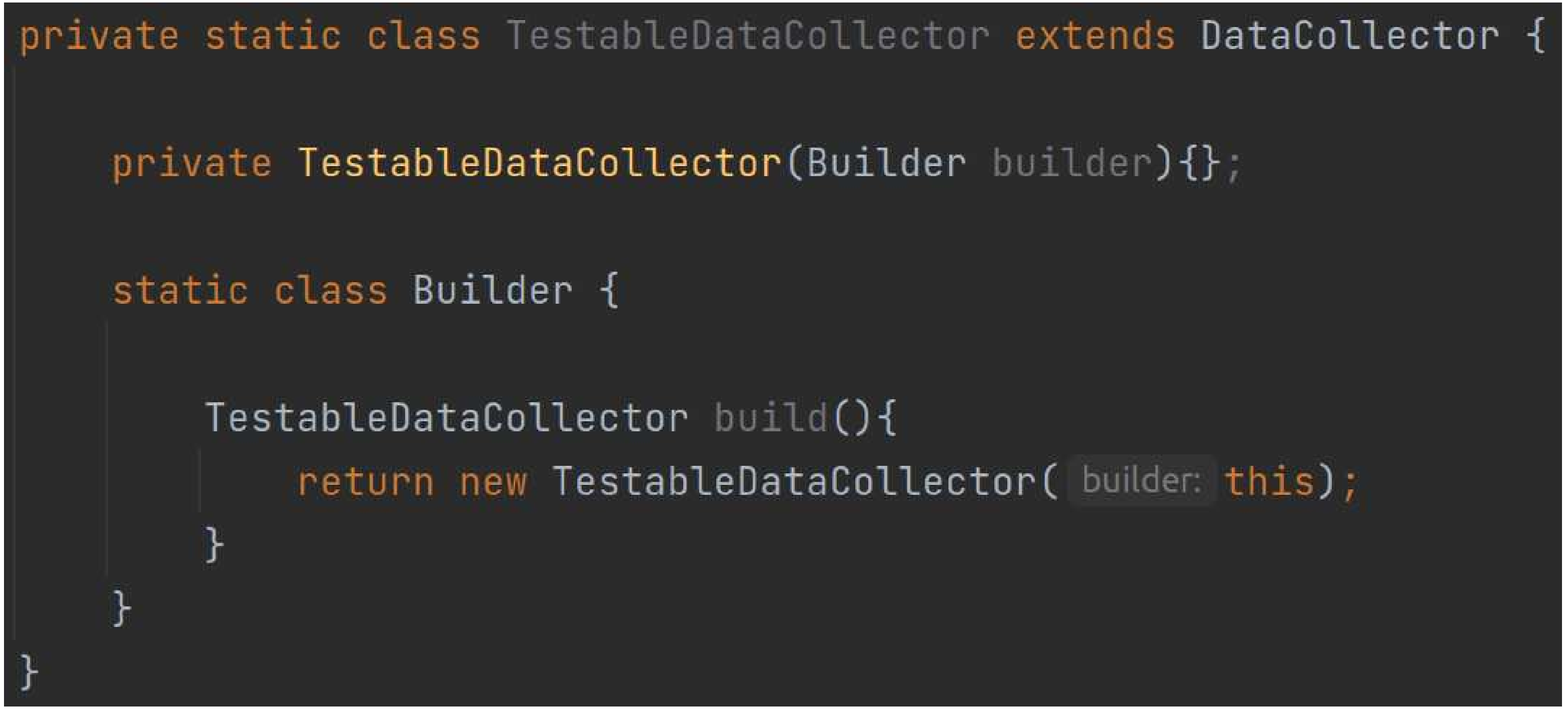
Alcor Code Renovation Course 2021

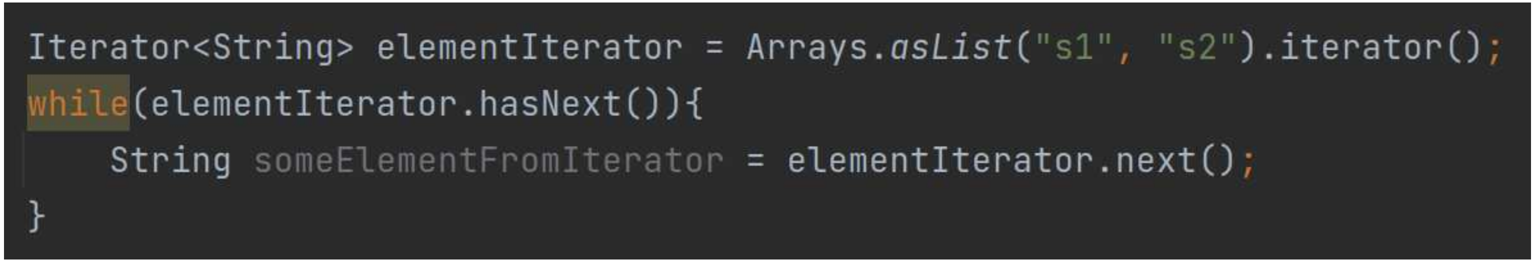
More readable Unit Test using the Builder-Pattern with Seam-Objects

Building a Seam



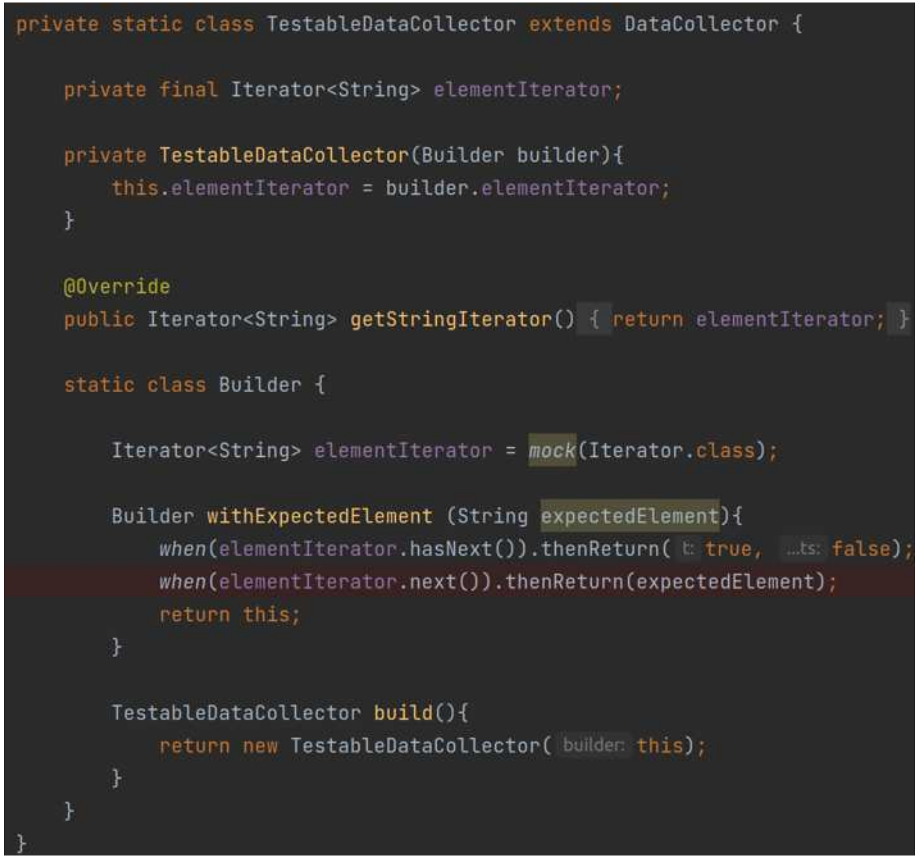
```
private static class TestableDataCollector extends DataCollector {

    private TestableDataCollector(Builder builder){};

    static class Builder {

        TestableDataCollector build(){
            return new TestableDataCollector( builder: this);
        }
    }
}
```

Building a Seam

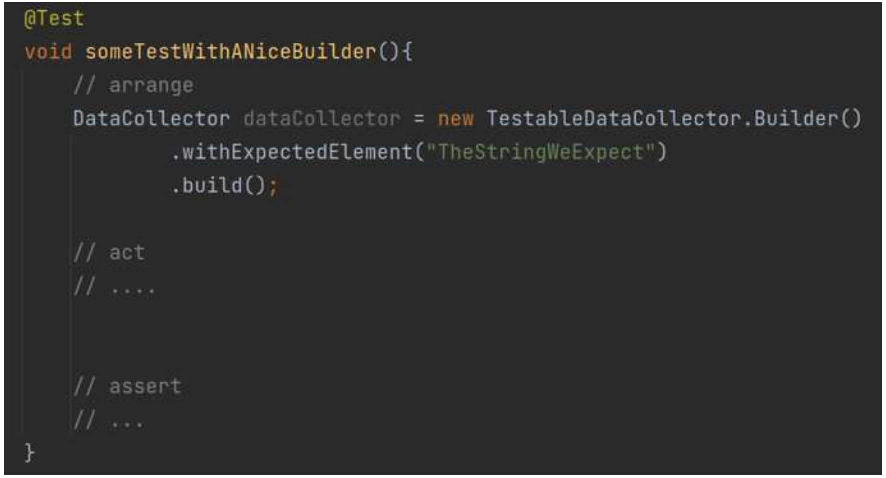


```
Iterator<String> elementIterator = Arrays.asList("s1", "s2").iterator();
while(elementIterator.hasNext()){
    String someElementFromIterator = elementIterator.next();
}
```

Building a Seam



```
private static class TestableDataCollector extends DataCollector {

    private final Iterator<String> elementIterator;

    private TestableDataCollector(Builder builder){
        this.elementIterator = builder.elementIterator;
    }

    @Override
    public Iterator<String> getStringIterator() { return elementIterator; }

    static class Builder {

        Iterator<String> elementIterator = mock(Iterator.class);

        Builder withExpectedElement (String expectedElement){
            when(elementIterator.hasNext()).thenReturn( t: true,  ...ts: false);
            when(elementIterator.next()).thenReturn(expectedElement);
            return this;
        }

        TestableDataCollector build(){
            return new TestableDataCollector( builder: this);
        }
    }
}
```



```
@Test
void someTestWithANiceBuilder(){
    // arrange
    DataCollector dataCollector = new TestableDataCollector.Builder()
            .withExpectedElement("TheStringWeExpect")
            .build();

    // act
    // ....


    // assert
    // ...
}
```

When ROI?

- Testing complex Classes with a Mix of Mocks, Dummys and Overrides
- Testing Classes that require a lot of mocking
- Messy arrange part

Small Example from the Course

- https://github.com/Alcor-Academy/RefactoringCalisthenicsDependencyBreaking/blob/artem/Java/src/test/java/AdvancedTechniques/SchedulerShould.java