# ONE - Workshop

## DQ Aggregation Rules

Prepared for: v15.4

Prepared by: Ataccama

Dated: November 2024

**Contents of the Document**

# Introduction

Whenever a DQ check logic requires grouping and/or aggregation functions, you are required to create DQ Aggregation rules.

This can be achieved in three possible ways:

- Grouping of the data (using **Group by** section) and directly using it in the Rule Logic - **Condition Builder**.
- Grouping of the data (using **Group by** section) followed by the use of aggregate functions in Rule Logic - **Advanced Expressions**.
- Direct use of aggregate functions in the Rule Logic - **Condition Builder**/ **Advanced Expressions**.

In this workshop, we will demonstrate aggregation rules in the following two examples:

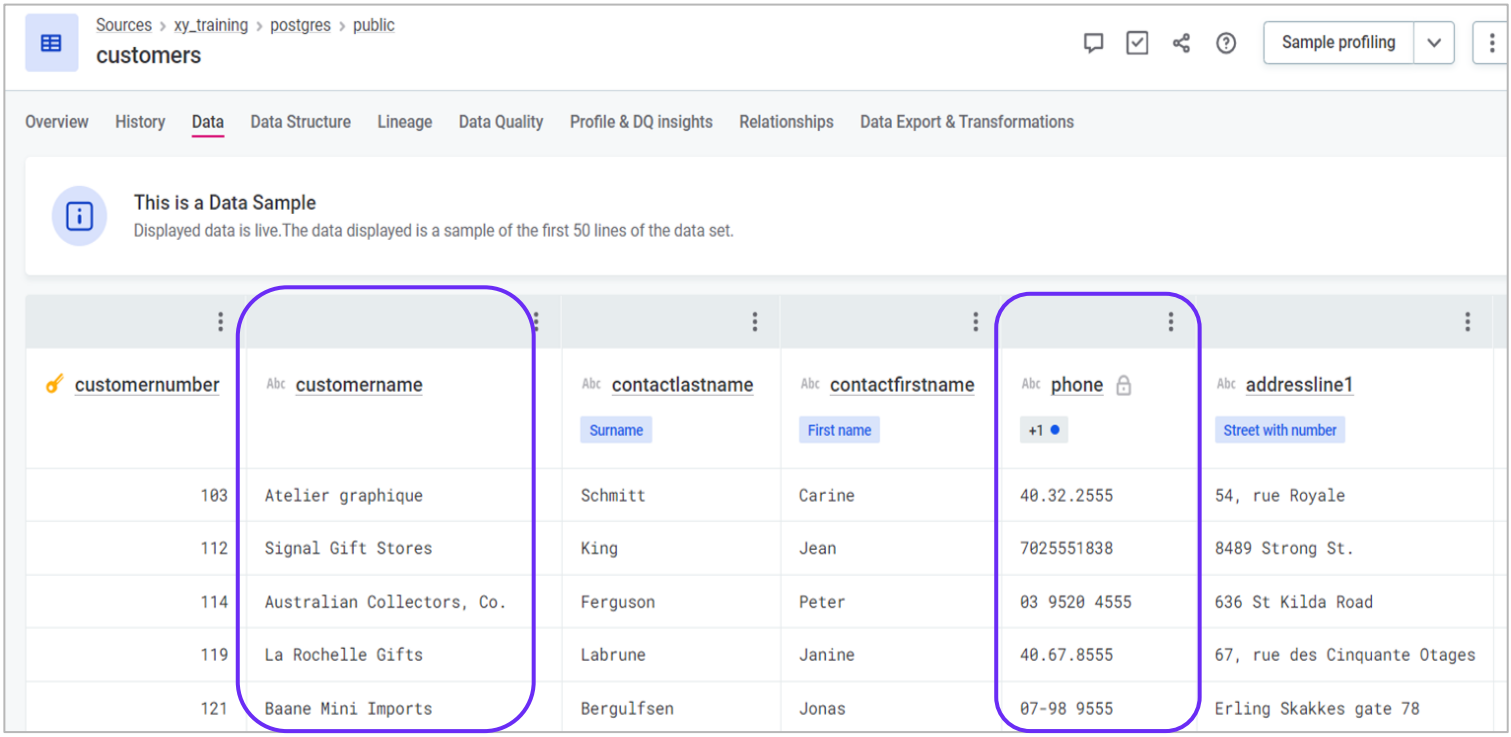
Check whether a combination of two attributes is unique within the whole dataset.

Calculate an average value of a group and see which values are outside its scope with the threshold of 10%.

## 1 – Create a DQ aggregation rule to check the uniqueness of a group of values

Firstly, we will see how an aggregation rule can combine several values together and evaluate them against the dataset. Values from a combination of '**customername**' and '**phone**' columns will be used for this example.

- Navigate to the **Knowledge Catalog** and review the **customers** catalog item.
- Look at the results for the column's **customername** and **phone**. In the **customers** table, we are supposed to have one record per customer. Therefore, we want to evaluate the uniqueness of the combination of these two across the dataset.




Let's start with creating a new rule and defining its logic:

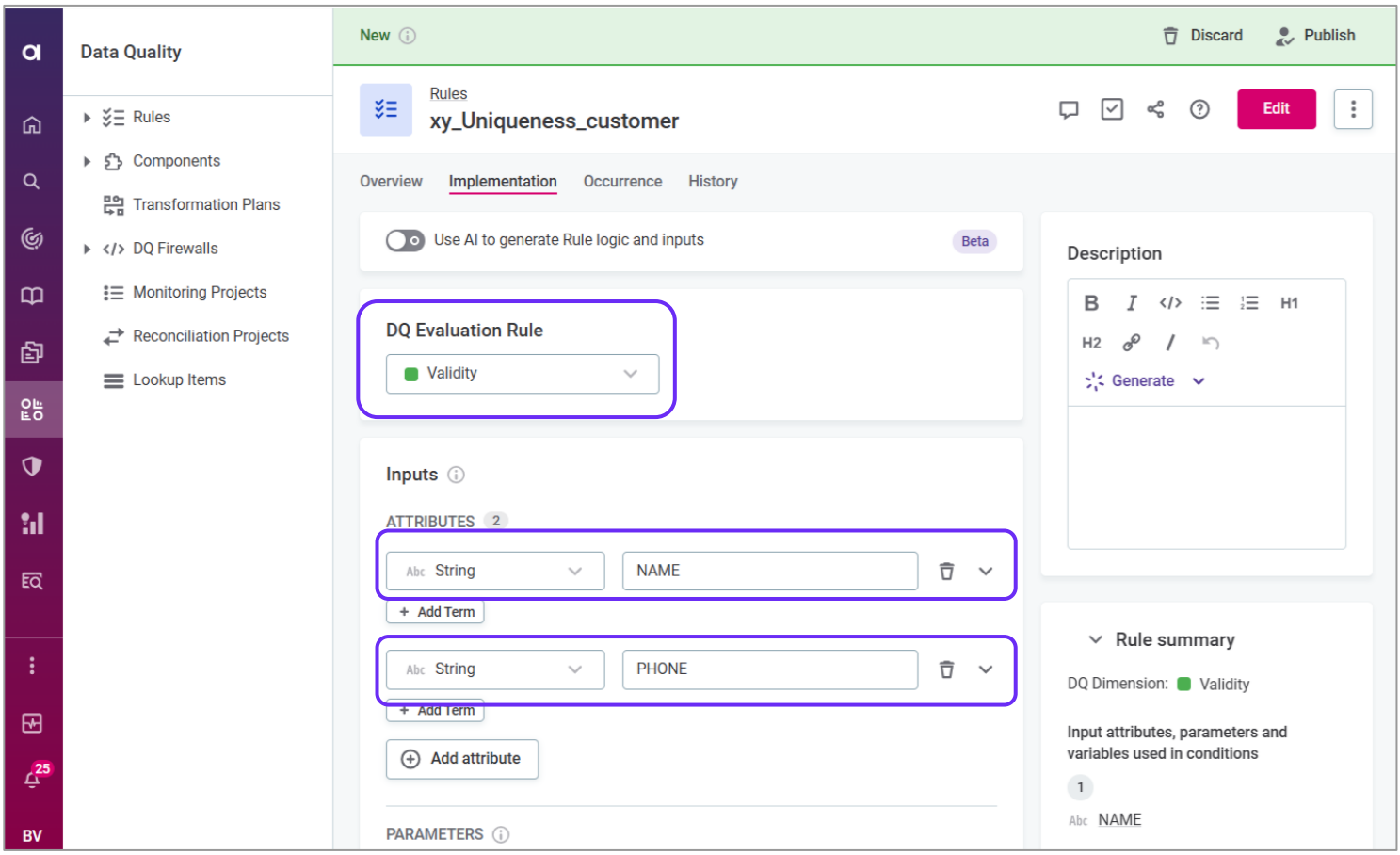
- Navigate to the **Data Quality** → **Rules** section.
- Create a new rule and provide a **Name**, for example: **`<prefix>_Uniqueness_customer`**.
- Select the **Rule Type** as **DQ Evaluation: Validity**.
- (Optional) Add a **Description** if needed.
- Save your changes. You will then be redirected to the **Implementation** tab.

Once the rule is created, we need to specify the rule's logic definition:

- In the **Implementation** tab, go to the Input Attributes section.

› Define the following two input attributes:

- **NAME** of type **STRING**.
- **PHONE** of type **STRING**.




› Right next to it, select the **Aggregation rule** option from the drop-down menu (it will become available after selecting this rule type).

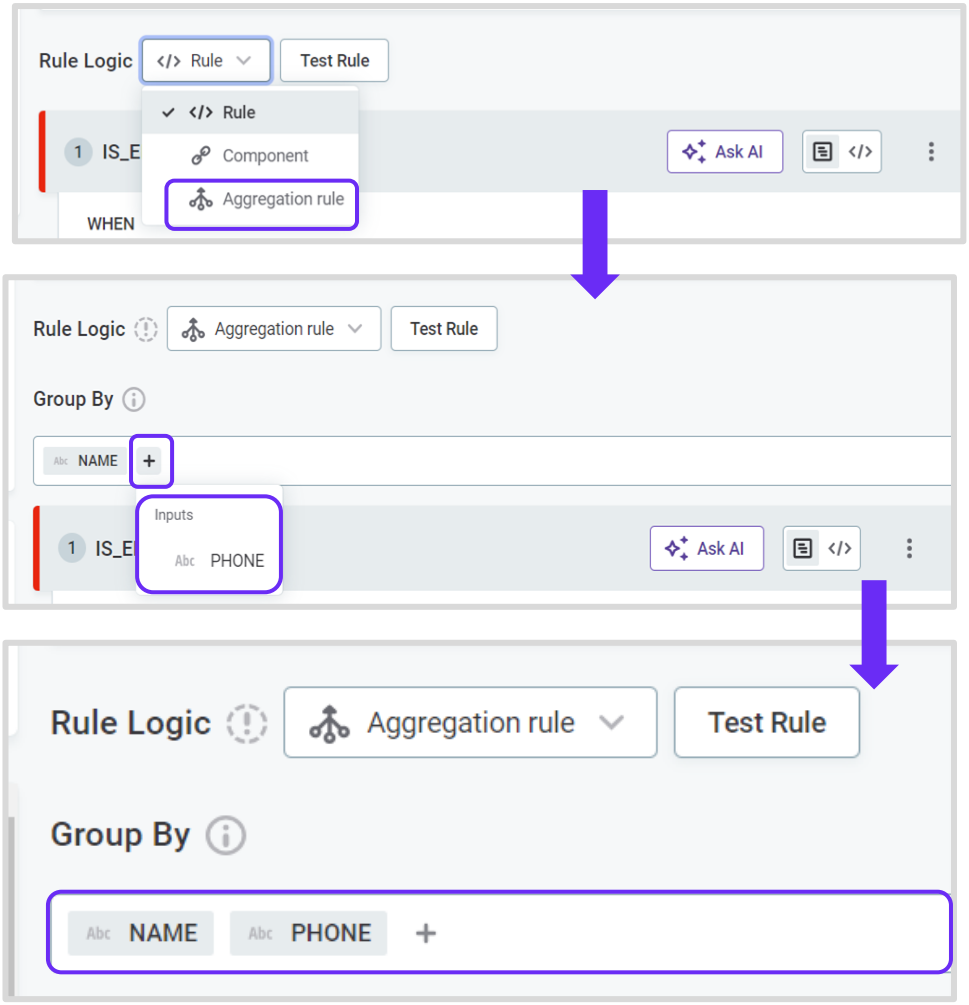
A new section named **Group by** will be made visible.

Using the method that follows, we can use the **Group by** to group a single selected attribute or several attributes. These groups will be available as input in the Rule **Logic** section, under the **Group by results** provided.

The following instructions will show how to form a group of input values and use it for further evaluation:

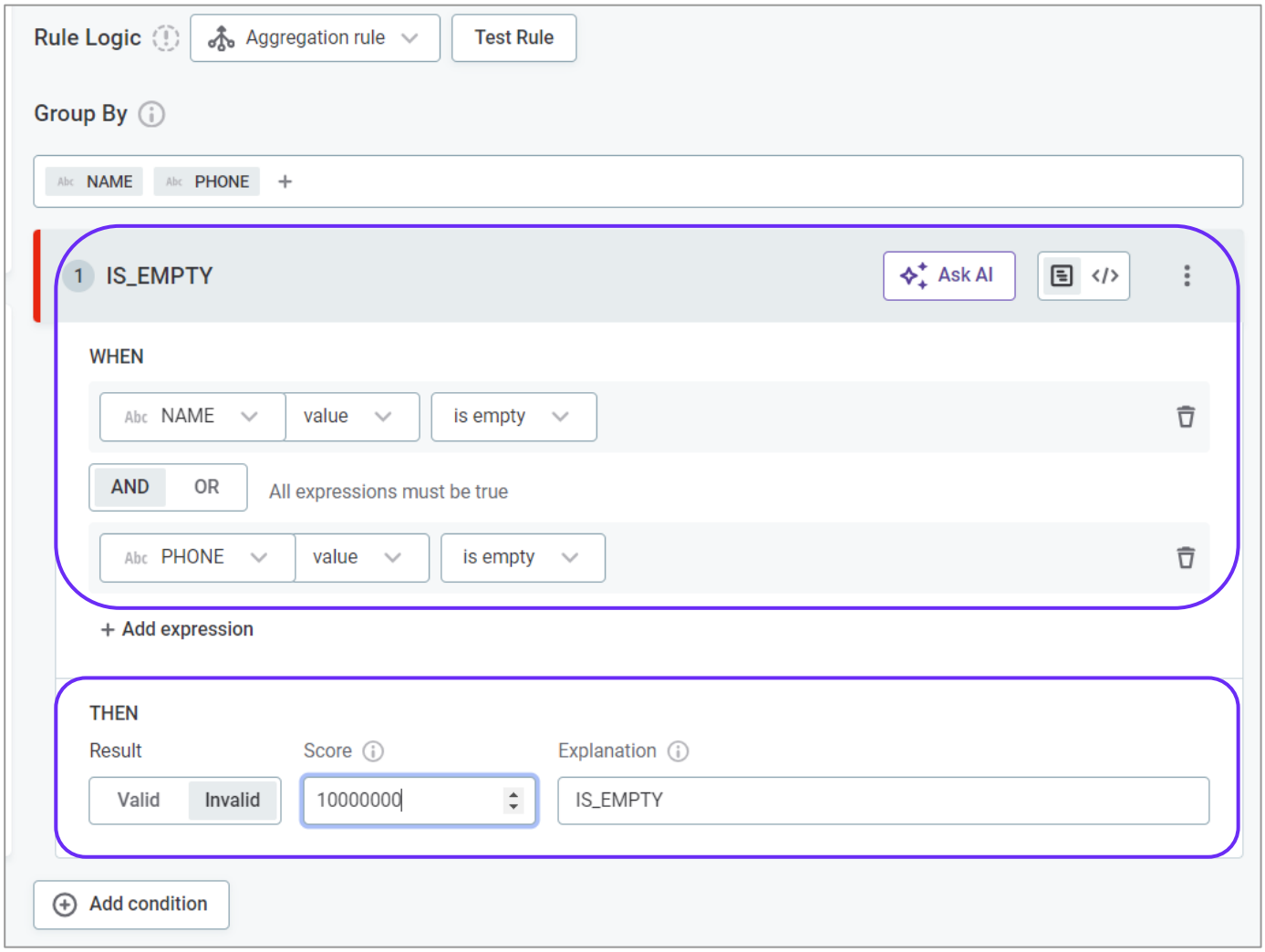
## 1a - Creating a Group

› Group by the two defined input attributes.

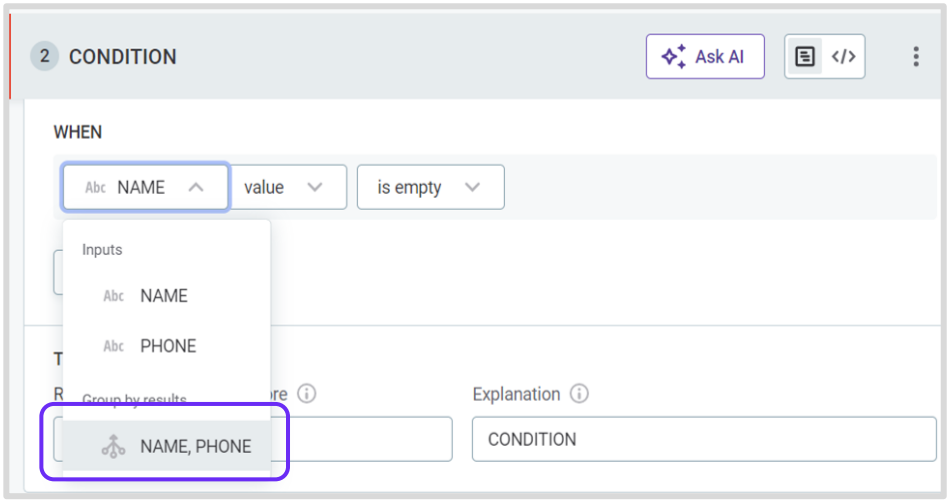


### 1b – Developing Rule Logic

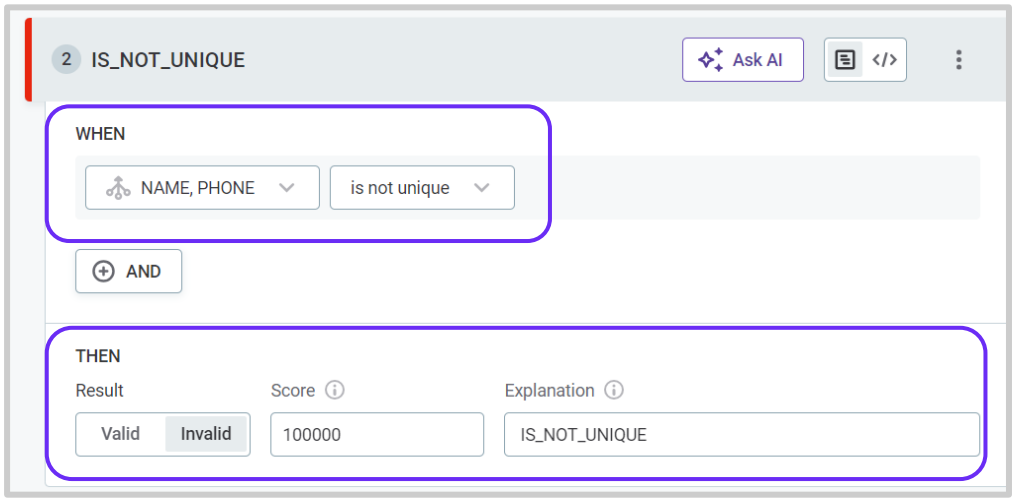
› Select the Condition Builder and check the emptiness of the grouped attributes first (as a best practice). In the WHEN section, select each of these attributes from the list of available inputs and retain the default settings of the WHEN section which is "is empty". Then, use the **'+ AND'** button to add the same condition for the other inputs. Don't forget the explanation code!



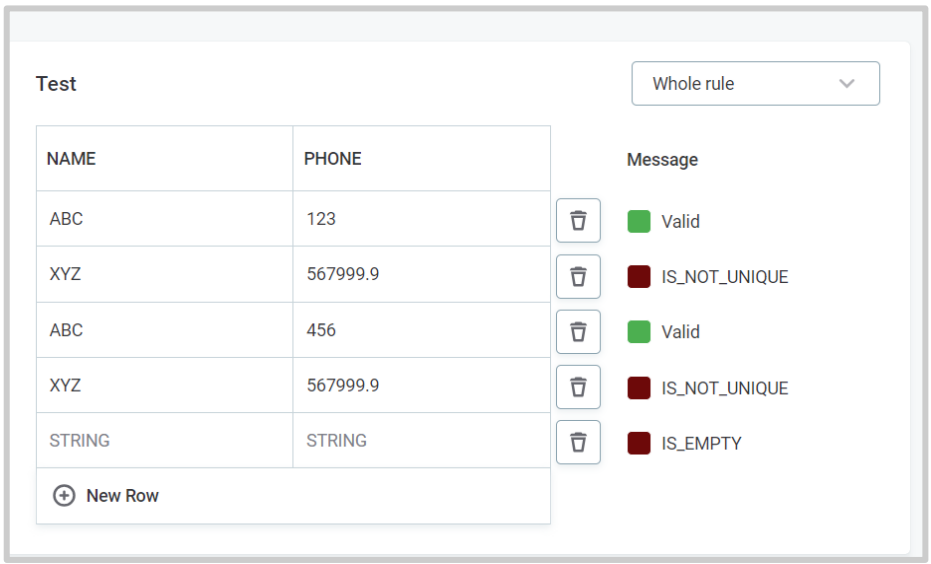

Now, use the **'+ Add condition'** button and select the **Group by results** which is **NAME**, **PHONE** from the list of available inputs.




- Complete the WHEN configuration by changing the already existing condition to **'IS_NOT_UNIQUE'**, score: **100000**
- Make sure correct values are filled in the **Explanation Code** for INVALID and VALID statements in the THEN section.



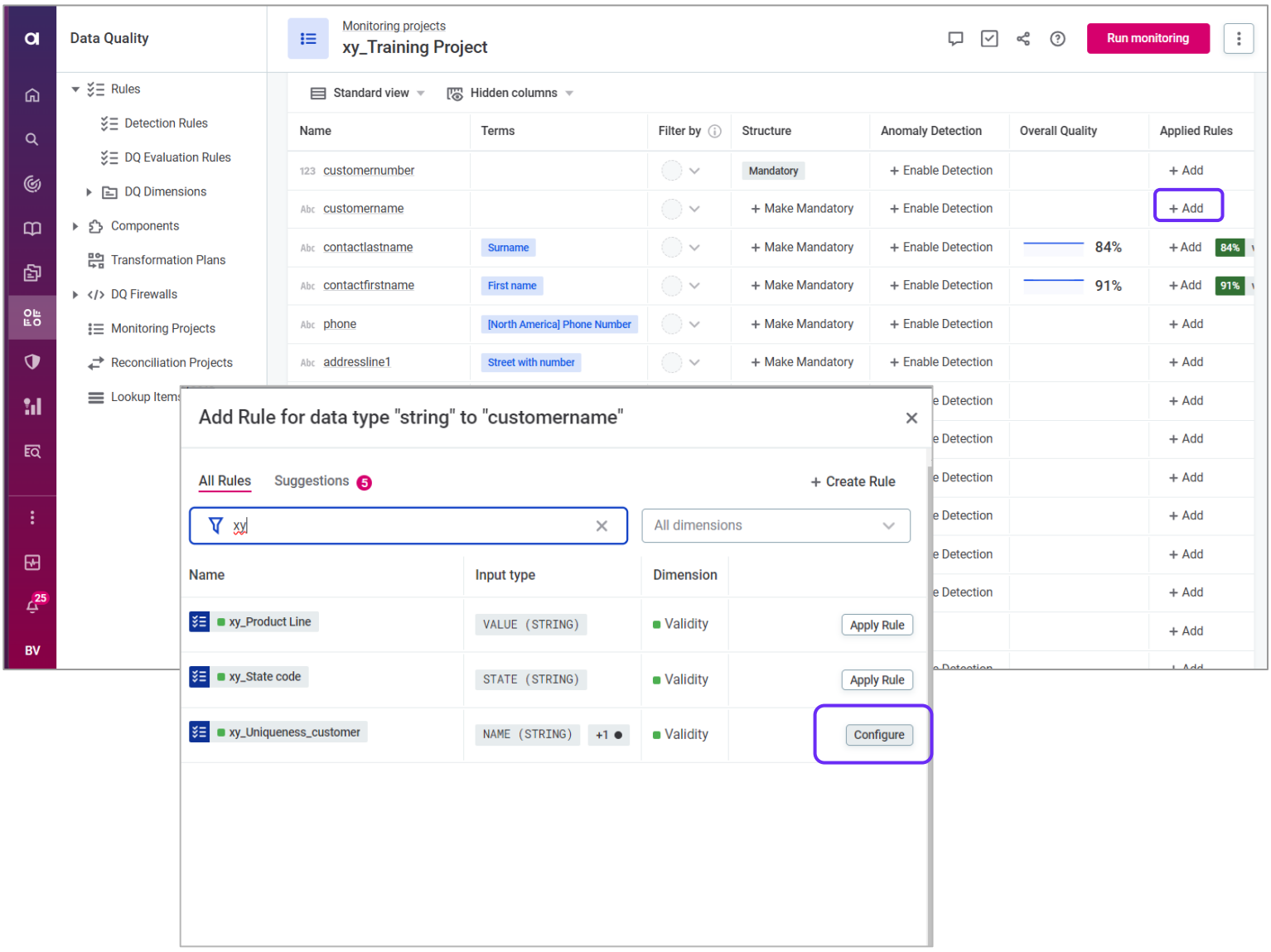

- Test the rule with appropriate values and publish it.



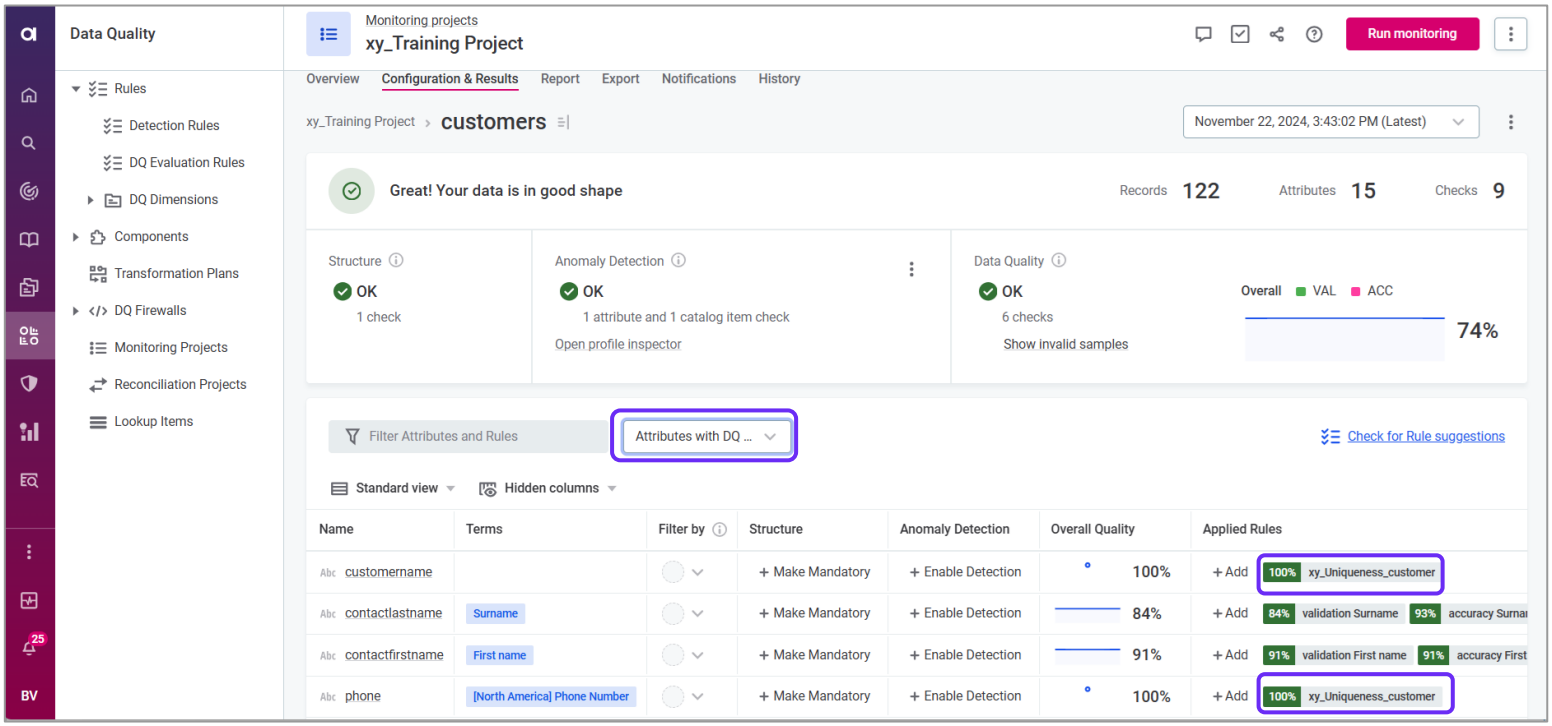
## 1c - Adding Aggregation rule to a Monitoring Project

To see how this rule works, link it to one of the existing Monitoring projects (e.g. Training project).

- Go to the configuration and results tab.
- Select and open the **customers** catalog item.
- Add the rule in the Applied DQ Checks to either the **customername** or **phone** attribute.




- Now you should see the rule in DQ Checks of both attributes (**customername** and **phone**).
- **Publish** the changes and **run** the project to see the results.
- Choose the **attributes with DQ checks** option from the drop-down menu to display a list of **only** attributes with applied rules.

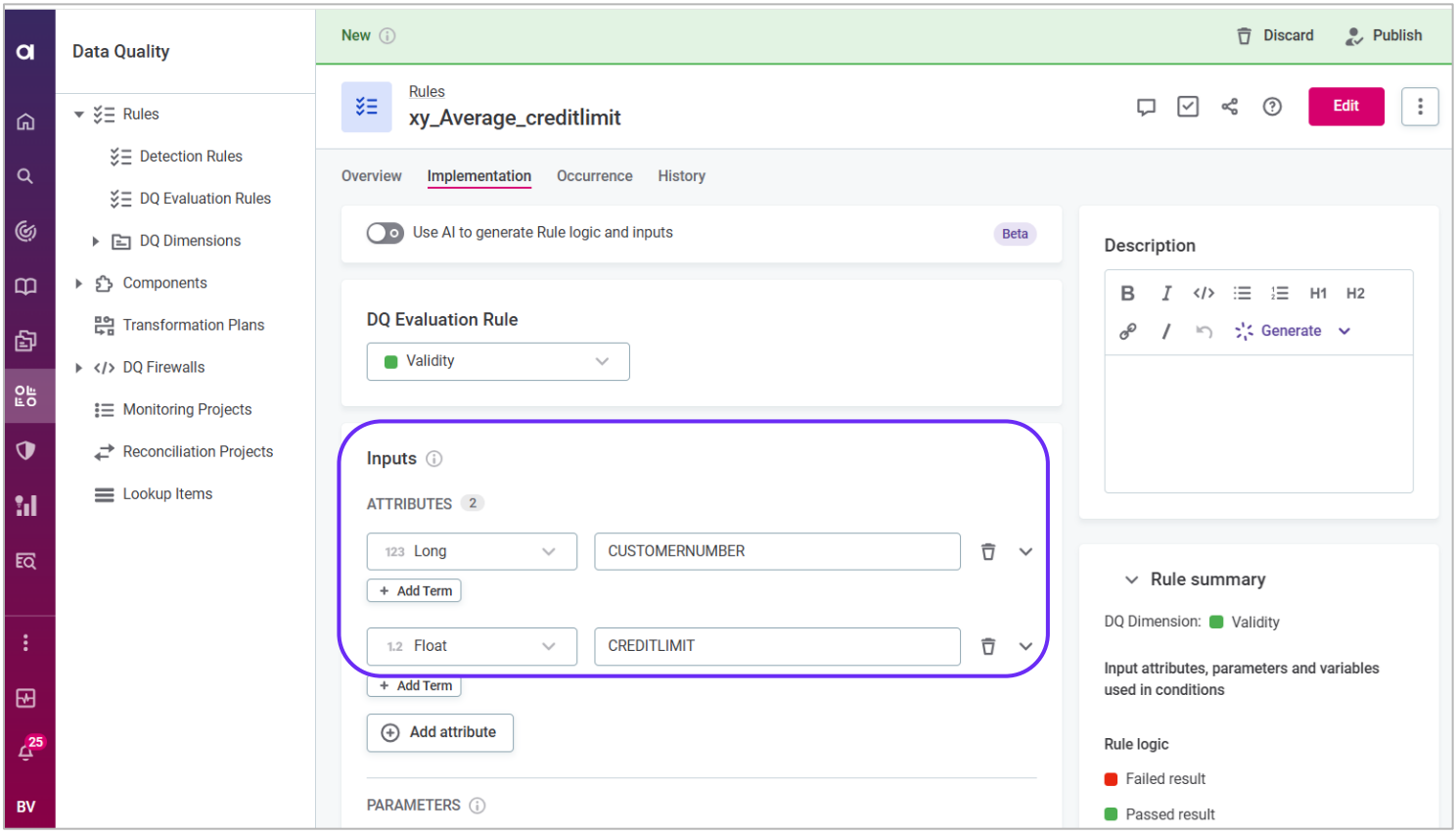


## 2 – Create a DQ aggregation rule to compare each value with the average value

Navigate again to the **Knowledge Catalog,** catalog item **customers,** and look at the values of the column called **creditlimit.** We want to evaluate **whether the credit limit of each customer is within 60% of the total average credit limit (+/-30% from AVG).**

### 2a – Implementing Rule Logic

- Create a new rule (**<prefix>_Average_creditlimit**) in Data Quality > Rules and select **validity** rule.
- Choose **Aggregation rule** Rule Logic in implementation tab and leave **Group by** blank.
- Add the input attributes **CUSTOMERNUMBER(LONG)** and **CREDITLIMIT(FLOAT).**

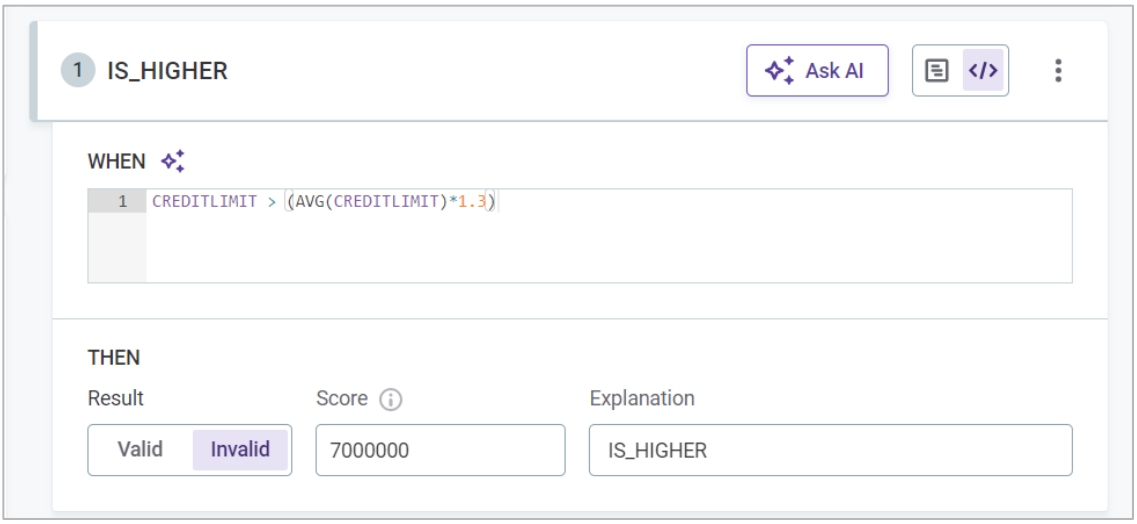



- Set the rule logic so that higher or lower the 30% range of the average is considered **INVALID**. This has two conditions:
  - If the value is higher than 30% of the average.
  - Use the below expression or Ask AI to write appropriate code for you.

    **Expression:** CREDITLIMIT > (AVG(CREDITLIMIT)*1.3)

    **Explanation:** IS_HIGHER

**Score**: 700000

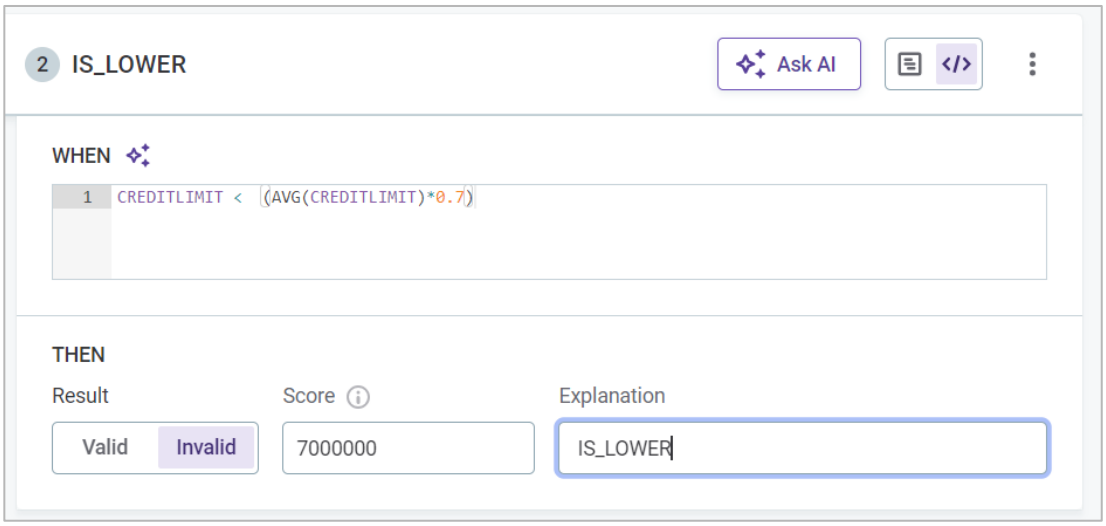



- If the value is lower than 30% of the average.
- Use the below expression or Ask AI to write appropriate code for you.

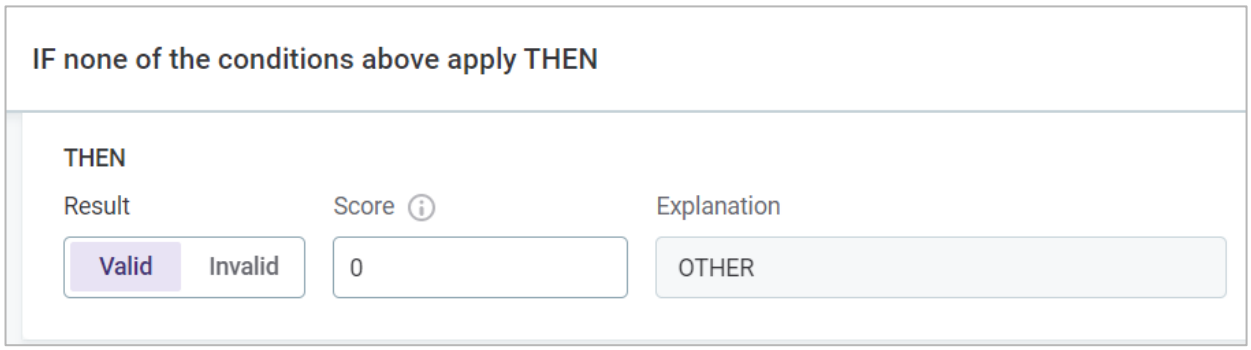
**Expression:** CREDITLIMIT < (AVG(CREDITLIMIT)*0.7)

**Explanation:** IS_LOWER

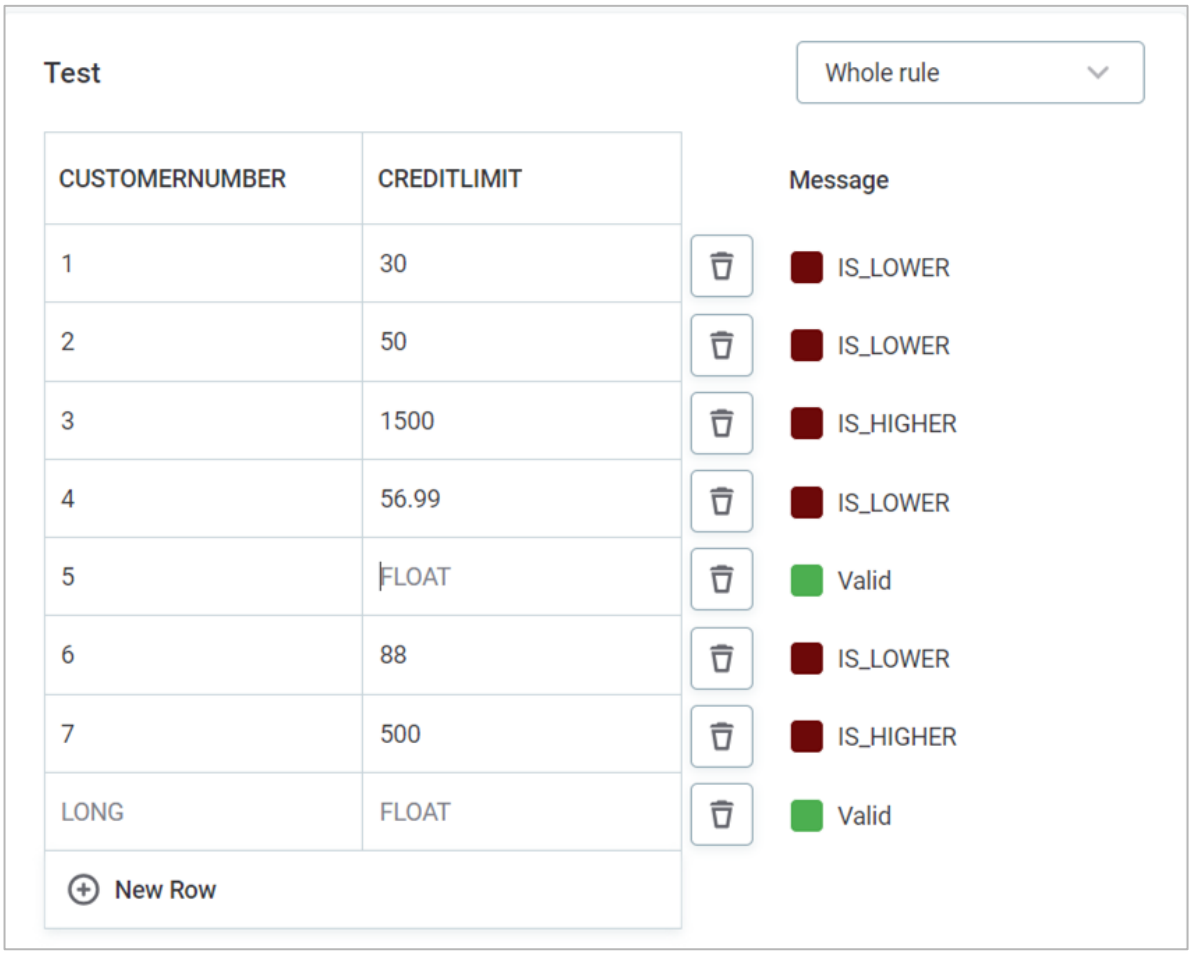
**Score:** 700 000




- The rest of the values that are within the range would be considered as **Valid**.



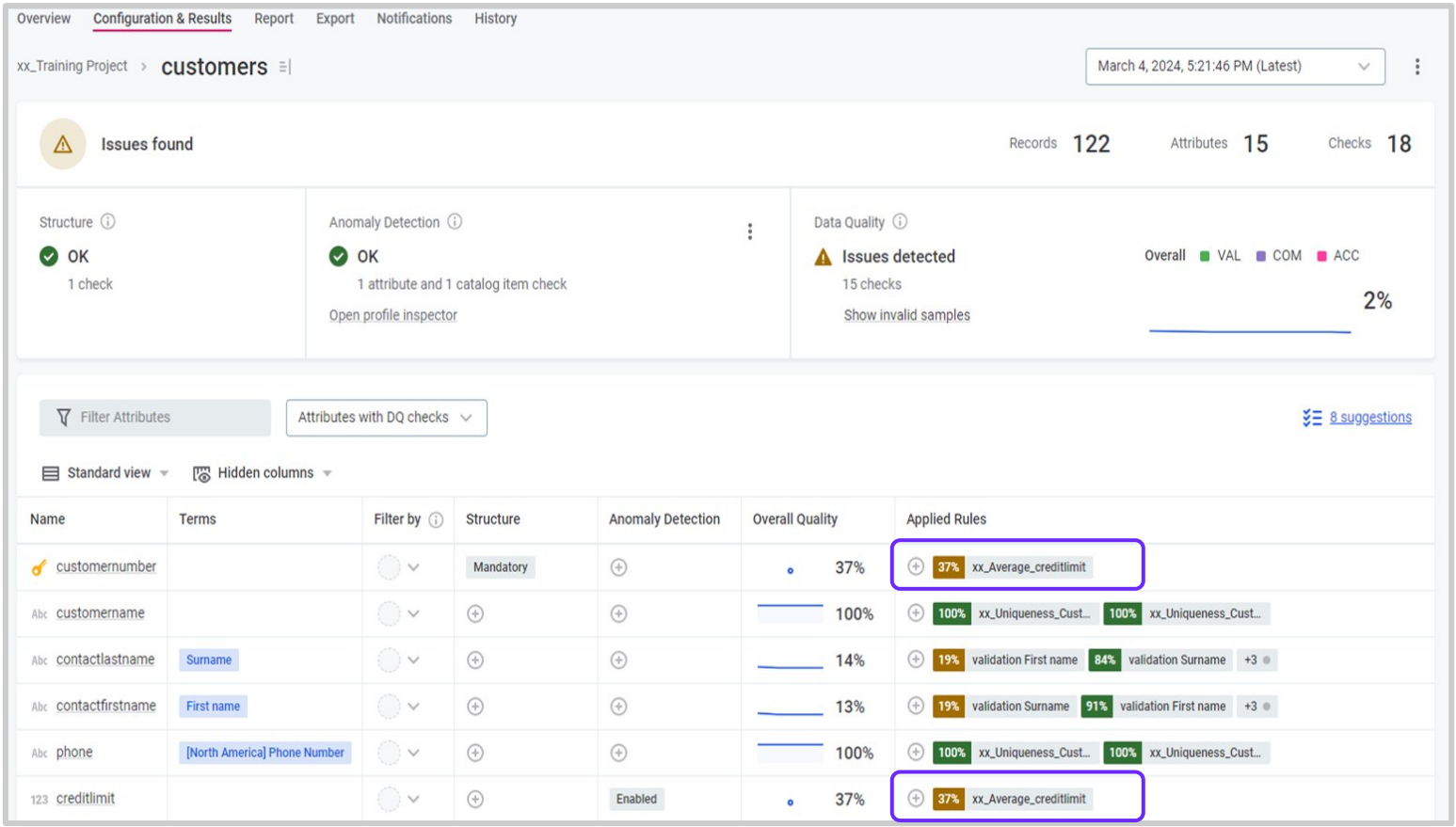
› Now test the rule to see if it is functioning as intended before publishing it.




TASK *All by yourself now! Try changing the rule to compare the credit limit of each customer with the average of his state.*

## 2b – Adding a rule to a Monitoring Project

- In **Monitoring Project**, go to **Training project** > **Configuration & Results** > **customers**
- Add the rule in the Applied DQ Checks to either the **customernumber** or **creditlimit** attribute and configure the inputs accordingly.
- **Publish** the changes and **run** the project to monitor the results.



# Conclusion

We have come to the end of this workshop!
We have created two Aggregation rules using Group by and Aggregate functions. If applicable, you can practice creating more complex Aggregation rules using both Group by and Aggregate functions in the same logic.