

ASSESSMENT - 01

Agateeswaran K

ACE11689

Task: Create an API with a POST method

API Name: EmployeeAPI

API Context: /process-data

Description: This API endpoint accepts POST requests containing data to be processed. It verifies if a specific EmployeeId is present in the request payload. If the field is present, it respond with a modified payload. If the field is not present, it returns an error response.

Steps:

- 1) Read the request payload using the property mediator.
- 2) Verify the presence of a specific field using the filter mediator. Check the condition, whether the employeeId is present or not.

3) If the field is present:

Modify the FullName by concatenate of FirstName and LastName

Generate the Success response using payload factor Mediator

Process Stop

4) If the field is not present:

Set an error code and error message property.

Call a fault sequence to generate an error message response using payload factor Mediator.

Process Stop

Sample Payload Request:

```
{  
  "Employee": {  
    "FirstName": "John",  
    "LastName": "Micheal",  
    "Role": "Manager",
```

```
"EmployeeId": "1112233"
```

```
}
```

```
}
```

Sample Response (Success):

```
{
```

```
  "message": "Data processed successfully",
```

```
  "processed_data": {
```

```
    "FullName": "John Micheal", // Concat(first and lastname)
```

```
    "Role": "Manager",
```

```
    "EmployeeId": "1112233"
```

```
  }
```

```
}
```

Sample Response (Error):

```
{
```

```
  "error": {
```

```
    "code": "400",
```

```
    "message": "'EmployeeId' is missing in the request payload"
```

```
  }
```

```
}
```

Code:

EmployeeAPI.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<api context="/process-data" name="EmployeeAPI"
xmlns="http://ws.apache.org/ns/synapse">
  <resource methods="POST">
    <inSequence>
      <property name="employeeedata" expression="json-eval($)" type="STRING"
scope="default"/>
      <property name="firstname" expression="json-eval($.FirstName)" type="STRING"
scope="default"/>
```

```

    <property name="lastname" expression="json-eval($.LastName)" type="STRING"
scope="default"/>
    <property name="employeeerole" expression="json-eval($.Role)" type="STRING"
scope="default"/>
    <property name="employeeid" expression="json-eval($.EmployeeId)"
type="STRING" scope="default"/>
    <property name="employeeename" expression="fn:concat($ctx:firstname, ' ',
$ctx:lastname)" type="STRING" scope="default"/>
    <filter regex="true" source="boolean($ctx:employeeid)">
        <then>
            <payloadFactory media-type="json">
                <format>
                    {
                        "message": "Data processed successfully",
                        "processed_data": { "FullName": "$1", "Role": "$2", "EmployeeId": "$3" }
                    }
                </format>
                <args>
                    <arg evaluator="json"
expression="$ctx:employeeename"/>
                    <arg evaluator="json" expression="$ctx:employeeerole"/>
                    <arg evaluator="json" expression="$ctx:employeeid"/>
                </args>
            </payloadFactory>
        </then>
        <else>
            <property name="errorcode" value="400" type="STRING"
scope="default"/>
            <property name="errormessage" value="EmployeeId is missing in the
request payload" type="STRING" scope="default"/>
            <sequence key="FaultSequence"/>
        </else>
    </filter>
    <respond/>
</inSequence>
<outSequence/>
<faultSequence>
    <property name="errorcode" expression="get-property('ERROR_CODE')"
type="STRING" scope="default"/>
    <property name="errormessage" expression="get-property('ERROR_MESSAGE')"
type="STRING" scope="default"/>
    <sequence key="FaultSequence"/>
</faultSequence>
</resource>
</api>

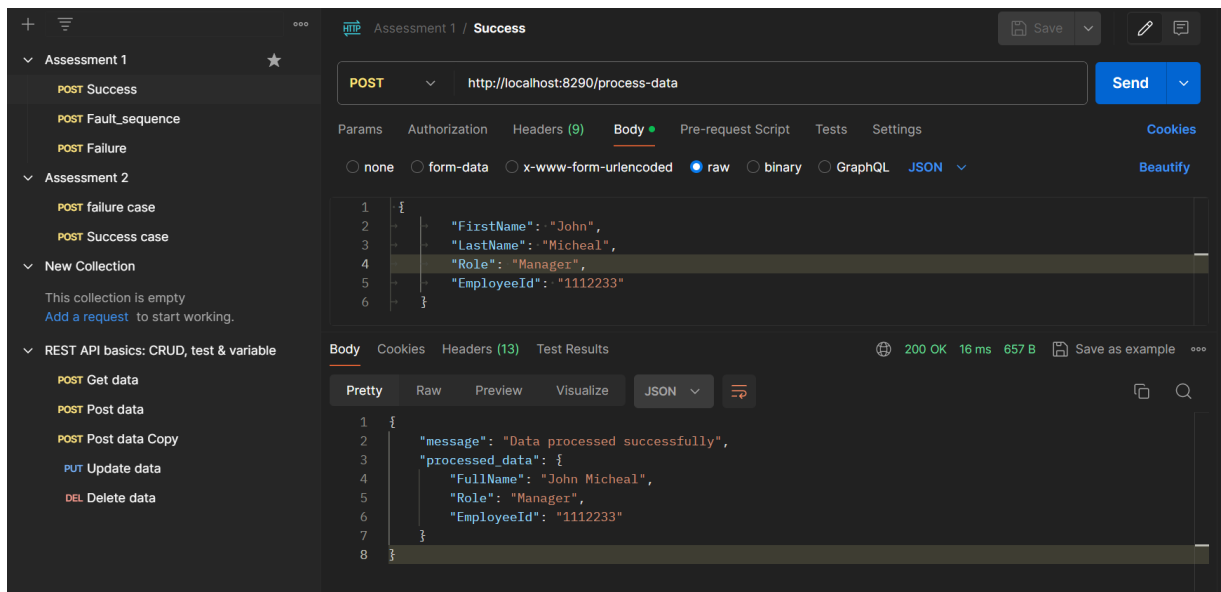
```

FaultSequence.xml

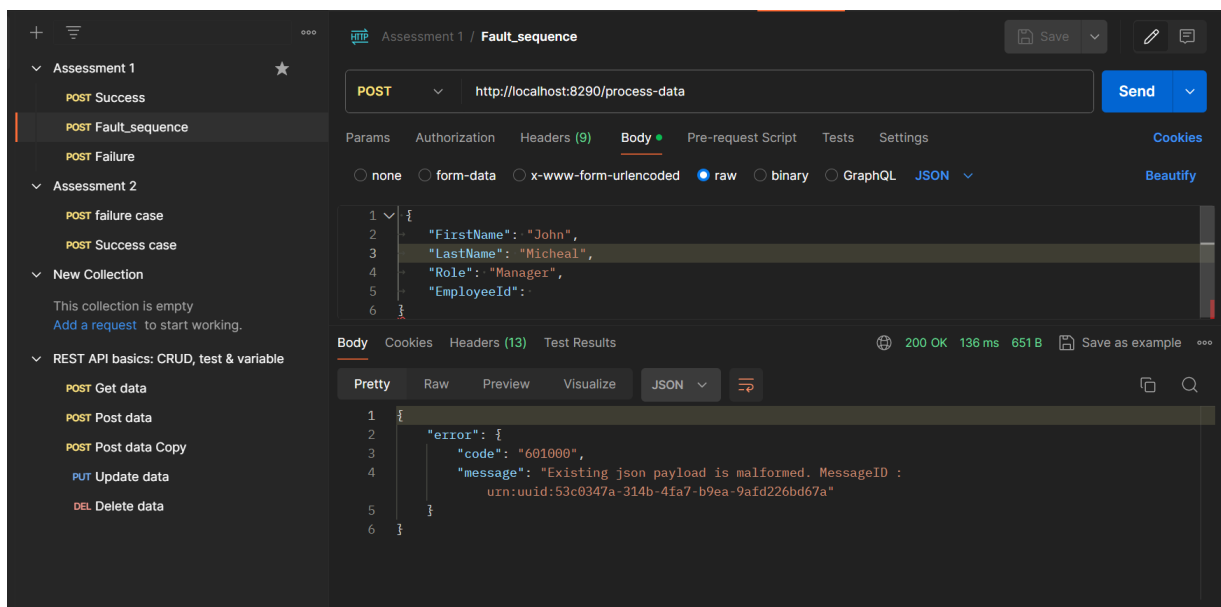
```
<?xml version="1.0" encoding="UTF-8"?>
<sequence name="FaultSequence" trace="disable" xmlns="http://ws.apache.org/ns/synapse">
  <payloadFactory media-type="json">
    <format>
      {
        "error": {
          "code": "$1",
          "message": "$2"
        }
      }
    </format>
    <args>
      <arg evaluator="json" expression="$ctx:errorCode"/>
      <arg evaluator="json" expression="$ctx:errorMessage"/>
    </args>
  </payloadFactory>
  <respond/>
</sequence>
```

Output:

Case1: Success scenario, with emp id.



Case 2: Fault Sequence scenario, without emp id.



Case 3: Error in accessing the API

kspace

NewImport

OverviewPOST SuccessPOST Fault_sequenPOST FailurePOST New Reque+No environment

Assessment 1 / Failure

Save

Send

POSThttp://localhost:8290/process-dat

ParamsAuthorizationHeaders (8)BodyPre-request ScriptTestsSettingsCookies

noneform-datax-www-form-urlencodedrawbinaryGraphQLJSON

Beautify

1

BodyCookiesHeaders (10)Test Results404 Not Found6 ms354 BSave as example

PrettyRawPreviewVisualizeText

1