

ASSESSMENT - 02

Agateeswaran K

ACE11689

Task:

Scheduler Task Name : EmployeeVerificationTrigger_SCHD Type :cron or simple

API Name: EmpDetailsVerification_API context:/EmployeeDetails

Sequence Name : ProcessEmployeeDetails_SEQ

Process:

1. Scheduler should call the ProcessEmployeeDetails_SEQ
2. Under sequence create the payloadFactory mediator and pass the below payload

```
{
  "EmployeeDetails": {
    "Employee": [
      {
        "Name": "Anu",
        "ID": "DEV789"
      },
      {
        "Name": "Priya",
        "ID": "ACE965"
      },
      {
        "Name": "Gayathri",
        "ID": "EMP123"
      }
    ]
  }
}
```

}

3. In sequence Make an endpoint call using call mediator and call EmpDetails_API
4. In API read the payload and create iterate mediator to loop the records
5. Inside Iterate create switch mediator and verify the ID field of the Employee starts with 'ACE'

If Employee ID starts with ACE means create a payload using PayloadFactory mediator as shown below

```
<VerificationDetails>  
<Name>Anu</Name>  
<IdVarification>Success</IdVerification>  
<VerificationDetails>
```

If Employee ID does not starts with ACE means create a payload using PayloadFactory mediator as shown below

```
<VerificationDetails>  
<Name>Anu</Name>  
<IdVarification>Failed</IdVerification>  
<VerificationDetails>
```

6. Aggregate the reponse in EmpDetails_API API and the final response should be displayed in log as follows

```
<EmployeeVerificationDetails>  
<IdVarification>Failed</IdVerification>  
<IdVarification>Succes</IdVerification>  
<IdVarification>Failed</IdVerification>  
<EmployeeVerificationDdetails>
```

EmployeeVerificationTrigger_SCHD.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<task class="org.apache.synapse.startup.tasks.MessageInjector"
group="synapse.simple.quartz" name="EmployeeVerificationTrigger_SCHD"
xmlns="http://ws.apache.org/ns/synapse">
  <trigger count="1" interval="10"/>
  <property name="message"
xmlns:task="http://www.wso2.org/products/wso2commons/tasks">
    <messages xmlns=""/>
  </property>
  <property name="sequenceName" value="ProcessEmployeeDetails_SEQ"
xmlns:task="http://www.wso2.org/products/wso2commons/tasks"/>
  <property name="injectTo" value="Sequence"
xmlns:task="http://www.wso2.org/products/wso2commons/tasks"/>
</task>
```

ProcessEmployeeDetails_SEQ.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<sequence name="ProcessEmployeeDetails_SEQ" trace="disable"
xmlns="http://ws.apache.org/ns/synapse">
  <payloadFactory media-type="json">
    <format>
    {
      "EmployeeDetails": {
        "Employee": [
          {
            "Name": "Anu",
            "ID": "DEV789"
          },
          {
            "Name": "Priya",
            "ID": "ACE965"
          },
          {
            "Name" : "Gayathri",
            "ID": "EMP123"
          }
        ]
      }
    }
  </format>
  <args/>
```

```
</payloadFactory>
<call>
  <endpoint key="EmpDetails_API"/>
</call>
</sequence>
```

EmpDetails_API.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<endpoint name="EmpDetails_API" xmlns="http://ws.apache.org/ns/synapse">
  <http method="post" uri-template="http://localhost:8290/EmployeeDetails">
    <suspendOnFailure>
      <initialDuration>-1</initialDuration>
      <progressionFactor>1.0</progressionFactor>
    </suspendOnFailure>
    <markForSuspension>
      <retriesBeforeSuspension>0</retriesBeforeSuspension>
    </markForSuspension>
  </http>
</endpoint>
```

EmpDetailsVerification_API.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<api context="/EmployeeDetails" name="EmpDetailsVerification_API"
xmlns="http://ws.apache.org/ns/synapse">
  <resource methods="POST">
    <inSequence>
      <iterate expression="//*[local-name()='Employee']"
id="EmployeeVerificationResponse" sequential="true">
        <target>
          <sequence>
            <property expression="//*[local-name()='Name']" name="empname"
scope="default" type="STRING"/>
            <property expression="//*[local-name()='ID']" name="empid" scope="default"
type="STRING"/>
            <switch source="$ctx:empid">
              <case regex="ACE.*">
                <payloadFactory media-type="xml">
                  <format>
                    <VerificationDetails>
                      <Name>$1</Name>
                      <ID>$2</ID>
                      <IdVarification>Success</IdVarification>
```

```

        </VerificationDetails>
    </format>
    <args>
        <arg evaluator="xml" expression="$ctx:empname"/>
        <arg evaluator="xml" expression="$ctx:empid"/>
    </args>
</payloadFactory>
</case>
<default>
    <payloadFactory media-type="xml">
        <format>
            <VerificationDetails>
                <Name>$1</Name>
                <ID>$2</ID>
                <IdVarification>Failed</IdVarification>
            </VerificationDetails>
        </format>
        <args>
            <arg evaluator="xml" expression="$ctx:empname"/>
            <arg evaluator="xml" expression="$ctx:empid"/>
        </args>
    </payloadFactory>
</default>
</switch>
<property name="Aggregated_Responses" type="STRING" scope="default">
    <EmployeeVerificationDetails/>
</property>
<aggregate id="EmployeeVerificationResponse">
    <completeCondition>
        <messageCount max="-1" min="-1"/>
    </completeCondition>
    <onComplete aggregateElementType="root"
enclosingElementProperty="Aggregated_Responses" expression="$body/*"
xmlns:ns="http://org.apache.synapse/xsd">
        <log level="custom">
            <property expression="//*" name="FinalResponse" scope="default"/>
        </log>
        <respond/>
    </onComplete>
</aggregate>
</sequence>
</target>
</iterate>
</inSequence>
<outSequence/>
<faultSequence/>
</resource>
</api>

```

Output:

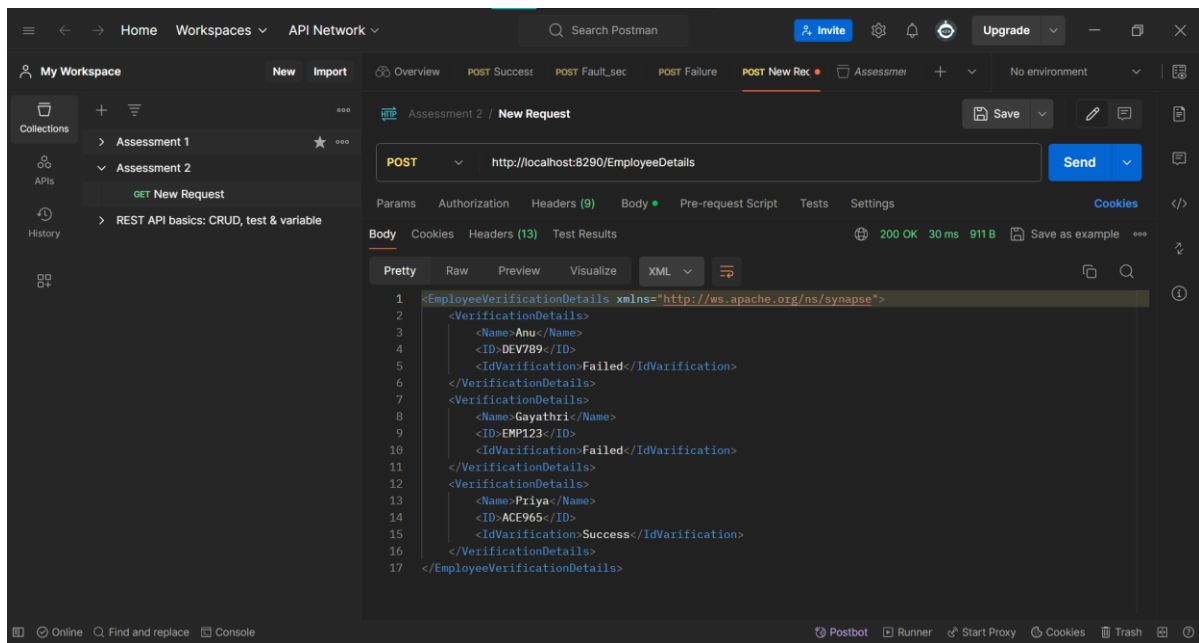
Success case:

Log Output:

```
[2024-03-18 10:33:20,340] INFO {LogMediator} - {api:EmpDetailsVerification_API}
FinalResponse = <soapenv:Body
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"><EmployeeVerificationDetails
xmlns="http://ws.apache.org/ns/synapse">
<VerificationDetails>
<Name>Anu</Name>
<ID>DEV789</ID>
<IdVarification>Failed</IdVarification>
</VerificationDetails>
<VerificationDetails>
<Name>Priya</Name>
<ID>ACE965</ID>
<IdVarification>Success</IdVarification>
</VerificationDetails>
<VerificationDetails>
<Name>Gayathri</Name>
<ID>EMP123</ID>
<IdVarification>Failed</IdVarification>
</VerificationDetails>
</EmployeeVerificationDetails></soapenv:Body>
```

Postman output:

Success Case:



Error Case:

