4-bdd-test-using-cucumber

Steping with Scientist

Scientist Steps on Field

Tests **Scientist**.**Move**() method. Given there is two fields and one of them contains the user's scientist when the user moves its scientist then the scientist's field should be changed. Alias, we compare the scientist.getField() method's results: one before and one after Move().

Uses **assertNotSame** as we check if the two field as objects are the same or not. If it's not then good, the user really moved its scientist so the test should pass.

Scientist Unable to Move

Tests **Scientist**.**Move**() method also however in this scenario the scientist has been affected by the **Stun** Virus. In this case the scientist should not be able to move its position during Move().

Uses **assertSame** as we check if the two fiels as objects are the same or not. If it's the same then good, the user really can't move its scientist – because of the virus – so the test should pass.

Scientist Steps to a Laboratory in order to Learn a Genetic Code

Tests **Scientist**.**Move**() along with Scientist.**Touch**() and Scientist.**Learn**(). The test put one **Genetic Code** on the **Laboratory**. The scientist moves there, starts touching and finally learns what it touched there.

Uses **assertTrue** as we check if the scientist's inventory contains the touched item. In order to pass it should be true.

Crafting with Scientist

The feature file only has one Scenario written but as Scenario Outline. Scenario Outline uses Examples keyword to avoid writing same scenarios with different data. This means the number of rows in Examples are equivalent how many times the Scenario Outline read which tells us how many „tests" in this feature file we have.

Crafting during Dementia

The first case is when Scientist tries to craft but it has active agent **Dementia**. Dementia makes **Scientist** forget every genetic code which in our case means Scientist should not be able to craft anything.

With **assertTrue** we check if the scientist’s inventory called crafted is empty, if it’s empty then the test should pass.

Crafting during Stun

The second case is when **Scientist** tries to craft but it has active agent **Stun**. Stun stops Scientist before it could craft anything.

With **assertTrue** we check if the scientist’s inventory called crafted is empty, if it’s empty then the test should pass.

Scientist versus Scientist

In this feature file (infect.feature) we manages multiple scientists and fields at once. The file contains two Scenarios.

Scientist infects

One of the **Scientist** tries to infect the other Scientist with **Bear**. If one scientist moves to a field where there is another scientist with Bear infection then it should be infected too. Scientist stores active agents in their **Inventory**.

With **assertTrue** we check if the scientist’s inventory called agents is not empty, if it’s not empty then the test should pass since it means it got infected and stored the agent in its inventory.

Scientist kills

One of the **Scientist** tries to kill a **Bear** infected Scientist with **Gear** like **Axe**.

With **assertTrue** we check if the scientist has been killed, it it’s killed then the test should pass.