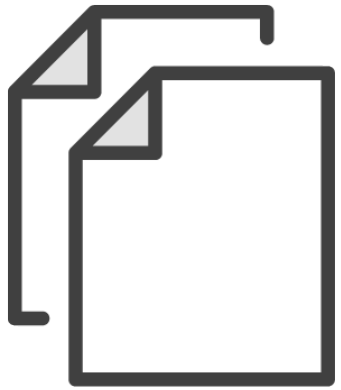
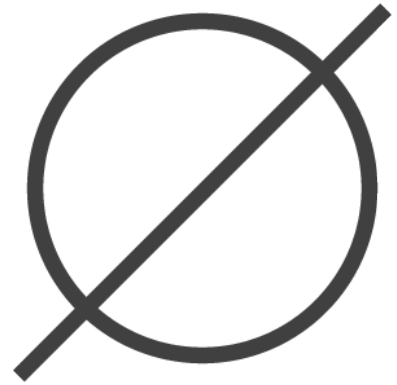
Annotations and Assertions

Andrejs Doronins
TEST AUTOMATION ENGINEER

Assertions





assertTrue()

assertFalse()

assertEquals()

assertNotEquals()

assertNull()

assertNotNull()

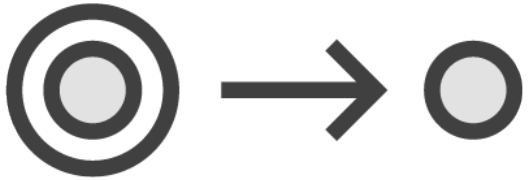
AAA

Arrange

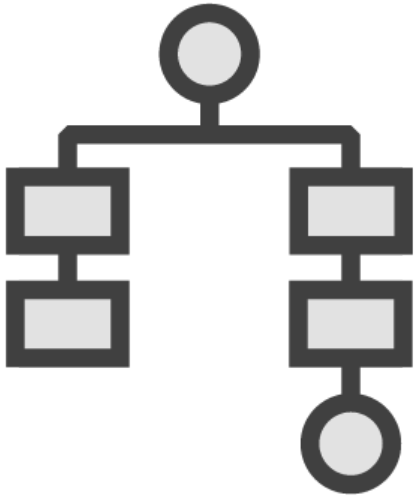
Act

Assert

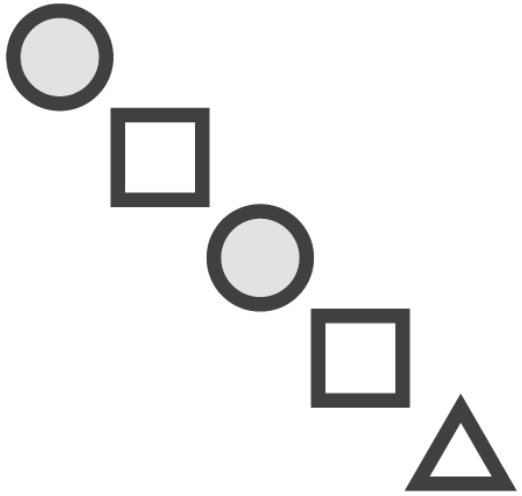
Assertions



assertSame()



assertEqualsDeep()



assertEqualsNoOrder()

Hard Assert

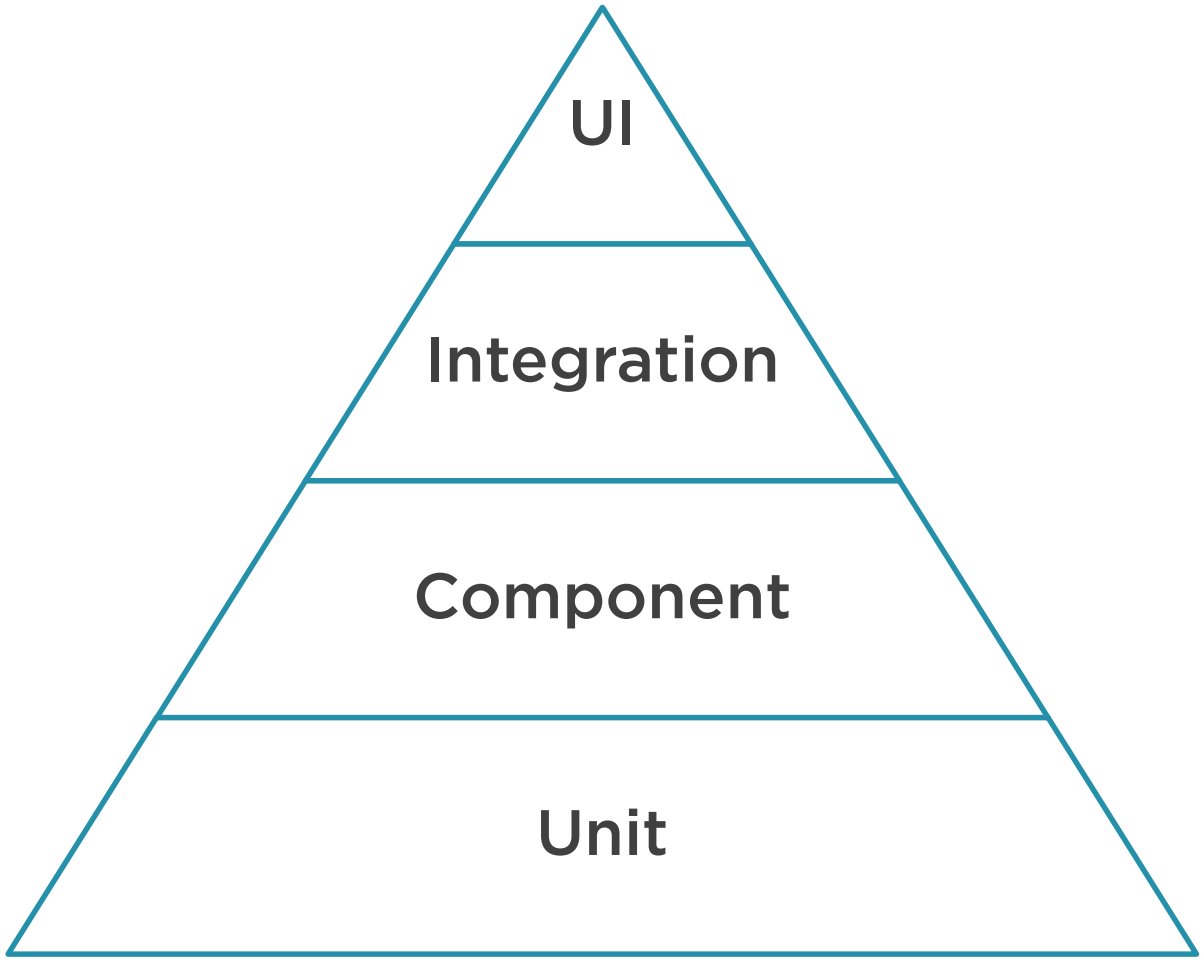
```
@Test
public void verifyX(){
    ...
    // Assert
    Assert.assertNotNull(v1);          X
    Assert.assertEquals(v2,v3);        // Won't run
    Assert.assertEquals(v4,v5);        // Won't run
}
```

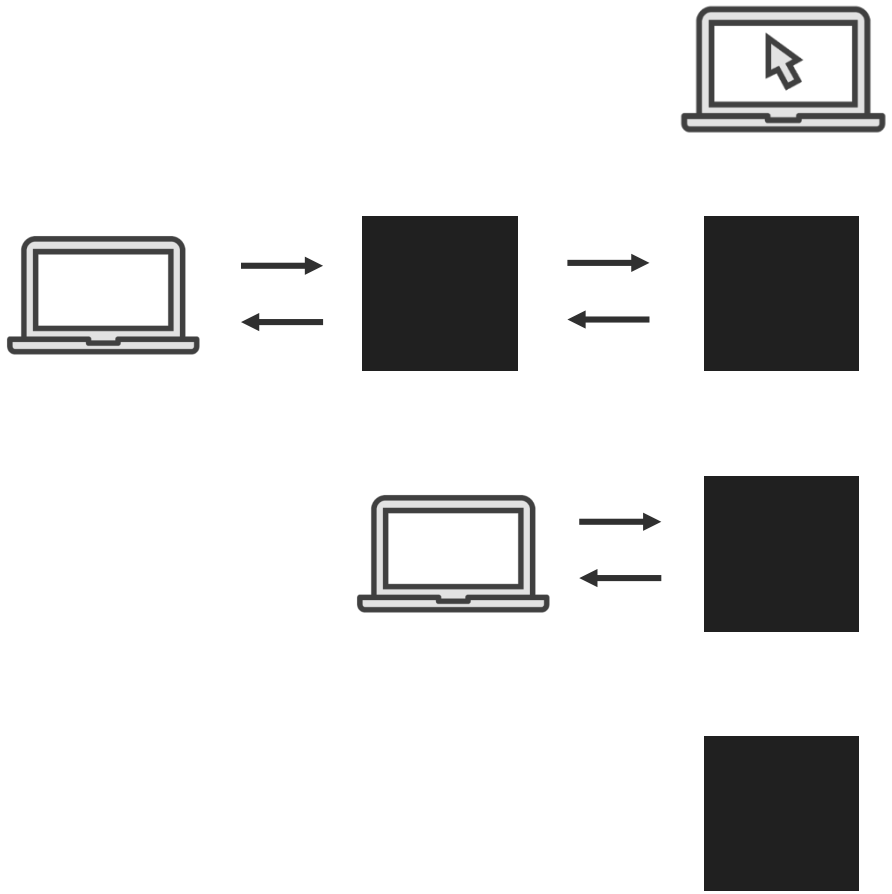
Soft Assert

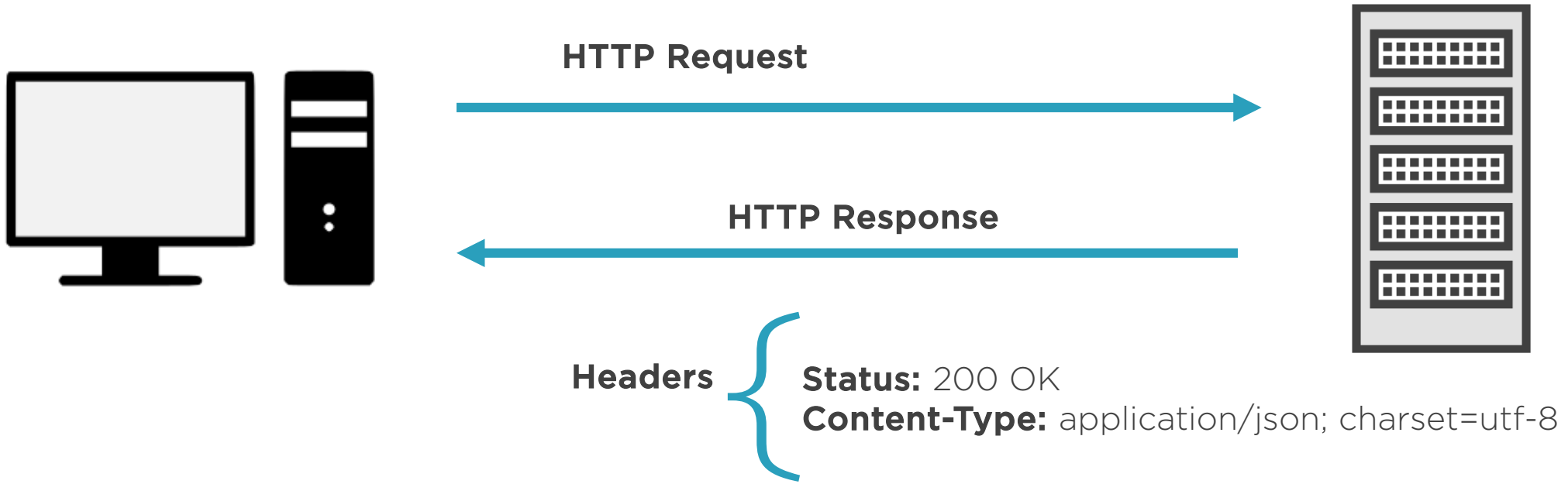
```
@Test
public void verifyZ(){
    ...
    // Assert
    SoftAssert sa = new SoftAssert();
    sa.assertNotNull(v1);
    sa.assertEquals(v2,v3);
    sa.assertAll();
}
```

X

```
// Will run
// Throw all exceptions
```


UI
Integration
Component
Unit




HTTP Request
HTTP Response
Headers
Status: 200 OK
Content-Type: application/json; charset=utf-8

Single Responsibility Principle for Tests

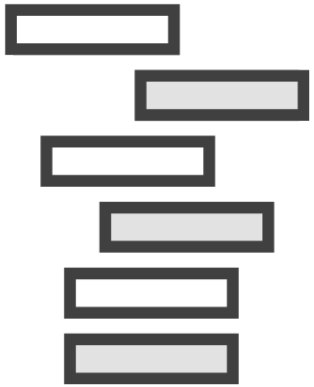
```
@Test
public void verifyX(){
    // Assert
    assertNotNull(v1);
}
```

```
@Test
public void verifyZ(){
    // Assert
    assertEquals(v1,v2);
}
```

Soft Assert Cases

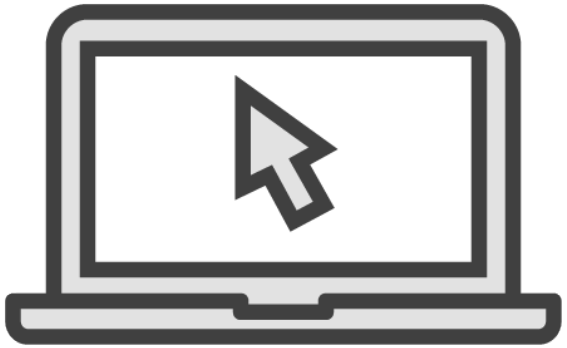
Unit

Almost never



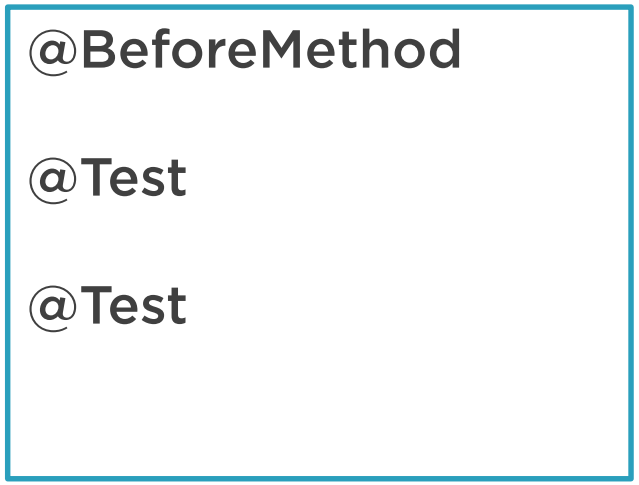
Component & Integration

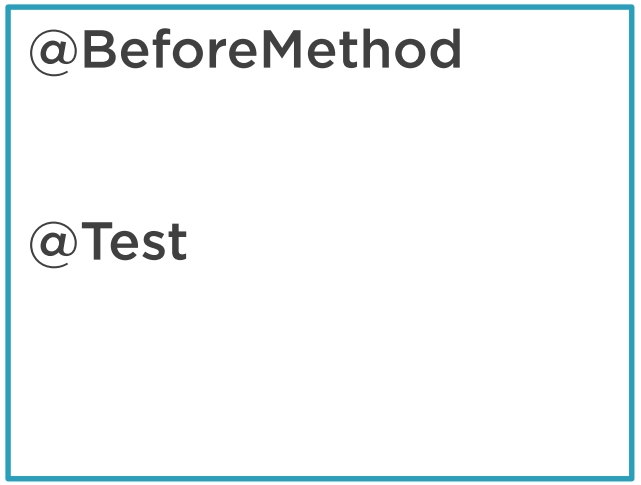
Infrequently

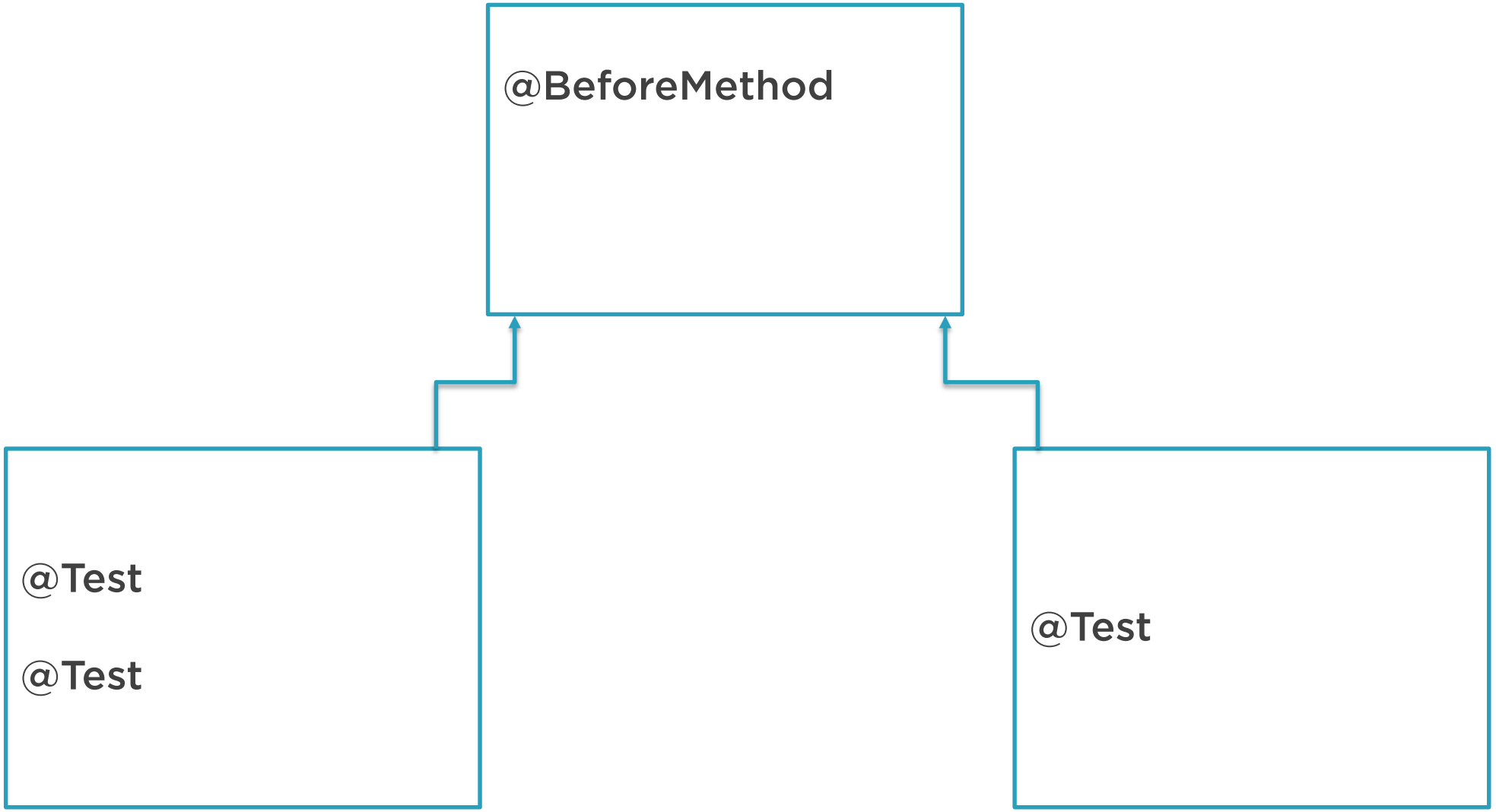


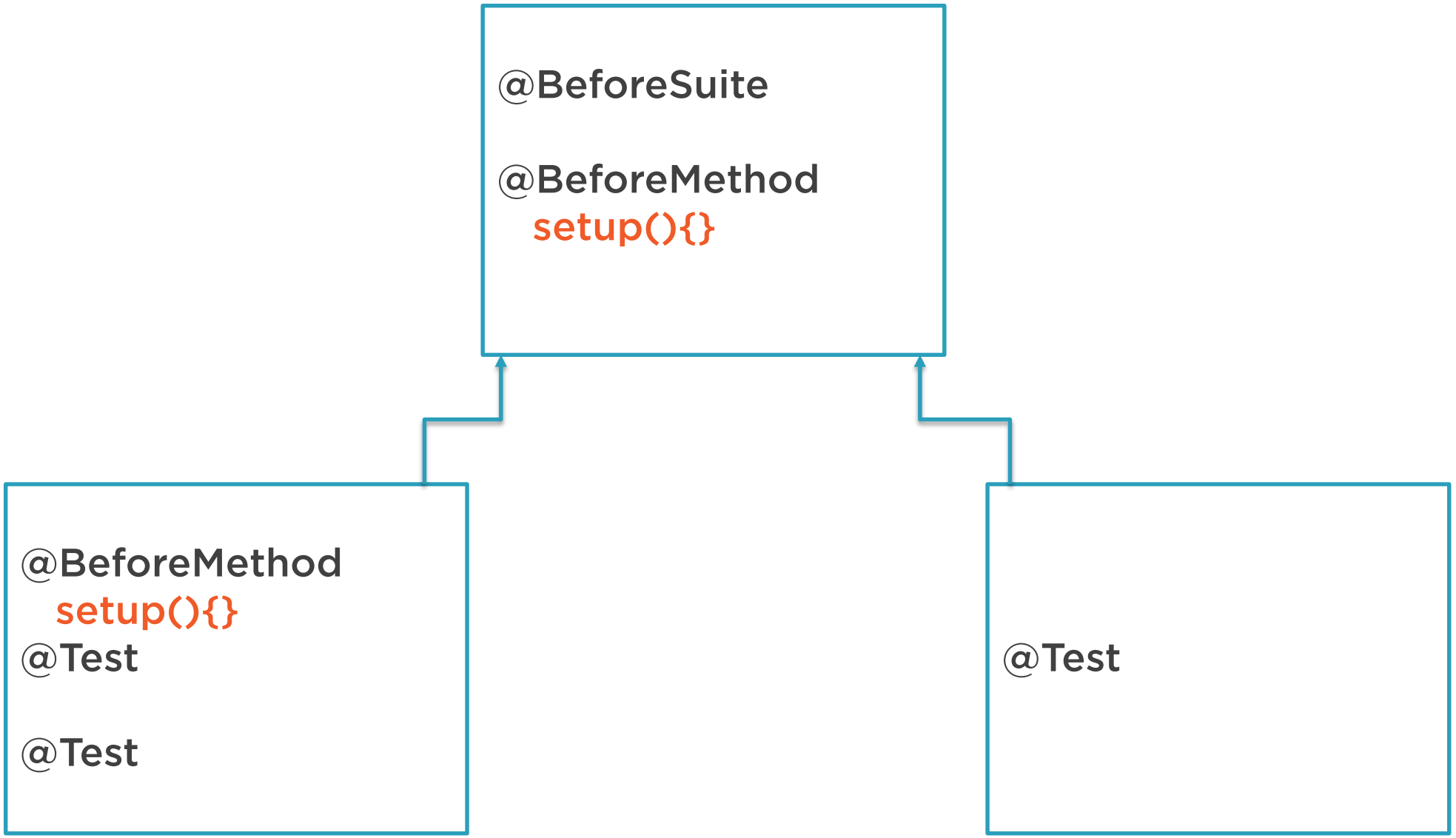
UI

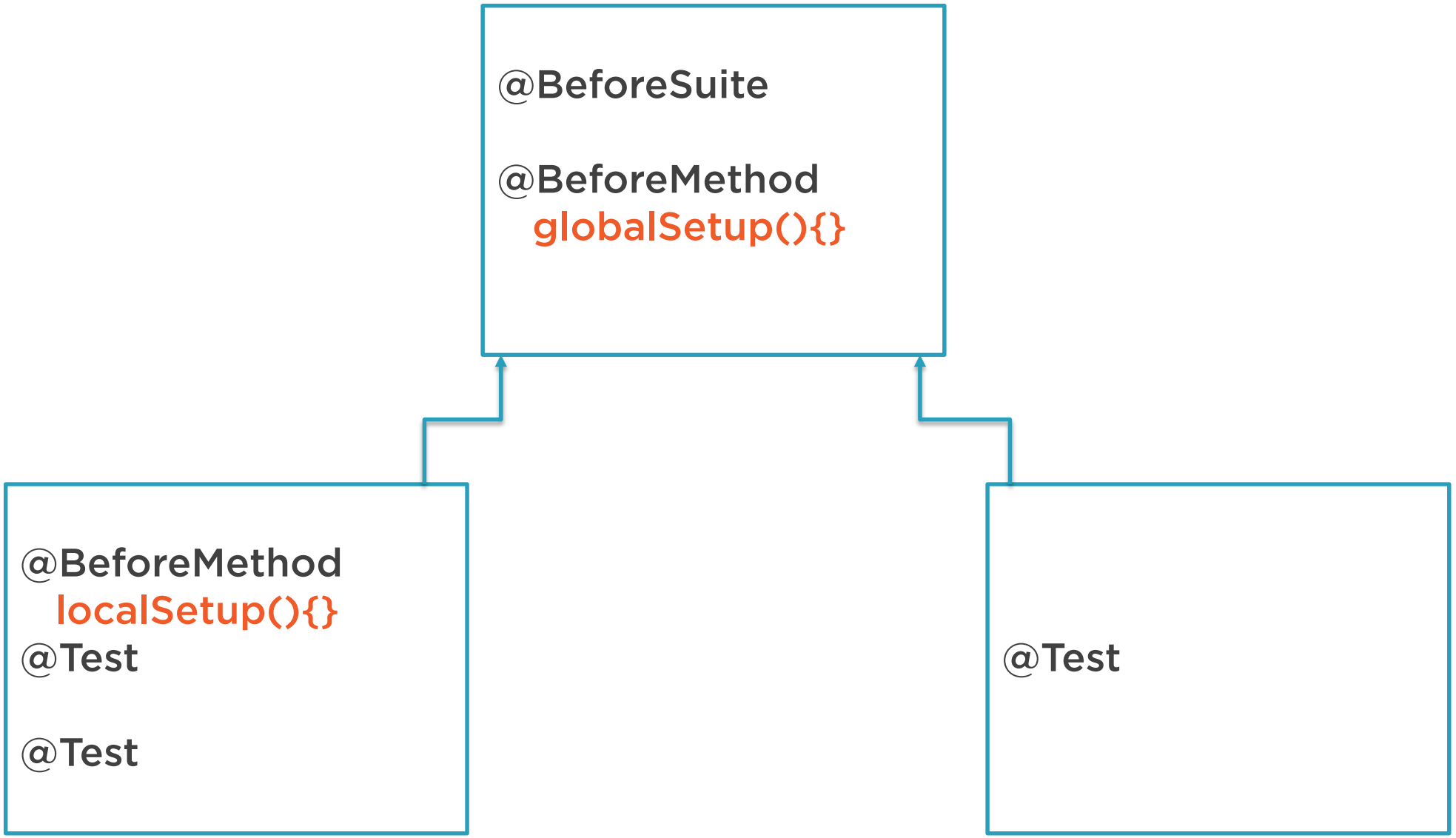
Sometimes


@BeforeMethod
@Test
@Test

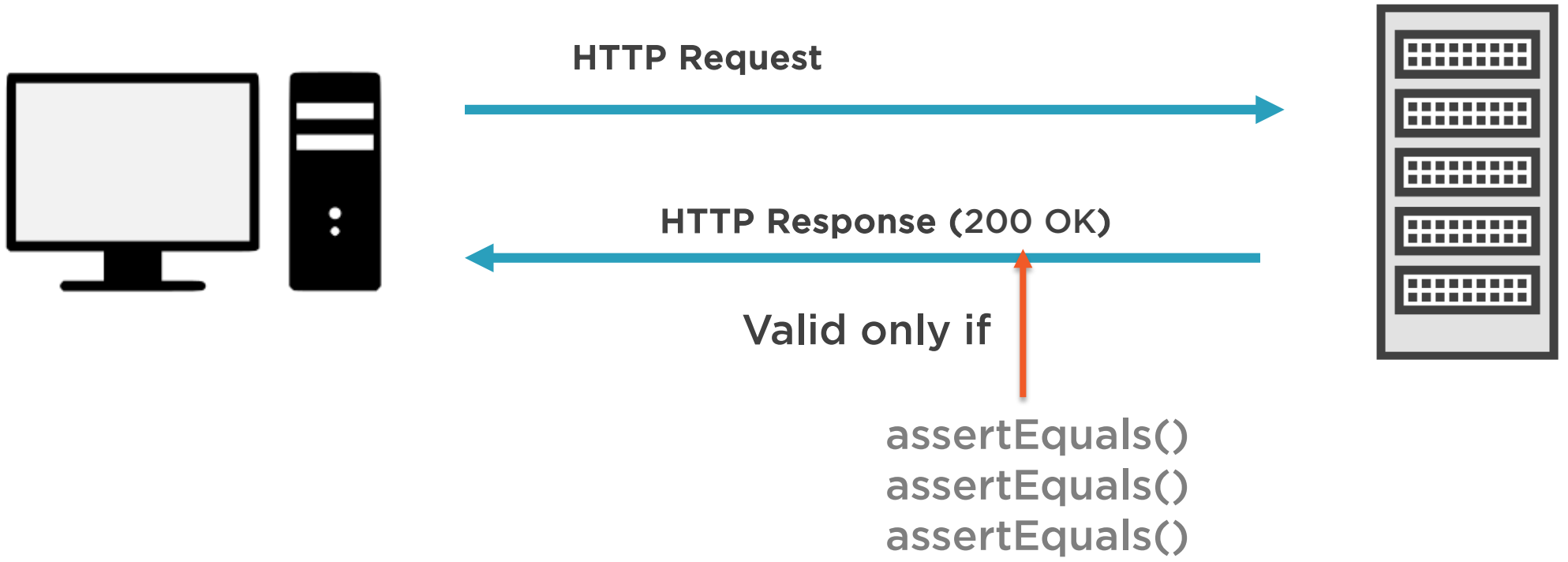

@BeforeMethod
@Test


@BeforeMethod
@Test
@Test
@Test


@BeforeSuite
@BeforeMethod
setup(){}
@BeforeMethod
setup(){}
@Test
@Test
@Test


@BeforeSuite
@BeforeMethod
globalSetup(){}
@BeforeMethod
localSetup(){}
@Test
@Test
@Test

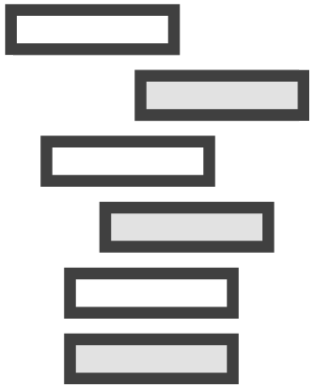
throw new SkipExcetion(“...”);


HTTP Request
HTTP Response (200 OK)
Valid only if
assertEquals()
assertEquals()
assertEquals()

SkipException Cases

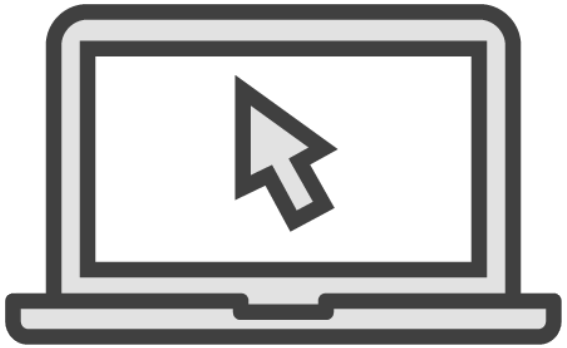
Unit

Yes



Component & Integration

Yes



UI

Yes

```
@Before____
public void setup(){

        if(!condition){ throw new SkipExcetion("..."); }
}
@Test
public void verifyX(){
        Assert.assertEquals(v1,v2);
        new SoftAssert().assertEquals(v1,v2);
}
@After____
public void cleanup(){}
```