

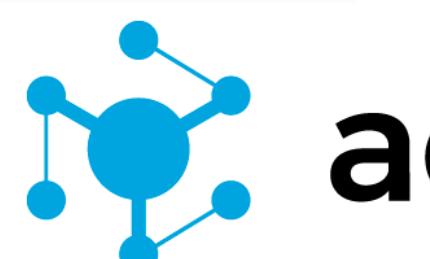


SCREENSTER

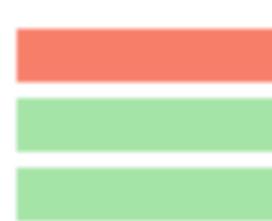
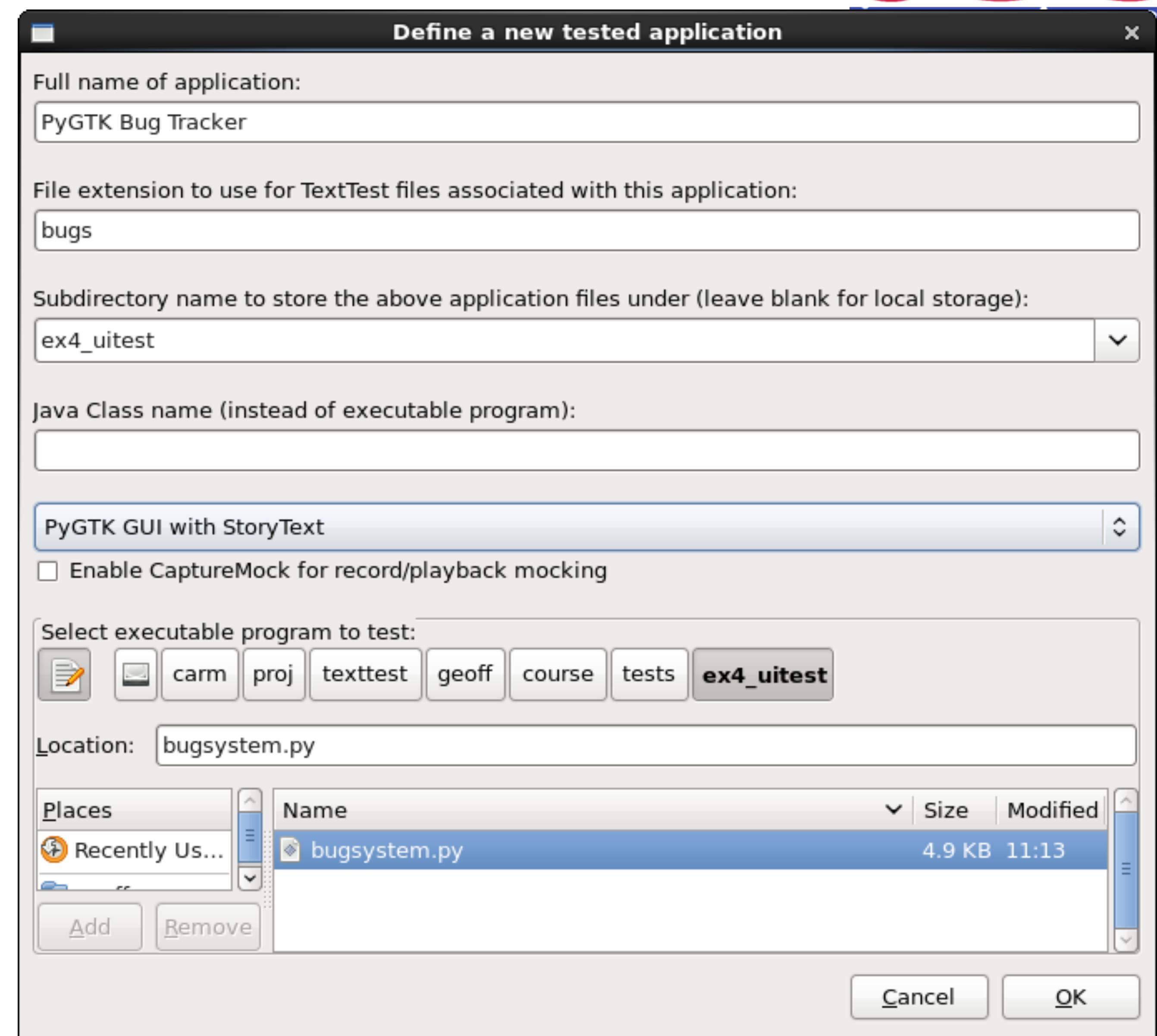
Blink-diff



Kobold



aet

 **TextTest**



Approval Tests

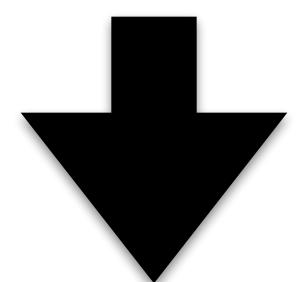
```
[TestMethod]
public void TestPurchase()
{
    // Create a new receipt
    var r = new Receipt();
    // Add 1 candy bar at $.50
    r.AddItem(1, "Candy Bar", 0.50);
    // Add 2 sodas bar at $1
    r.AddItem(2, "Soda", 1.0);
    // verify the receipt
    Approvals.Verify(r.ToString());
}
```

ReceiptTest.TestPurchase.receipt	ReceiptTest.TestPurchase.app
1	1
2 1-Candy-Bar @ \$0.50	2 1-Candy-Bar @ \$0.50
3 2-Soda @ \$1.00 = \$2.00	3 2-Soda @ \$1.00 = \$2.00
4 ... Subtotal = \$2.50	+4 Total = \$2.50
5 --> Tax (10%) = \$0.25	
6 --> Total = \$2.75	

```
@Test
public void extended_xml_should_be_removed_correctly_Approvals() throws Exception {
    // Prepare state
    final String xml = "<?xml version=\"1.0\" encoding=\"UTF-8\" ?>\n" //
        + "<root>\n" //
        + "\t<parent>\n" //
        + "\t\t<remove>should be removed</remove>\n" //
        + "\t\t<keep>should not be removed</keep>\n" //
        + "\t</parent>\n" //
        + "</root>";
    final RemoveElementTransformer transformer = new RemoveElementTransformer( "parent", "remove" );

    final InputStream transform = transformer.transform( new ByteArrayInputStream( xml.getBytes() ) );
    ApprovalsUtil.verifyXml( IOUtils.toString( transform, "UTF-8" ) );
}
```

```
<?xml version="1.0" encoding="UTF-8" ?>
<root>
    <parent>
        <remove>should be removed</remove>
        <keep>should not be removed</keep>
    </parent>
</root>
```



```
<?xml version="1.0" encoding="UTF-8" ?>
<root>
    <parent>
        <keep>should not be removed</keep>
    </parent>
</root>
```



Jest 23.0

Docs API Help Blog A ⚙ English Search

GitHub

IIGS
up stuttgart

Jest

Delightful JavaScript Testing

TRY OUT JEST

GET STARTED

WATCH TALKS

LEARN MORE

★ Star 18,072



Developer Ready

Complete and ready to set-up JavaScript testing solution. Works out of the box for any React project.



Instant Feedback

Fast interactive watch mode runs only test files related to changed files and is optimized to give signal quickly.



Snapshot Testing

Capture snapshots of React trees or other serializable values to simplify testing and to analyze how state changes over time.

```
</script>

```

Screens: <input type="checkbox" name="component01fb193bb67c2b69cb52e802bc6ae256e6821c44b7bba170a300267">

Tests:

Screens: <input type="checkbox" name="component3da99a586c4486a95e016670576d623af13b864da05dd3fa4e1ba9b">

Tests:

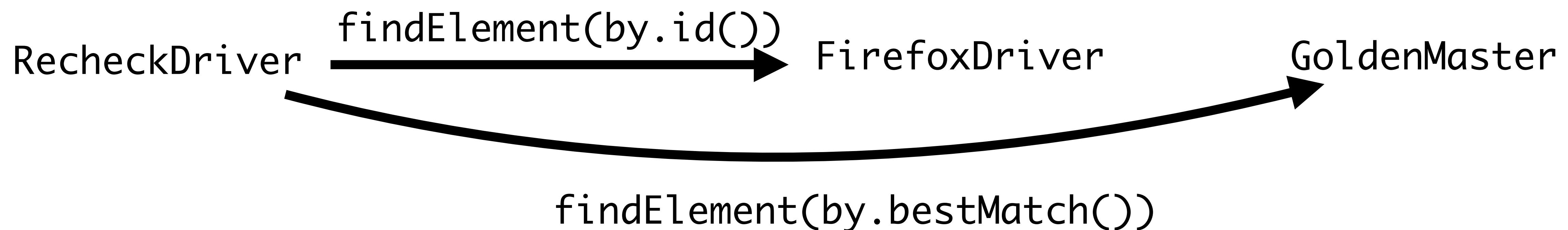
Screens: <input type="checkbox" name="component400be9709e7ce5ef547eaac73a7de3ac17e14d528fe2714499501c31">

Tests:

Screens:

```
@Test  
public void test_login() throws Exception {  
    final WebDriver driver = new FirefoxDriver();  
    driver.get( "http://www.retest.de" );  
    recheck.check( driver );  
  
    driver.findElement( By.id( "username" ) ).sendKeys( username );  
    driver.findElement( By.id( "pwd" ) ).sendKeys( password );  
    driver.findElement( By.className( "button" ) ).submit();  
    recheck.check( driver );  
}
```

```
@Test  
public void test_login() throws Exception {  
    final WebDriver driver = new RecheckDriver(new FirefoxDriver());  
    driver.get( "http://www.retest.de" );  
    recheck.check( driver );  
  
    driver.findElement( By.id( "username" ) ).sendKeys( username );  
    driver.findElement( By.id( "pwd" ) ).sendKeys( password );  
    driver.findElement( By.className( "button" ) ).submit();  
    recheck.check( driver );  
}
```



```
***** recheck warning *****
The HTML id attribute used for element identification changed from 'intro-slider' to 'introSlider'.
retest identified the element based on the persisted old state.
If you apply these changes to the state , your test will break.
Use `By.id("introSlider")` or `By.retestId("9c40281d-5655-4ffa-9c6d-d079e01bb5a3")` to update your test.
```

```
@Test  
public void test_login() throws Exception {  
    final WebDriver driver = new RecheckDriver(new FirefoxDriver());  
    driver.get( "http://www.retest.de" );  
    recheck.check( driver );  
  
    driver.findElement( By.retestId( "username" ) ).sendKeys( username );  
    driver.findElement( By.retestId( "pwd" ) ).sendKeys( password );  
    driver.findElement( By.retestId( "button" ) ).submit();  
    recheck.check( driver );  
}
```

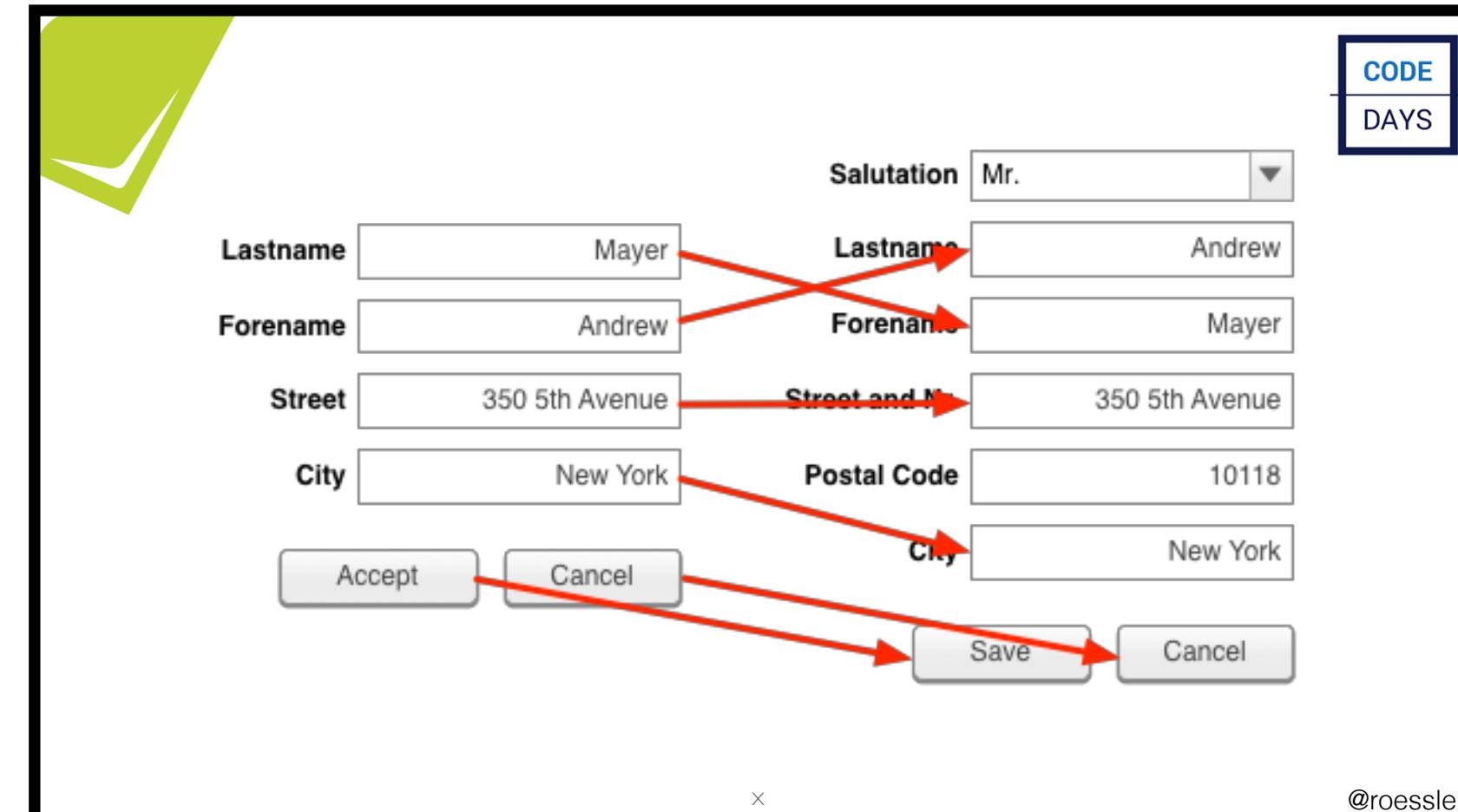
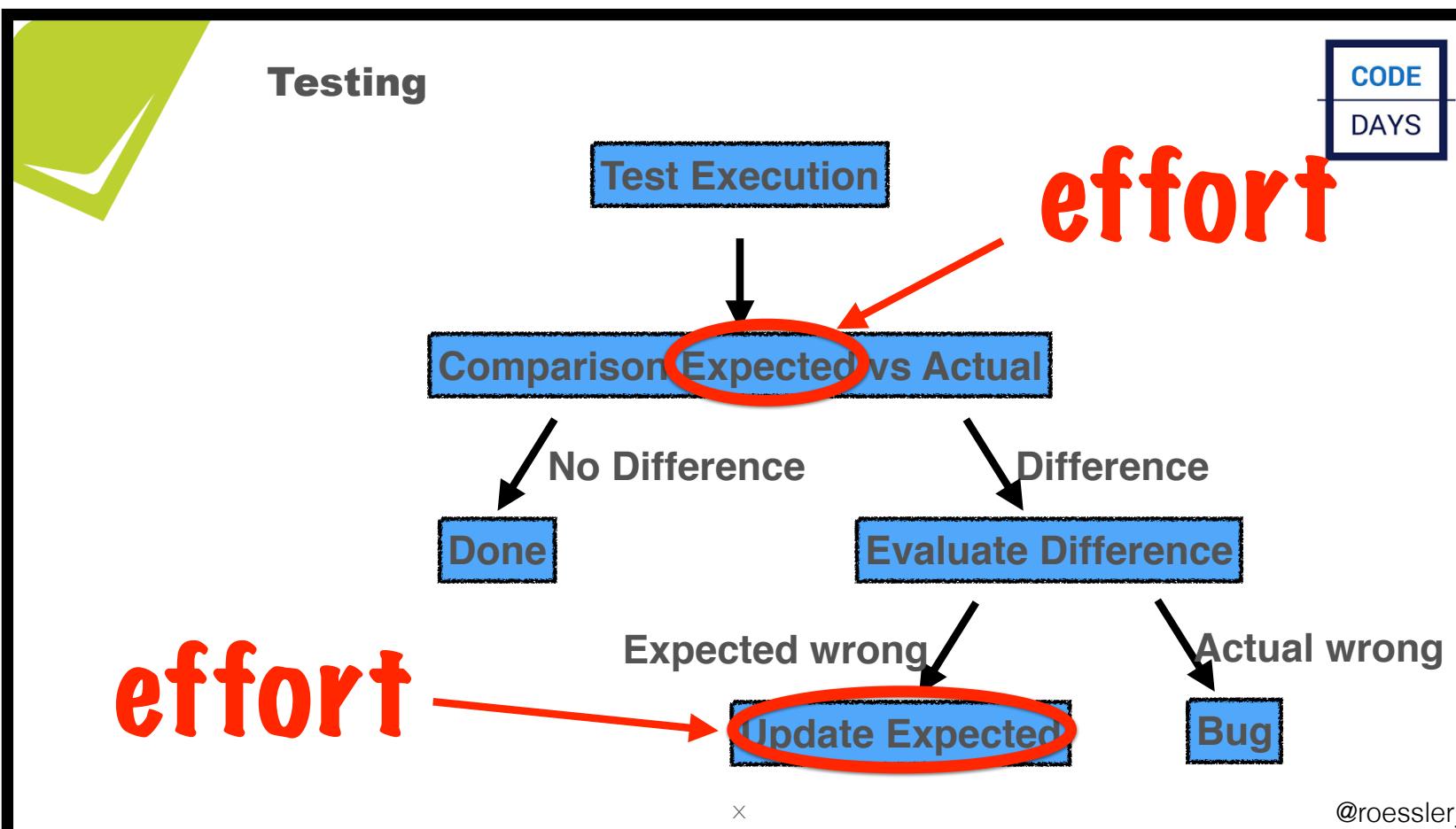
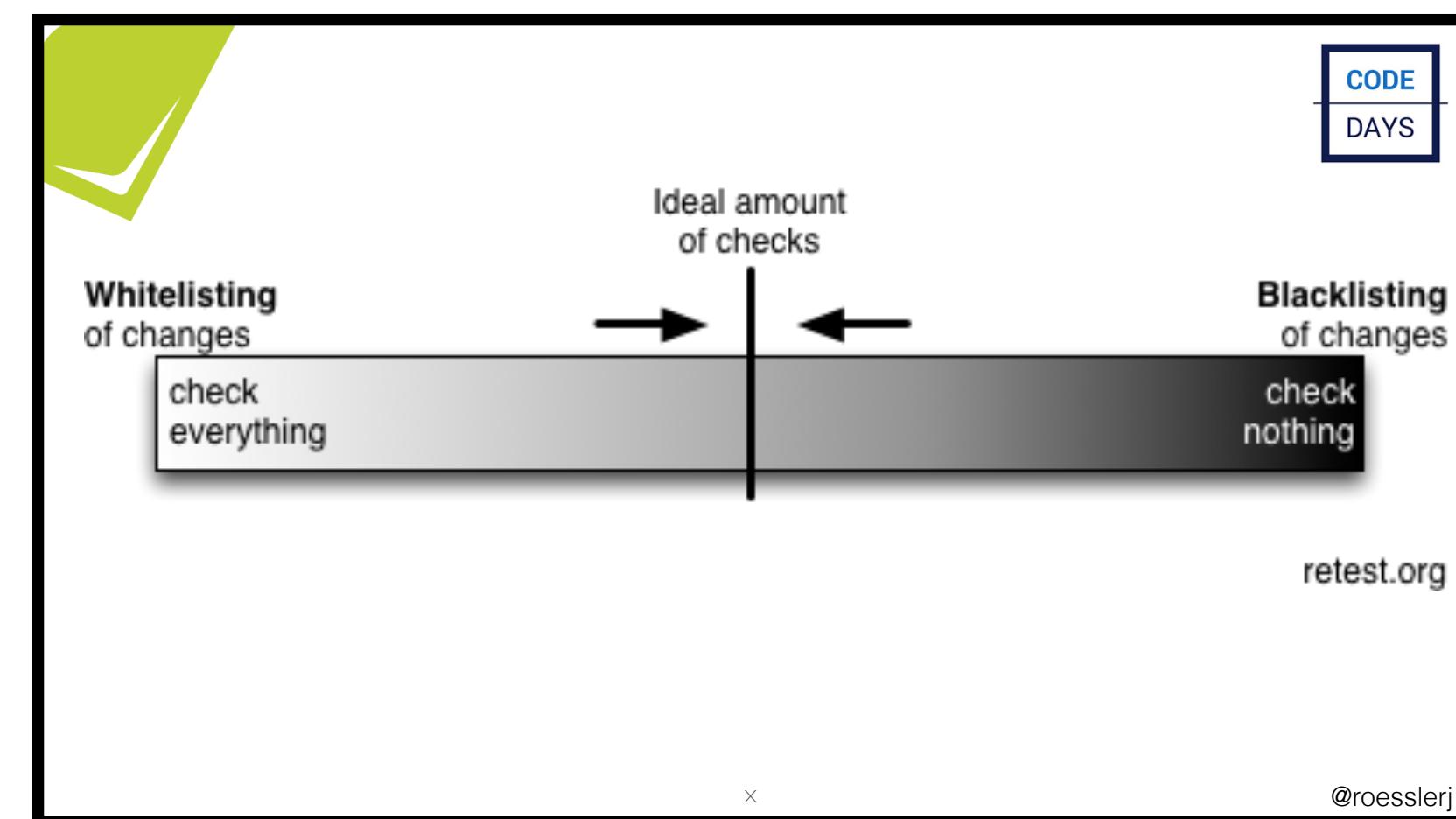
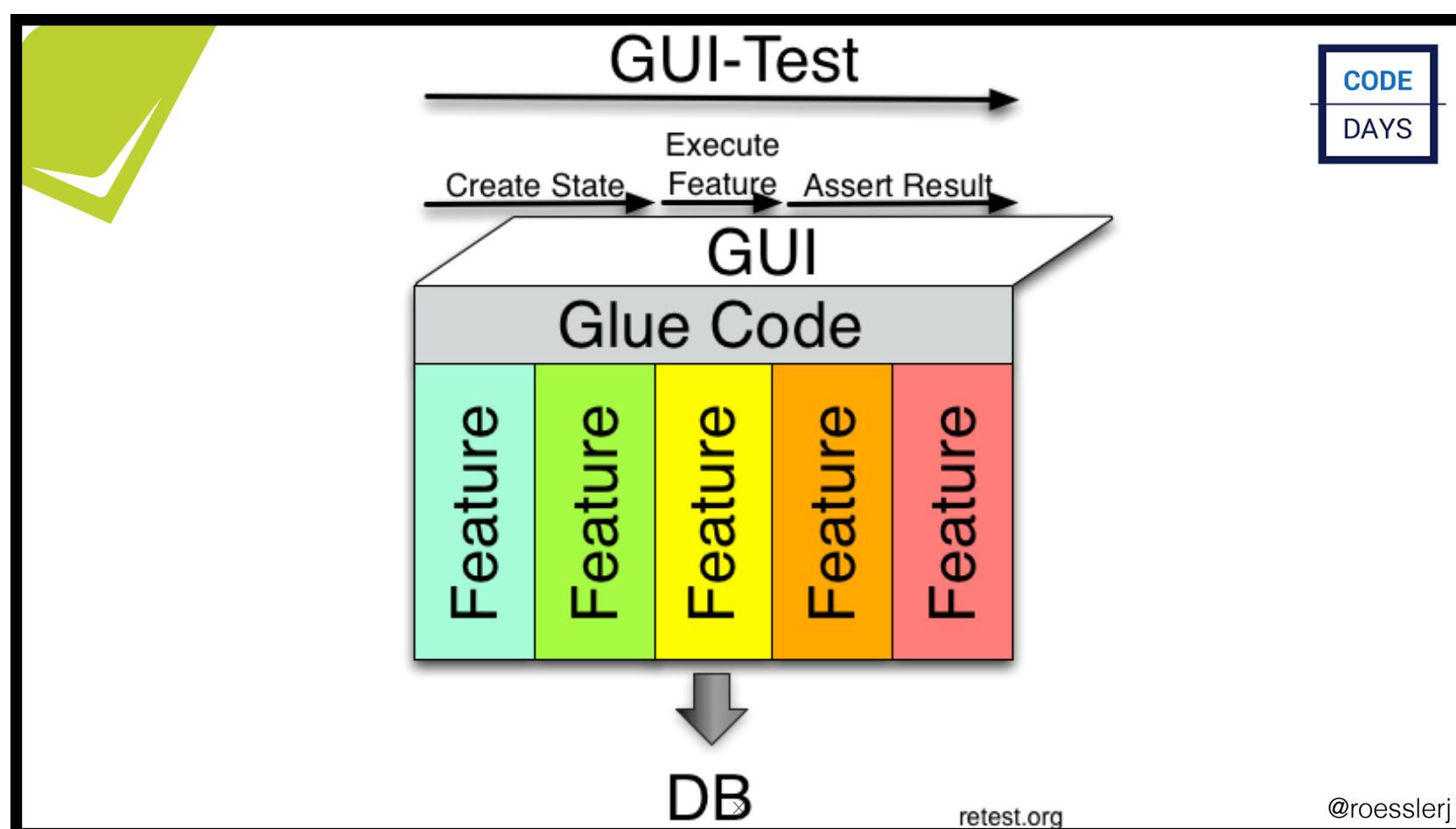
<https://github.com/retest/recheck-web>

The screenshot shows the GitHub README.md page for the recheck project. At the top left is a green checkmark icon. The title "recheck for web apps" is displayed with "recheck" in large blue font and "for web apps" in smaller black font. Below the title are several status badges: "build passing", "maven central 0.6.2", "license AGPL", "PRs welcome", and "</> with ❤ by retest". A descriptive paragraph states: "recheck is a Golden Master-based test framework on top of Selenium that comes with powerful features. Check everything at once and create unbreakable tests." Below this is a Java code editor window showing the beginning of a test class:

```
17 public class SeleniumLargeTest {  
18  
19     private Vertx vertx;  
20     private WebDriver driver;  
21     private Recheck re;  
22
```

DEMO

Questions?



Thank you!



@roesslerj



roessler@retest.de



<https://www.facebook.com/retest.de>



<http://retest.rocks>