



## Purpose

Purpose of this task is to introduce modification of the definition of validation rule.

## Description

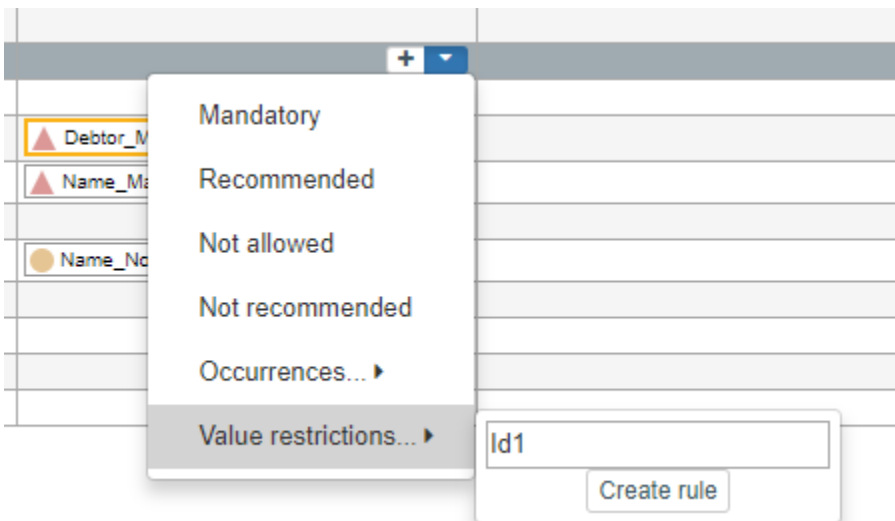
Following steps are done here:

1. Creating a new validation rule(Tx/Id must be value "Id1")
2. Modifying OCL in an existing validation rule(Tx/Id must be value "Id1" or "Id2")
3. Compiling project and testing

## Instructions

### 1: Creating a new validation rule (Tx/Id must be value "Id1")

We start by using the dropdown menu to generate OCL code automatically, as done in basics section. We'll have to find the relevant element (Transaction/Id), press the down-icon, select value restriction and write the value we want to limit this element to be ("Id1").



### 2: Modifying OCL in an existing validation rule(Tx/Id must be value "Id1" or "Id2")

This phase requires us to look at the generated OCL code for the rule we have just made. We'll open the rule by double clicking it. Definition section will contain following:

**Context:** TransactionType1

**Definition:** self.Id.matches("Id1")

OCL allows us to use following logical operators: and, or, xor, not(), implies

In this case, as we want to make either value Id1 or Id2 valid to be used, we'll have to use operator or between the valid values. We can use the existing definition as our basis for completing the OCL to be as follows:

```
self.Id.matches("Id1") or self.Id.matches("Id2")
```

**Additional note**

In this case, if we initially already know that two or more values are accepted, instead of adding one value in phase 1 and then modifying it in phase 2, the quickest way would have been to use regular expression to state all the values at once. Pipe character | is used as "or" in regular expression, so we could have added value

Id1|Id2

in phase 1. Therefore no modification would have been needed in the above example. Modifications are usually relevant in more complex rules

**3: Compiling project and testing**

Finally, we can test the implementation. Our test should include both values "Id1" and "Id2" to be in the XML file.

During the test, we should be able to take notice that the error message is invalid, as it only refers to "Id1" even though the rule itself allows both values.

**4: Modifying error message**

To make rule to be valid, we shall modify error message to match the rule definition. It is important to keep in mind that error messages are generated at the point when rule is generated, but error messages are not automatically changed when changes are made within rule.