

Lab 4.2 - Readable Tests

Introduction

A set of unit tests have already been written for you, but they could be better - in particular, they could be more *readable*.

Starter projects

There is a starter project for this lab, available in C#, Java and JavaScript versions.

Goals

Your job is to analyse the provided unit tests, and think of ways to make the tests more readable.

Apply your improvements to the tests, then make sure they all still pass when run.

Think about...

- Do the names of each test follow a consistent, useful naming convention?
- Is there any repeated code? How could you remove the duplication yet still have the tests function the same as before?
- How many things are being tested in each test? If there is more than one assert statement, is that acceptable?
- Does each test clean up well enough after itself?

Java notes

You can define a method that runs before every test using this code:

```
@Before
public void setUp() {
    // Code goes here...
}
```

You can define a method that runs after every test using this code:

```
@After
public void tearDown() {
    // Code goes here...
}
```

C# notes

You can define a method that runs before every test using this code:

```
[SetUp]
public void SetUp()
{
    // Code goes here...
}
```

You can define a method that runs after every test using this code:

```
[TearDown]
public void TearDown()
{
    // Code goes here...
}
```

JavaScript notes

You can define a method that runs before every test using this code:

```
beforeEach(() => {  
  // Code goes here...  
});
```

You can define a method that runs after every test using this code:

```
afterEach(() => {  
  // Code goes here...  
});
```