* Explain the purpose of the Test (what the original test exposed, and what your test expose).

This test simplifies the testing of the sensitivity of the number plate scanner in a very effective way.

* Explain about Parameterized Tests in JUnit and how you have used it in this exercise.

It allows the test of the same method with many different inputs. In this case the method that goes under testing needs a file and there for reading multiple files and checking them against the specific method.

* Explain the topic Data Driven Testing, and why it often makes a lot of sense to read test-data from a file.

Because usually we create a data set while we do the partition and boundary analyses. Therefore we already have a file to test. Also it usually the case that we just receive files with test cases and this way of testing makes it convenient to test any test set delivered in a file.

* Your answers to the question; whether what you implemented was a Unit Test or a JUnit Test, the problems you might have discovered with the test and, your suggestions for ways this could have been fixed.

It is a unit testing as we test a specific unit of the project. ???

* The steps you took to include Hamcrest matchers in the project, and the difference they made for the test.

Just included the dependency in the pom.xml file. That added more methods that better report that went wrong in the test if there is something wrong...