BDD + Gherkin + Cucumber in action

Tools

Java (8+ recomendado)

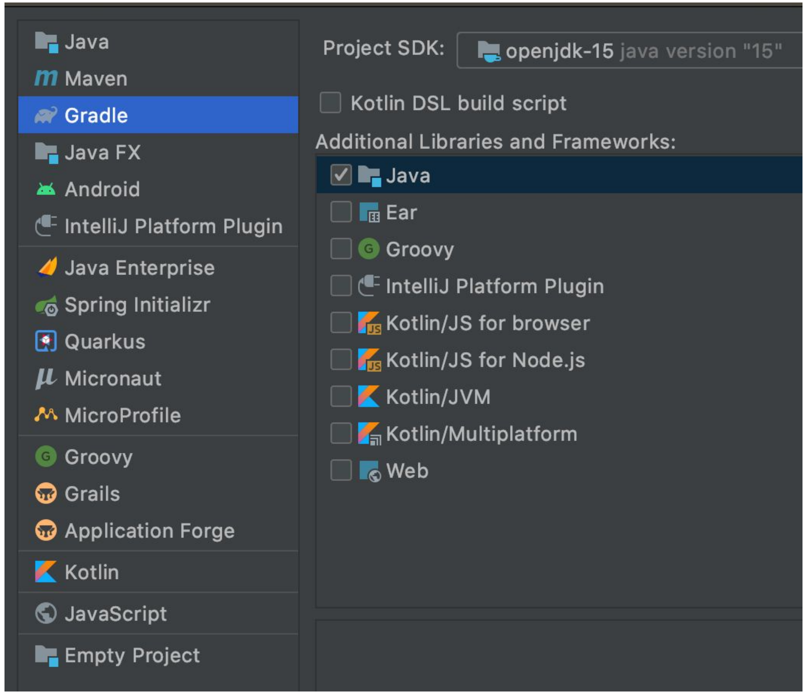
Intellij

Gradle (dependency manager)

Cucumber JVM (info.cukes dependency)

JUnit

Creating a Gradle Project


Java
Maven
Gradle
Java FX
Android
IntelliJ Platform Plugin
Java Enterprise
Spring Initializr
Quarkus
Micronaut
MicroProfile
Groovy
Grails
Application Forge
Kotlin
JavaScript
Empty Project

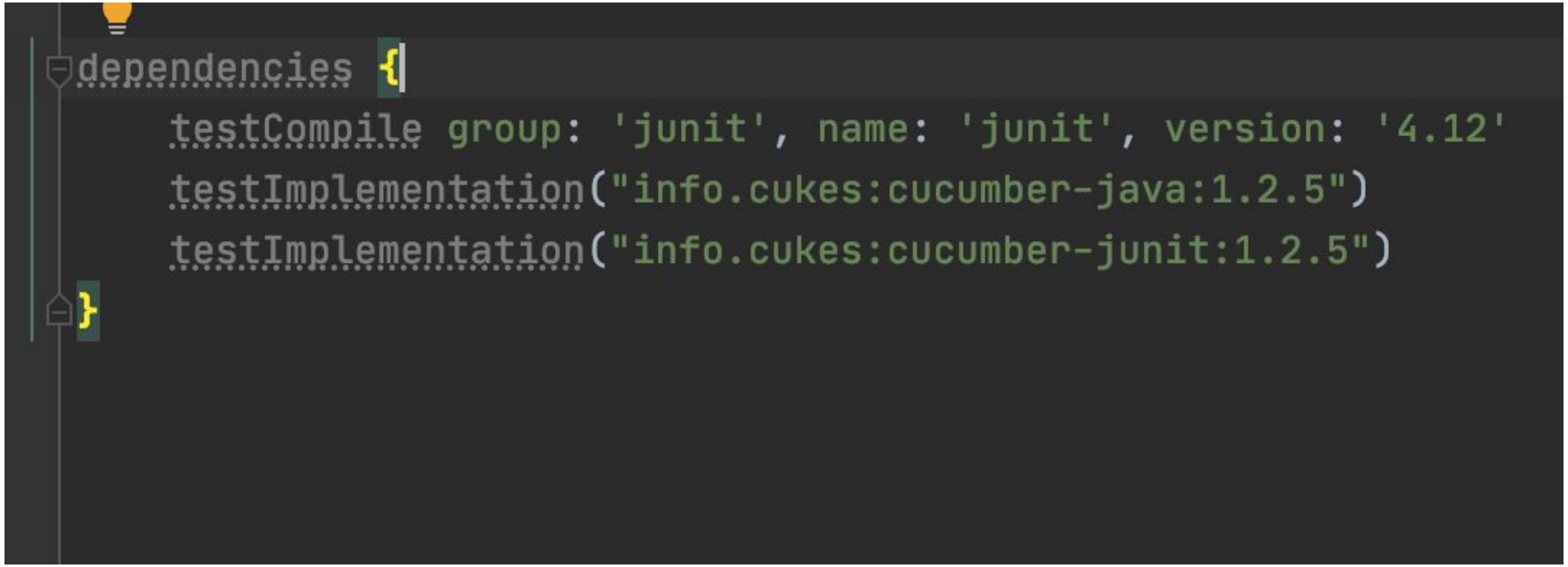
Project SDK: openjdk-15 java version "15"

☐ Kotlin DSL build script

Additional Libraries and Frameworks:

☑ Java
☐ Ear
☐ Groovy
☐ IntelliJ Platform Plugin
☐ Kotlin/JS for browser
☐ Kotlin/JS for Node.js
☐ Kotlin/JVM
☐ Kotlin/Multiplatform
☐ Web

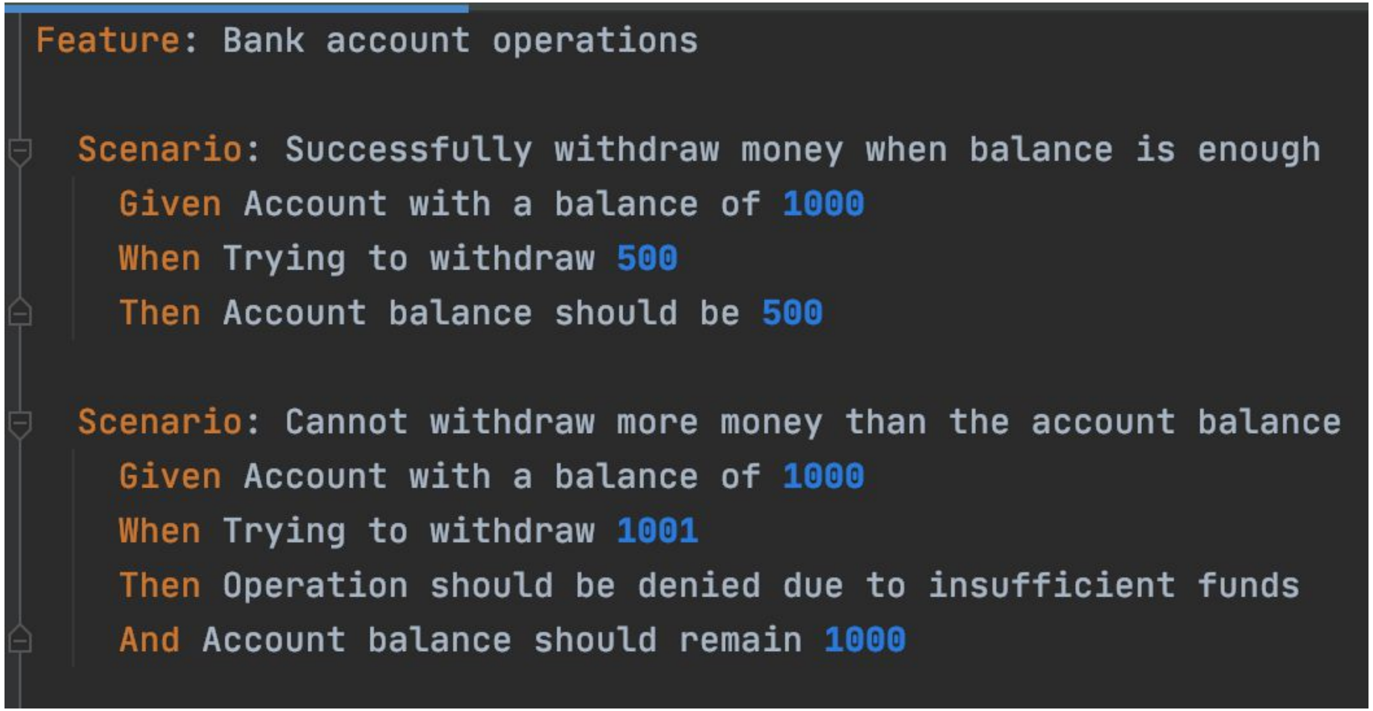
Adding Cucumber dependencies



```
dependencies {
    testCompile group: 'junit', name: 'junit', version: '4.12'
    testImplementation("info.cukes:cucumber-java:1.2.5")
    testImplementation("info.cukes:cucumber-junit:1.2.5")
}
```

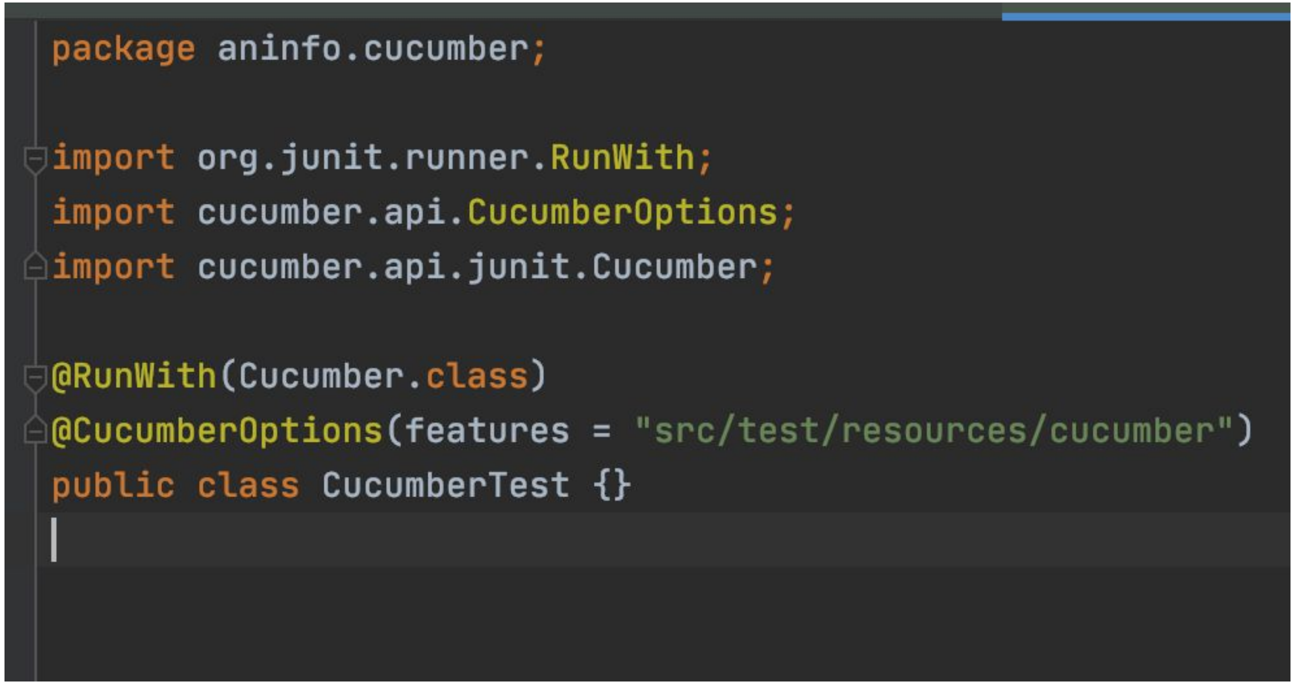
https://mvnrepository.com/artifact/info.cukes/cucumber-junit/1.2.5
https://mvnrepository.com/artifact/info.cukes/cucumber-java/1.2.5

Writing Gherkin Tests



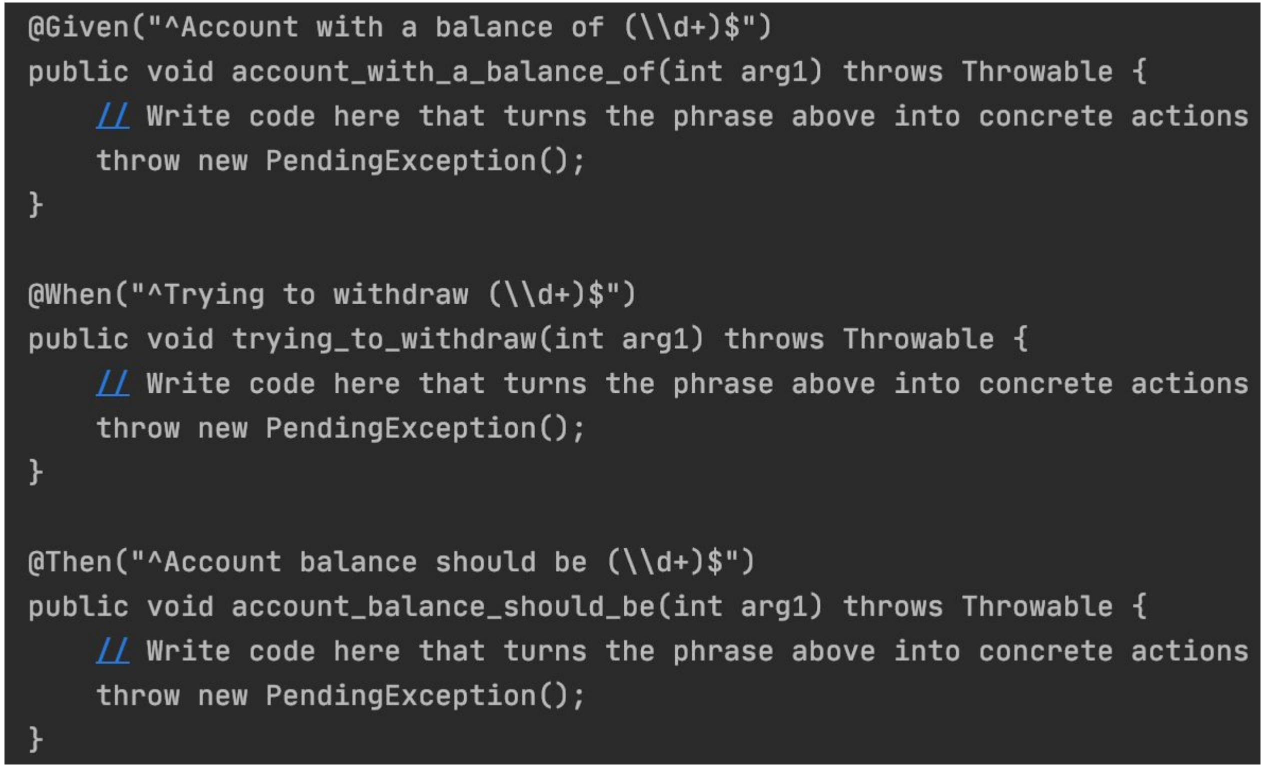
```
Feature: Bank account operations

  Scenario: Successfully withdraw money when balance is enough
    Given Account with a balance of 1000
    When Trying to withdraw 500
    Then Account balance should be 500

  Scenario: Cannot withdraw more money than the account balance
    Given Account with a balance of 1000
    When Trying to withdraw 1001
    Then Operation should be denied due to insufficient funds
    And Account balance should remain 1000
```

Creating Cucumber runner

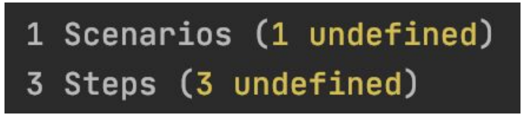


```java
package aninfo.cucumber;

import org.junit.runner.RunWith;
import cucumber.api.CucumberOptions;
import cucumber.api.junit.Cucumber;

@RunWith(Cucumber.class)
@CucumberOptions(features = "src/test/resources/cucumber")
public class CucumberTest {}
```

RUNNING TESTS (NO CODE AT ALL)



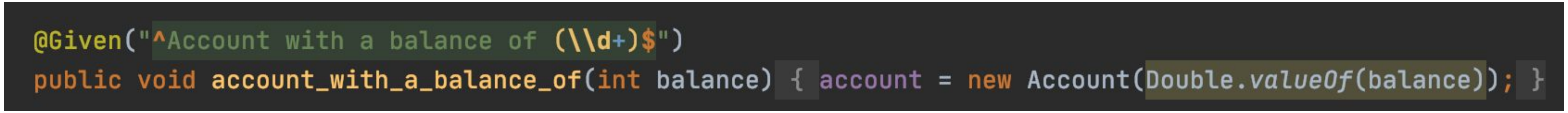
```
@Given("^Account with a balance of (\\d+)$")
public void account_with_a_balance_of(int arg1) throws Throwable {
    // Write code here that turns the phrase above into concrete actions
    throw new PendingException();
}

@When("^Trying to withdraw (\\d+)$")
public void trying_to_withdraw(int arg1) throws Throwable {
    // Write code here that turns the phrase above into concrete actions
    throw new PendingException();
}

@Then("^Account balance should be (\\d+)$")
public void account_balance_should_be(int arg1) throws Throwable {
    // Write code here that turns the phrase above into concrete actions
    throw new PendingException();
}
```



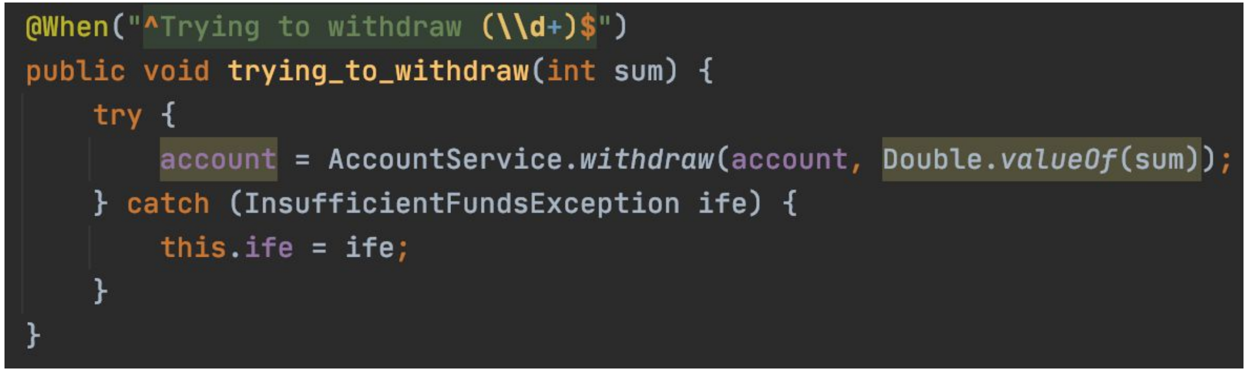
```
1 Scenarios (1 undefined)
3 Steps (3 undefined)
```

Nos dice como codearlo!

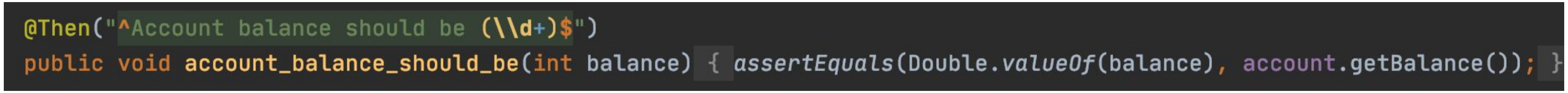
Coding the tests



```java
@Given("^Account with a balance of (\\d+)$")
public void account_with_a_balance_of(int balance) { account = new Account(Double.valueOf(balance)); }
```

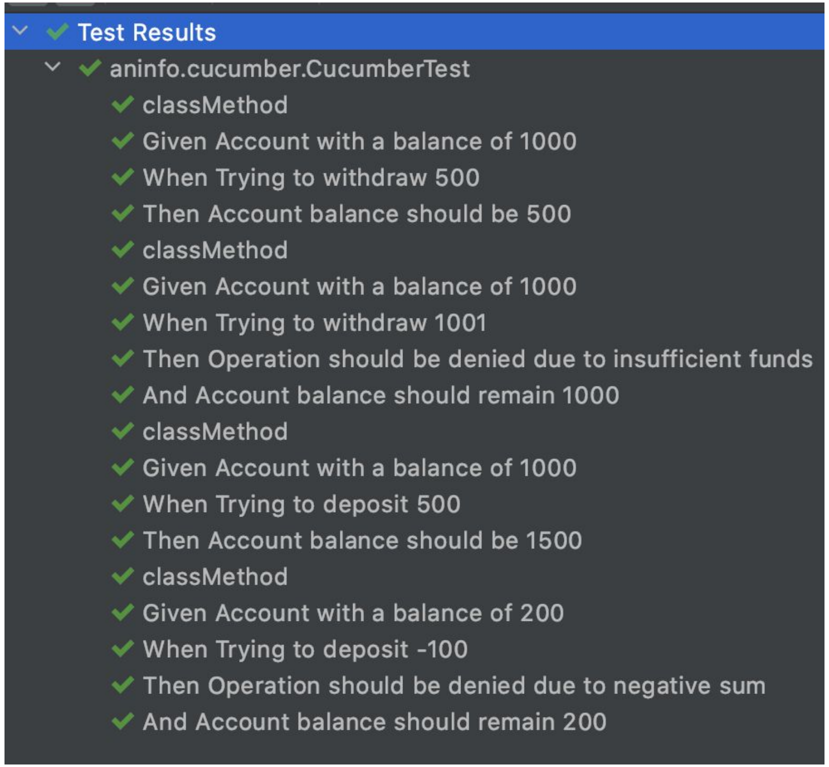


```java
@When("^Trying to withdraw (\\d+)$")
public void trying_to_withdraw(int sum) {
    try {
        account = AccountService.withdraw(account, Double.valueOf(sum));
    } catch (InsufficientFundsException ife) {
        this.ife = ife;
    }
}
```

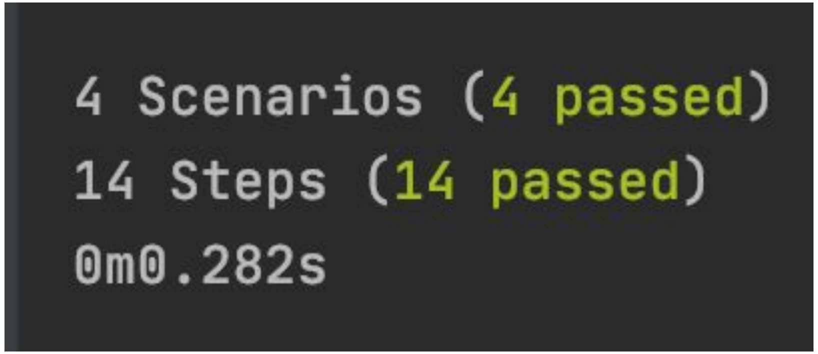


```java
@Then("^Account balance should be (\\d+)$")
public void account_balance_should_be(int balance) { assertEquals(Double.valueOf(balance), account.getBalance()); }
```

TESTS IN GREEN!



```
✓ Test Results
  ✓ aninfo.cucumber.CucumberTest
    ✓ classMethod
    ✓ Given Account with a balance of 1000
    ✓ When Trying to withdraw 500
    ✓ Then Account balance should be 500
    ✓ classMethod
    ✓ Given Account with a balance of 1000
    ✓ When Trying to withdraw 1001
    ✓ Then Operation should be denied due to insufficient funds
    ✓ And Account balance should remain 1000
    ✓ classMethod
    ✓ Given Account with a balance of 1000
    ✓ When Trying to deposit 500
    ✓ Then Account balance should be 1500
    ✓ classMethod
    ✓ Given Account with a balance of 200
    ✓ When Trying to deposit -100
    ✓ Then Operation should be denied due to negative sum
    ✓ And Account balance should remain 200
```



```
4 Scenarios (4 passed)
14 Steps (14 passed)
0m0.282s
```

PAPERS AND DOCS

https://martinfowler.com/bliki/GivenWhenThen.html

https://martinfowler.com/bliki/SpecificationByExample.html

https://dannorth.net/introducing-bdd/

GRACIAS!

nmouteda@gmail.com

nouteda@fi.uba.ar