# **Rabobank Customer Statement Processor**

## Steps to run the application

Here, I have used eclipse IDE for development.

**Step 1:** Download the code base from below URL

<https://github.com/karthik-ramachandran/rabo.git>

If zip file please extract.

**Step 2:** Import the project into your work space as a maven project.

File > Import > choose “Maven” option > Existing Maven Project [Browse to the local drive and choose the extracted project]

**Step 3:**  Select the project “rabo-services” > right click > Run As > Maven Install

**Step 4:** Navigate to “com.rabo.statementprocessing.RaboApplication” > right click > Run As > Java

Application

Now your application started and run in the default port 8080, now you can access the end-point.

## To reach the end-point:

You can use Postman, to hit the end point by providing the below detail

**URL:** [**http://localhost:8080/process/customer**](http://localhost:8080/process/customer)

**METHOD: POST**

**BODY :** choose “raw” radio button and select “text/plain”, now you can give your input.

*[*

*{*

*"reference": 194261,*

*"accountNumber": "NL91RABO0315273637",*

*"description": "Clothes from Jan Bakker",*

*"startBalance": 21.6,*

*"mutation": -41.83,*

*"endBalance": -20.23*

*}*

*]*

You can execute the below scenario to test the application.

All the scenario displayed below are given as postman collection, you can import in you postman



## **Scenario 1:** **No duplicate reference and correct end balance**

The below input are unique in reference number

## Input Data:

[

{

"reference": 194261,

"accountNumber": "NL91RABO0315273637",

"description": "Clothes from Jan Bakker",

"startBalance": 21.6,

"mutation": -41.83,

"endBalance": -20.23

},

{

"reference": 112806,

"accountNumber": "NL27SNSB0917829871",

"description": "Clothes from Jan Bakker",

"startBalance": 91.23,

"mutation": +15.57,

"endBalance": 106.8

},

{

"reference": 183049,

"accountNumber": "NL69ABNA0433647324",

"description": "Clothes from Jan Bakker",

"startBalance": 86.66,

"mutation": +44.5,

"endBalance": 131.16

},

{

"reference": 183356,

"accountNumber": "NL74ABNA0248990274",

"description": "Subscription for Peter de Vries",

"startBalance": 92.98,

"mutation": -46.65,

"endBalance":46.33

}

]

### Output:

The status code shout be 200

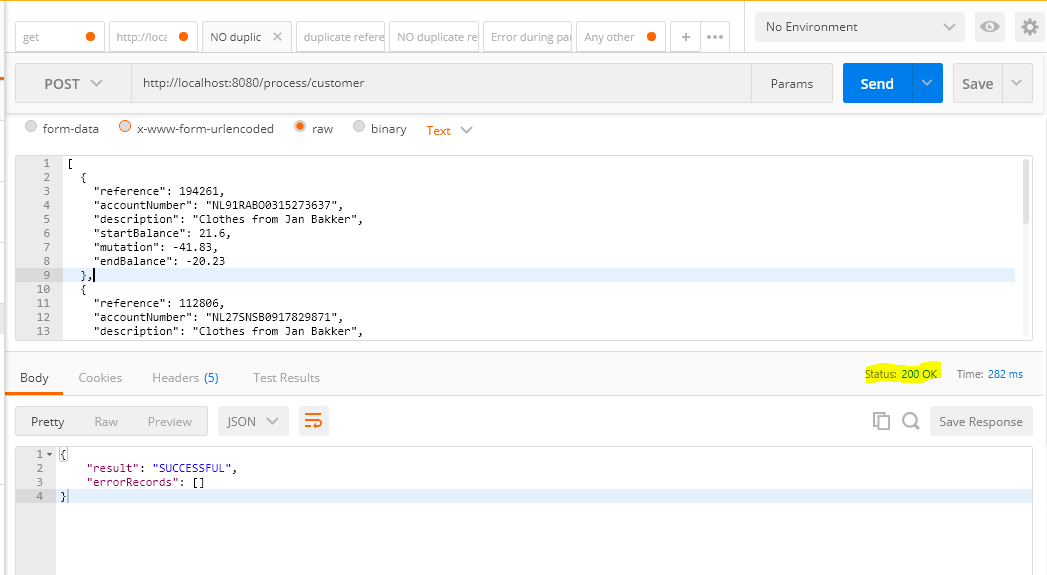
{

"result": "SUCCESSFUL",

"errorRecords": []

}

### **Screen short:**



## **Scenario 2:** **With duplicate reference and correct end balance**

The below input are duplicate in reference number

## Input Data:

[

{

"reference": 194261,

"accountNumber": "NL91RABO0315273637",

"description": "Clothes from Jan Bakker",

"startBalance": 21.6,

"mutation": -41.83,

"endBalance": -20.23

},

{

"reference": 112806,

"accountNumber": "NL27SNSB0917829871",

"description": "Clothes from Jan Bakker",

"startBalance": 91.23,

"mutation": +15.57,

"endBalance": 106.8

},

{

"reference": 194261,

"accountNumber": "NL69ABNA0433647324",

"description": "Clothes from Jan Bakker",

"startBalance": 86.66,

"mutation": +44.5,

"endBalance": 131.16

},

{

"reference": 183356,

"accountNumber": "NL74ABNA0248990274",

"description": "Subscription for Peter de Vries",

"startBalance": 92.98,

"mutation": -46.65,

"endBalance":46.33

}

]

### Output:

The status code shout be 200

{

"result": "DUPLICATE\_REFERENCE",

"errorRecords": [

{

"reference": "194261",

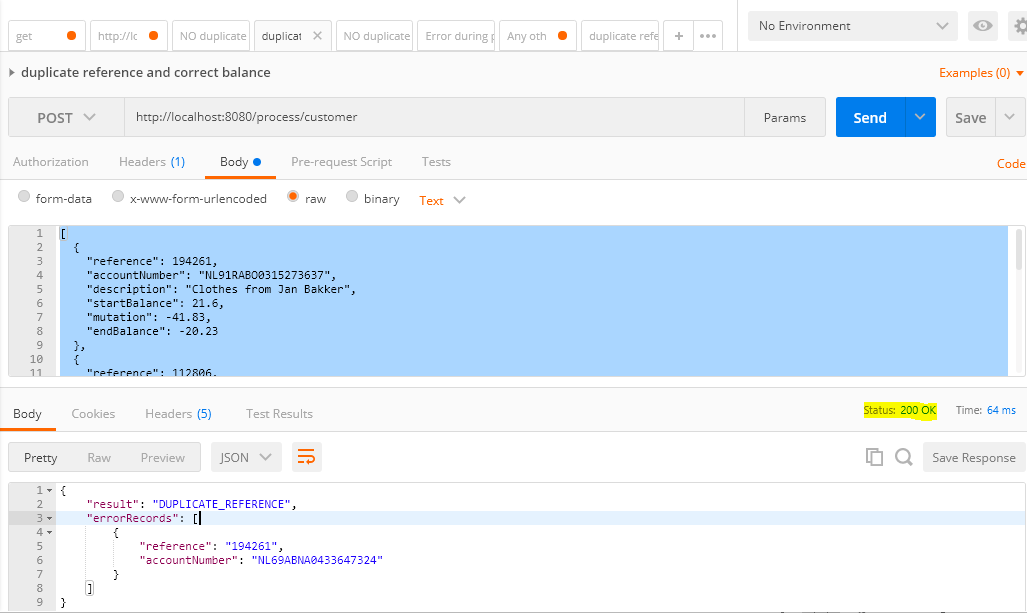
"accountNumber": "NL69ABNA0433647324"

}

]

}

### Screen short:



## **Scenario 3:** **No duplicate reference and incorrect end balance**

The below input are unique in reference number and incorrect end balance

## Input Data:

[

{

"reference": 194261,

"accountNumber": "NL91RABO0315273637",

"description": "Clothes from Jan Bakker",

"startBalance": 21.6,

"mutation": -41.83,

"endBalance": -20.23

},

{

"reference": 112806,

"accountNumber": "NL27SNSB0917829871",

"description": "Clothes from Jan Bakker",

"startBalance": 91.23,

"mutation": "+15.57",

"endBalance": 106.8

},

{

"reference": 183049,

"accountNumber": "NL69ABNA0433647324",

"description": "Clothes from Jan Bakker",

"startBalance": 86.66,

"mutation": +44.5,

"endBalance": 131.16

},

{

"reference": 183356,

"accountNumber": "NL74ABNA0248990274",

"description": "Subscription for Peter de Vries",

"startBalance": 92.98,

"mutation": -46.65,

"endBalance":46.33

},

{

"reference": 139524,

"accountNumber": "NL43AEGO0773393871",

"description": "Flowers from Jan Bakker",

"startBalance": 99.44,

"mutation": +41.23,

"endBalance":1.67

}

]

### Output:

The status code shout be 200

{

"result": "INCORRECT\_END\_BALANCE",

"errorRecords": [

{

"reference": "139524",

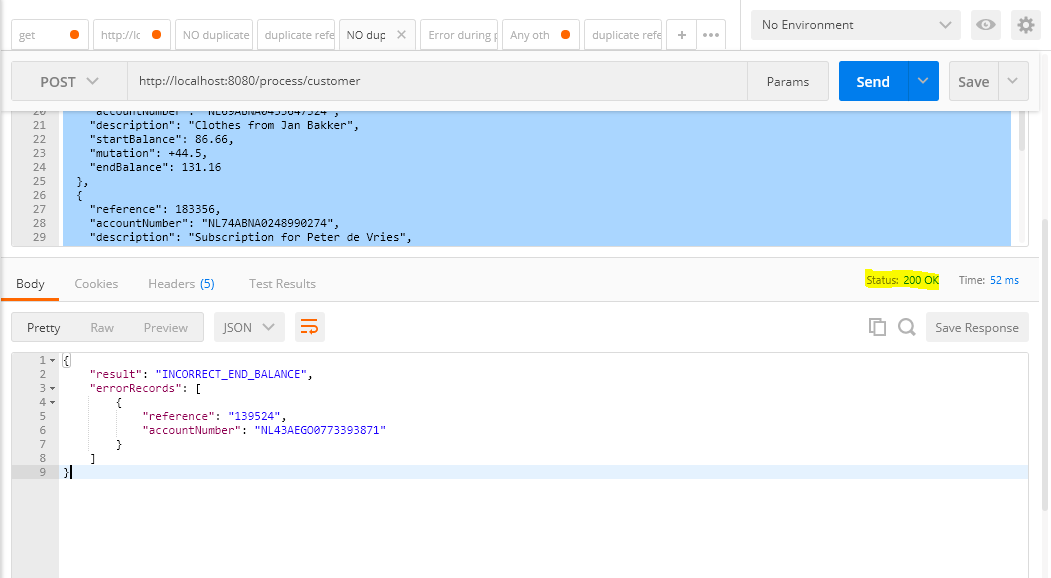
"accountNumber": "NL43AEGO0773393871"

}

]

}

### Screen short



# **Scenario 4:** duplicate reference and Incorrect balance

The below input are duplicate in reference number and incorrect end balance

## Input Data:

[

{

"reference": 194261,

"accountNumber": "NL91RABO0315273637",

"description": "Clothes from Jan Bakker",

"startBalance": 21.6,

"mutation": -41.83,

"endBalance": -20.23

},

{

"reference": 194261,

"accountNumber": "NL27SNSB0917829871",

"description": "Clothes from Jan Bakker",

"startBalance": 91.23,

"mutation": +15.57,

"endBalance": 106.8

},

{

"reference": 183049,

"accountNumber": "NL69ABNA0433647324",

"description": "Clothes from Jan Bakker",

"startBalance": 91.23,

"mutation": +44.5,

"endBalance": 131.16

},

{

"reference": 183356,

"accountNumber": "NL74ABNA0248990274",

"description": "Clothes from Jan Bakker",

"startBalance": 92.98,

"mutation": -46.65,

"endBalance":46.33

}

]

### Output:

The status code shout be 200

{

"result": " DUPLICATE\_REFERENCE\_INCORRECT\_END\_BALANCE",

"errorRecords": [

{

"reference": "194261",

"accountNumber": "NL27SNSB0917829871"

},

{

"reference": "183049",

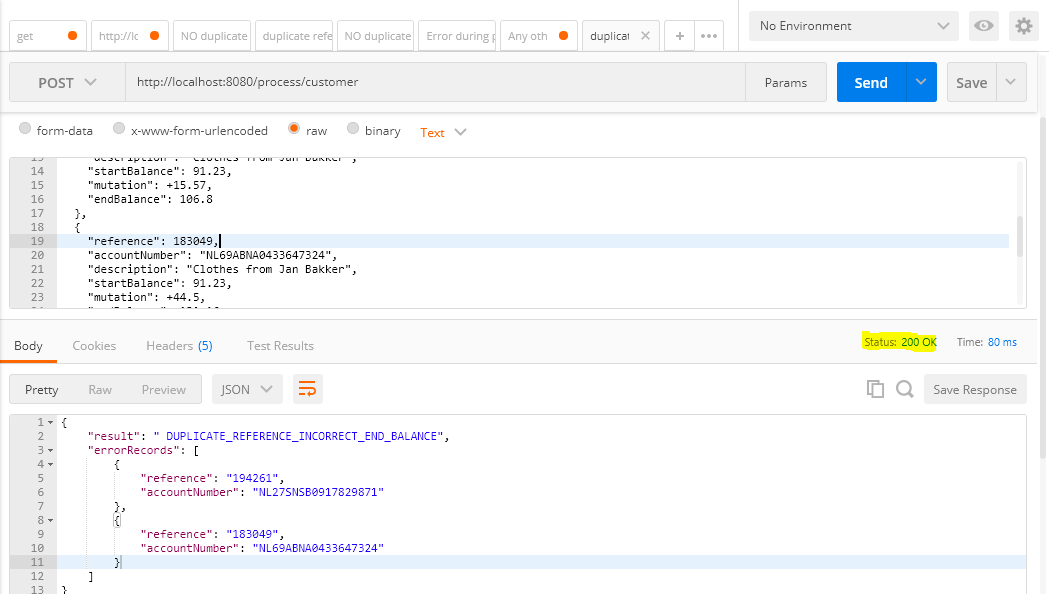
"accountNumber": "NL69ABNA0433647324"

}

]

}

### Screen short



## **Scenario 5:** Error during parsing JSON

The below input is invalid and create parser exception

## Input Data:

[

{

"reference": 194261,

"accountNumber": "NL91RABO0315273637",

"description": "Clothes from Jan Bakker",

"startBalance": 21.6,

"mutation": -41.83,

"endBalance": -20.23

},

{

"reference": 194261,

### Output:

The status code shout be 400

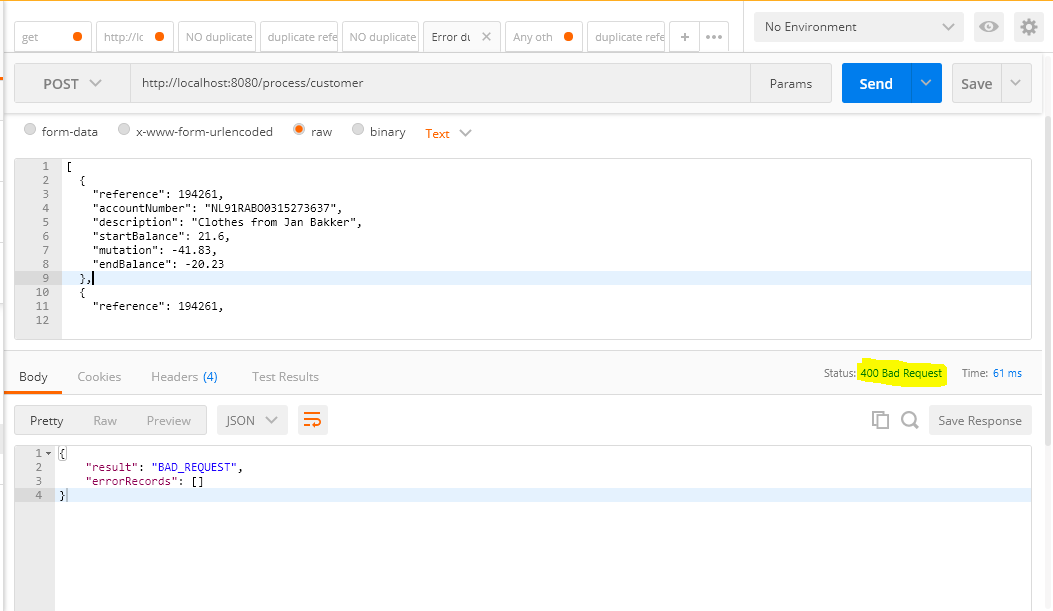
{

"result": "BAD\_REQUEST",

"errorRecords": []

}

### Screen short



## **Scenario 5:** Any other situation

The below input is invalid empty creates internal server error

## Input Data:

<<EMPTY>>

### Output:

The status code shout be 500

{

"result": "INTERNAL\_SERVER\_ERROR",

"errorRecords": []

}

### Screen short

