



AWS FOR DQE

Legal Notice: This document contains privileged and/or confidential information and may not be disclosed, distributed or reproduced without the prior written permission of EPAM®.

Confidential

1. Lambda function created for parsing JSON.

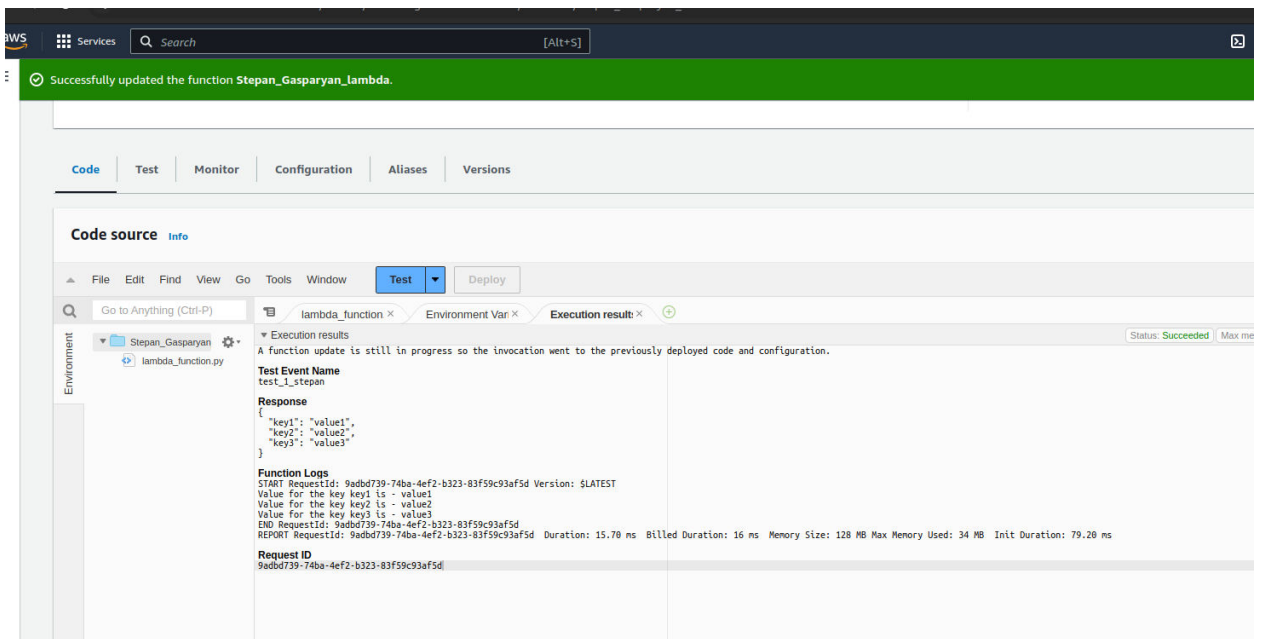
The screenshot displays the AWS Lambda console for the function 'Stepan_Gasparyan_lambda'. The 'Function overview' tab is active, showing a diagram of the function with one layer. The 'Configuration' tab is also visible, showing the VPC configuration (vpc-06706d0e729f3fb1a) and subnets (subnet-0d47558495737dd53, subnet-0583a807e45f16167, subnet-08b167bfa25c0b0dd). The function is configured to run in the 'eu-central-1' region.

Code in lambda function:

The screenshot displays the AWS Lambda console for the function 'Stepan_Gasparyan_lambda', showing the 'Code source' tab. The code is a Python script named 'lambda_function.py' that imports 'json' and 'redshift_connector' and defines a 'lambda_handler' function. The code is as follows:

```
1 import json
2 import redshift_connector
3
4
5
6
7 def lambda_handler(event, context):
8     for key, value in event.items():
9         print(f"Value for the key {key} is - {value}")
10    return event
11
12
```

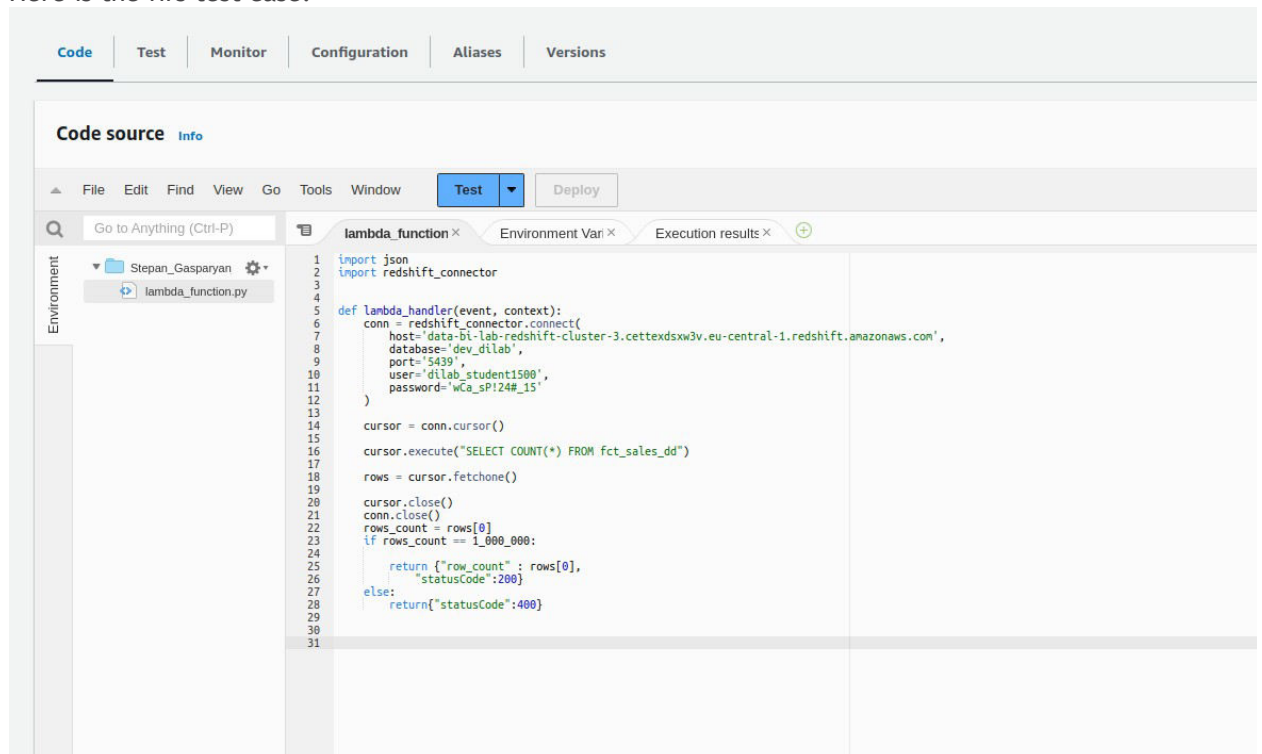
Result:



2. I changed previous lambda function code to test case, but unfortunately can't name it to put corresponding name to it, but hope it's ok :)

I'm checking the row count in fact table: It should be 1.000.000

Here is the first test case:



Then we are going to delete one row from our fact table.

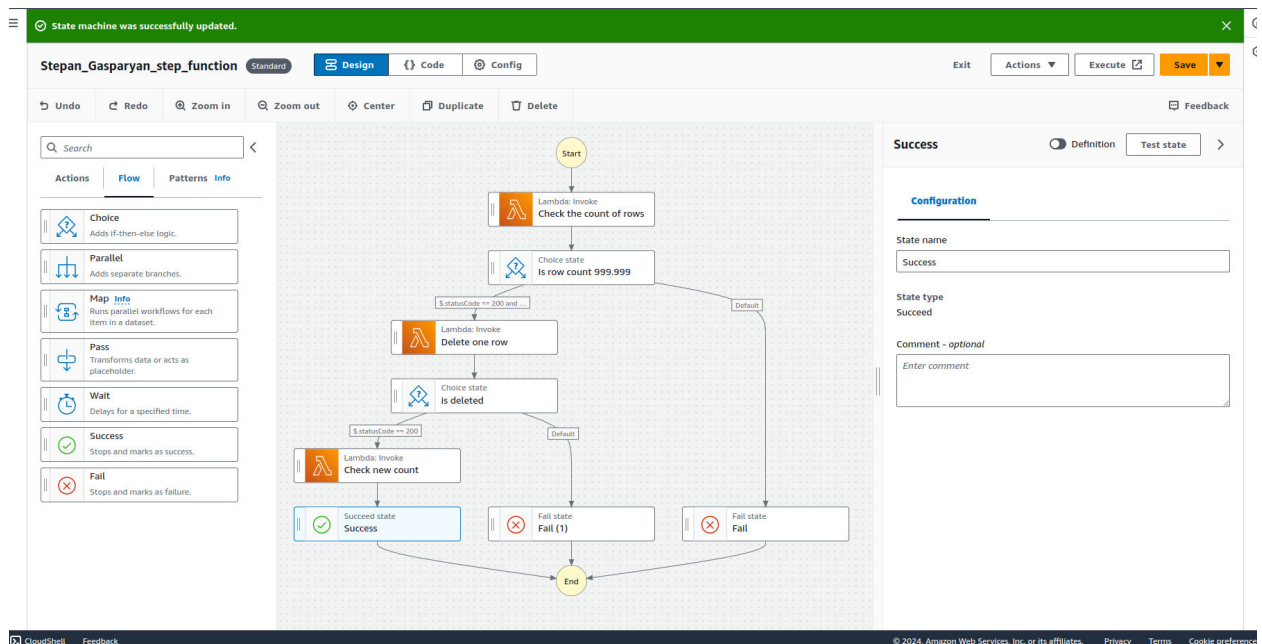
```
1 import json
2 import redshift_connector
3
4
5 def lambda_handler(event, context):
6     conn = redshift_connector.connect(
7         host='data-bi-lab-redshift-cluster-3.cettexdxw3v.eu-central-1.redshift.amazonaws.com',
8         database='dev_dilab',
9         port='5439',
10        user='dilab_student1500',
11        password='wCa_sP!24#_15'
12    )
13
14    cursor = conn.cursor()
15
16    cursor.execute('DELETE FROM fct_sales_dd WHERE sales_surr_id = 2688078')
17
18    rows_deleted = cursor.rowcount
19    conn.commit()
20    cursor.close()
21    conn.close()
22    id rowcount ==0:
23        return {'statusCode':400}
24    return {'statusCode':200}
25
26
27
```

It will return status code 200 if there is a row deleted , otherwise code 400.

Then we will check if it deletes exactly one row from table:

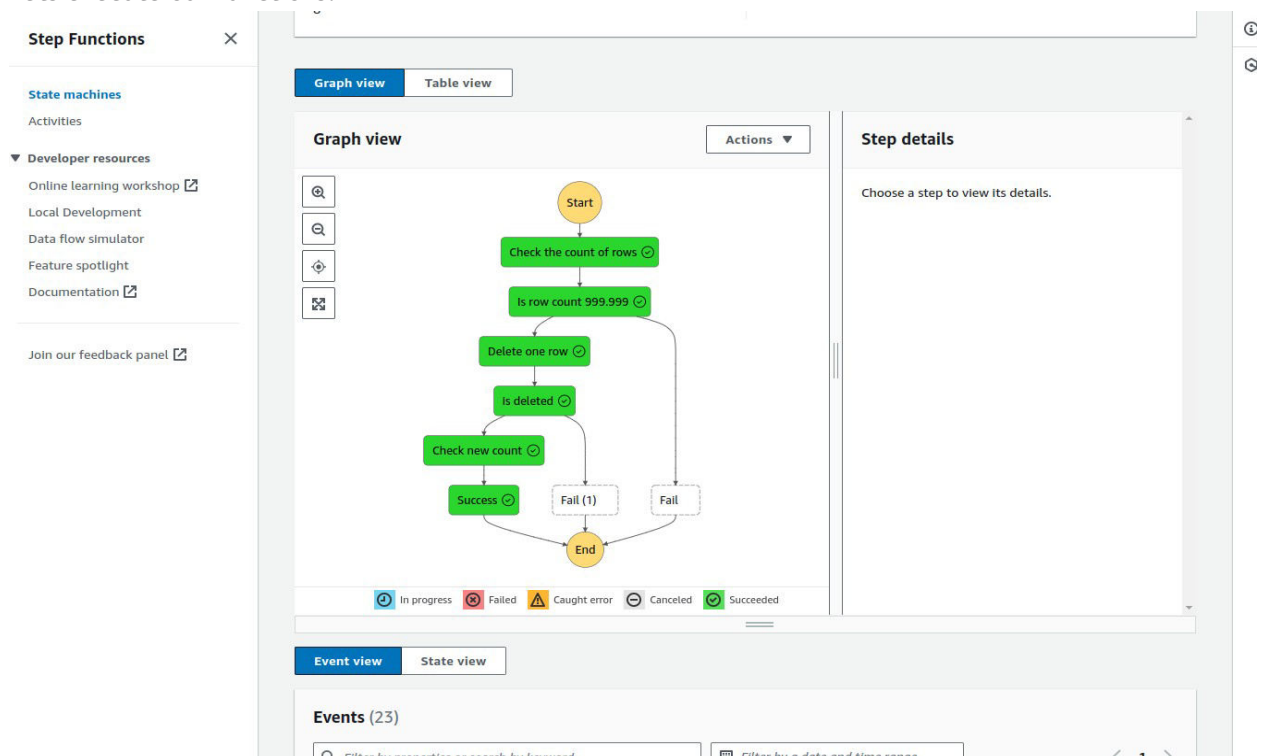
```
1 import json
2 import redshift_connector
3
4
5 def lambda_handler(event, context):
6     conn = redshift_connector.connect(
7         host='data-bi-lab-redshift-cluster-3.cettexdxw3v.eu-central-1.redshift.amazonaws.com',
8         database='dev_dilab',
9         port='5439',
10        user='dilab_student1500',
11        password='wCa_sP!24#_15'
12    )
13
14    cursor = conn.cursor()
15
16    cursor.execute('SELECT COUNT(*) FROM fct_sales_dd')
17
18    rows = cursor.fetchone()
19
20    cursor.close()
21    conn.close()
22    rows_count = rows[0]
23    if rows_count == 999_999:
24        return {'row_count': rows[0],
25            'statusCode':200}
26    else:
27        return {'statusCode':400}
28
29
30
31
32
```

Our Step Function diagram looks like this.



First we just run query to get count then it checks if status code is 200 goes to delete 1 row , then if delete was successful going to check new count of rows.

Lets execute our functions:



Success :)