Exercise 5: Testing Rules

**New requirements:**

In previous exercises, rule model testing was only introduced briefly. Now, the rule model should be tested carefully. A rule test with one test case for flow rule Rental Calculation is already existing.

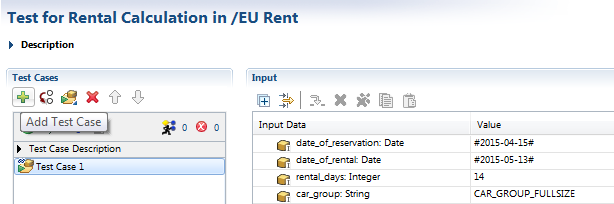
Define more test cases with different input parameters and expected values and execute the rule tests, again.

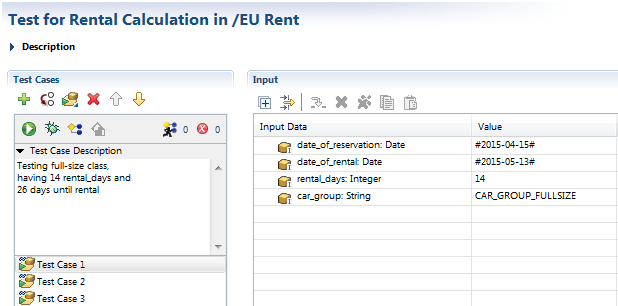
After having executed the rule tests successfully, e.g. change the compact classes’ price from 50 € to 55 € within the rule and test again. Compare and analyze the results.

**Steps:**

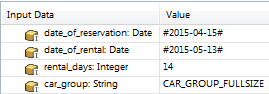
1. Rename the already existing test case with a meaningful name. Define more test cases with different input parameters for the rule test of Rental Calculation. As well, set the expected output values.

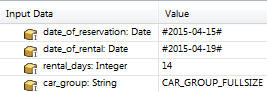
**Tipp**: Add adequate “Test Case Descriptions” for a better understanding of each test case.

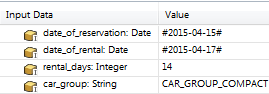




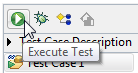
Possible input values are:

Test Case 1: 

Test Case 2: 

Test Case 3: 

1. Save the test cases and execute them.

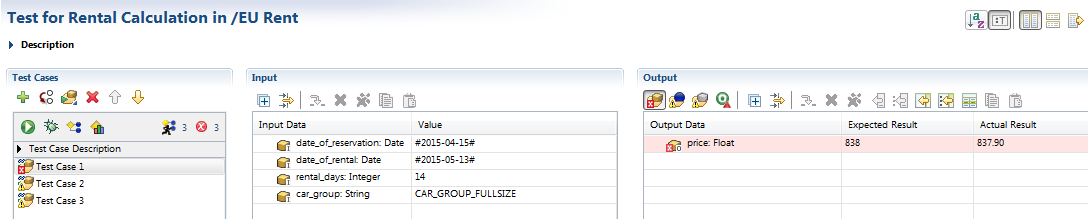


1. Now, the statistics for every test case of passed through rules can be displayed and analyzed, as well. Therefore, enable “Statistics”.

**Hint**: The “Statistics” icon is shown in the menu bar when switching from the rule test back to rule Rental Calculation.

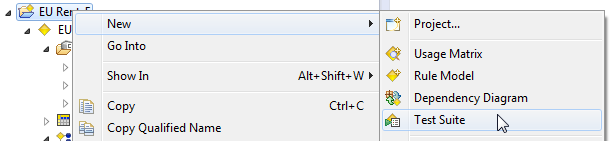


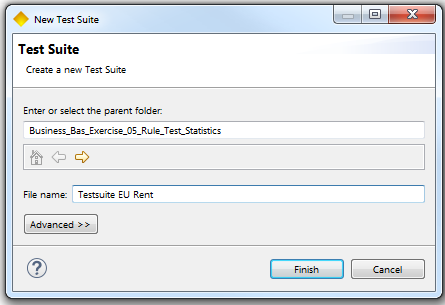
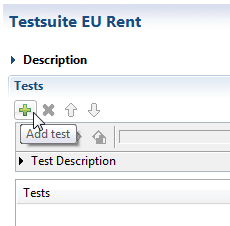
1. Change the rule as mentioned in the requirements, save changes and start the rule test all over in order to see the impact.
2. After having executed the test cases, the test results have to be analyzed, again: there should be red markings, where the compact class has been tested. Adjust the expected result in order to get a positive result when resuming the test cases.



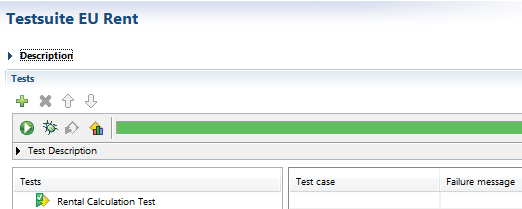
Optionally:

1. Create new test cases for all other possible input data combinations.
2. Create a test suite Testsuite EU Rent and add the rule test of Rental Calculation.

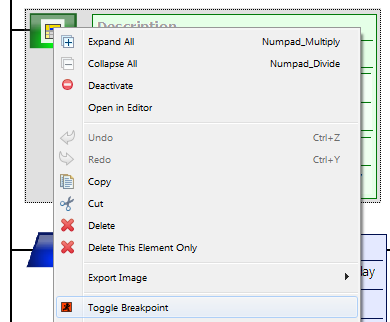
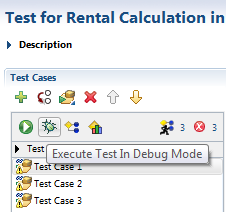


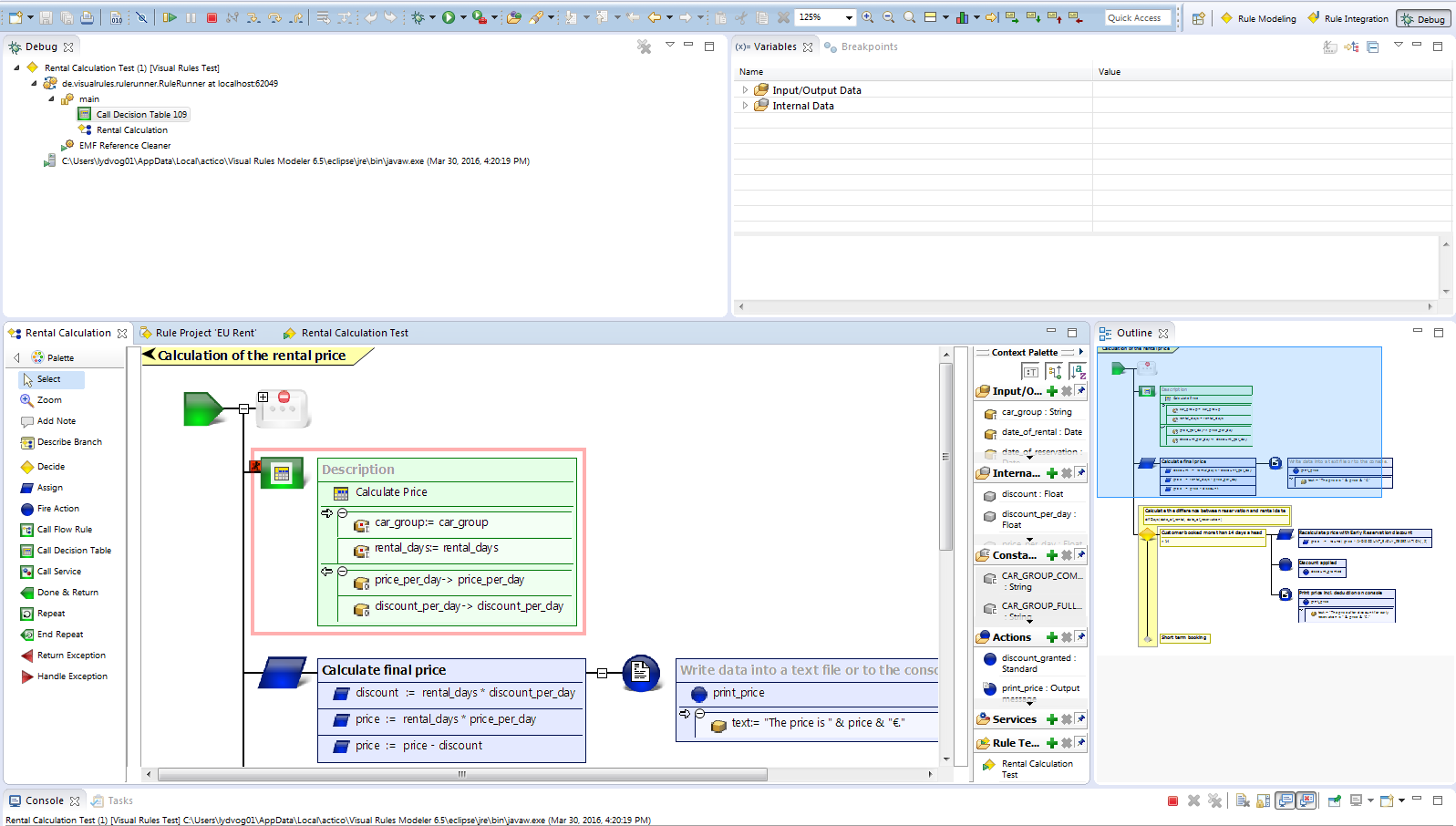
1. Execute the test suite and check the result. The test suite is marked green in case of successful execution.



1. Randomly within the flow rule, a breakpoint should be added to a rule element. Now, the rule is supposed to be executed in debug – mode: a step – by – step graphical debugging is executed while parallel tracing of the altering data elements is possible.

Possible Debug – View:



Questions:

1. Which test steps can be performed with the test editor?

|  |  |  |
| --- | --- | --- |
| a | Add new test cases |  |
| b | Add test descriptions |  |
| c | Insert expected values for output data elements |  |
| d | Display data results |  |
| e | Display fault tests |  |
| f | Execute rule |  |

1. Briefly describe how to perform testing in Visual Rules.
2. Which different adjustments can be performed within the test editor for creating results?

|  |  |  |
| --- | --- | --- |
| a | Create results for all output data values, internal data values and actions |  |
| b | Create results for all input-/output data elements |  |
| c | Create results for all constant data values |  |
| d | Create no result at all |  |
| e | Only create a result, when at least one action has been performed |  |
| f | Create a result for individual data elements |  |

1. Which of the following states can be accepted for statistics?

|  |  |  |
| --- | --- | --- |
| a | Executed path is marked in grey |  |
| b | Accumulated statistics are possible |  |
| c | Executed path is brought to foreground |  |
| d | Statistics can be displayed or faded - out |  |
| e | Statistics are possible for every single test case |  |

1. Briefly describe how to perform debugging in Visual Rules.

Results:

1. Which test steps can be performed with the test editor?

|  |  |  |
| --- | --- | --- |
| a | Add new test cases | x |
| b | Add test descriptions | x |
| c | Insert expected values for output data elements | x |
| d | Display data results | x |
| e | Display fault tests | x |
| f | Execute rule | x |

1. Briefly describe how to perform testing in Visual Rules.

* Create a new rule test
* Add a test description
* Add different test cases
* Inserting input data values
* Inserting expected results
* Execute rule
* Displaying results
* Checking results in case of appearing errors

1. Which different adjustments can be performed within the test editor for creating results?

|  |  |  |
| --- | --- | --- |
| a | Create results for all output data values, internal data values and actions | x |
| b | Create results for all input-/output data elements |  |
| c | Create results for all constant data values |  |
| d | Create no result at all | x |
| e | Only create a result, when at least one action has been performed |  |
| f | Create a result for individual data elements | x |

1. Which of the following states can be accepted for statistics?

|  |  |  |
| --- | --- | --- |
| a | Executed path is marked in grey |  |
| b | Accumulated statistics are possible | x |
| c | Executed path is brought to foreground | x |
| d | Statistics can be displayed or faded - out | x |
| e | Statistics are possible for every single test case | x |

1. Briefly describe how to perform debugging in Visual Rules.

1. Add a breakpoint to the required rule element within the rule to be executed. Here the execution of the rule will be interrupted

2. Start the flow rule by choosing the debug – symbol within the test editor, via toolbar or by selecting Run > Debug within the menu bar

3. Automatic switching to debug – perspective

4. Step – by – step execution of the referring rule by using the adequate symbols within the tool bar