TGH
Making Integrations Simpler

boomi Partner

Consume Salesforce API Using the REST Client Connector

Author
Sourav Karmakar

Consume Salesforce API Using the REST Client Connector

What is Rest?

➔ Rest stands for *representational state transfer.*
An API, or *application programming interface*, is a set of rules that define how applications or devices can connect to and communicate with each other. A REST API is an API that conforms to the design principles of the REST, or *representational state transfer* architectural style. For this reason, REST APIs are sometimes referred to as RESTful APIs.

Rest Principles

➔ Rest APIs are following six REST design principles

1. **Uniform interface**
2. **Client-server architecture**
3. **Statelessness**
4. **Cacheability**
5. **Layered system architecture**
6. **Code on demand (optional)**

Consume Rest API in Boomi

➔ To consume Rest API, we have 2 connectors in Boomi.

1. HTTP Client Connector
2. Rest Client Connector

Rest Client Connector

➔ The REST Client connector provides the flexibility to connect to any RESTful API service, extract the data, and use it in your process. The connector is generic and is not tailored to any specific REST data source. By being generic, the connector provides a standard way to connect to any RESTful API endpoint service and use the information from the service-specific API, rather than requiring the use of a specific branded connector.

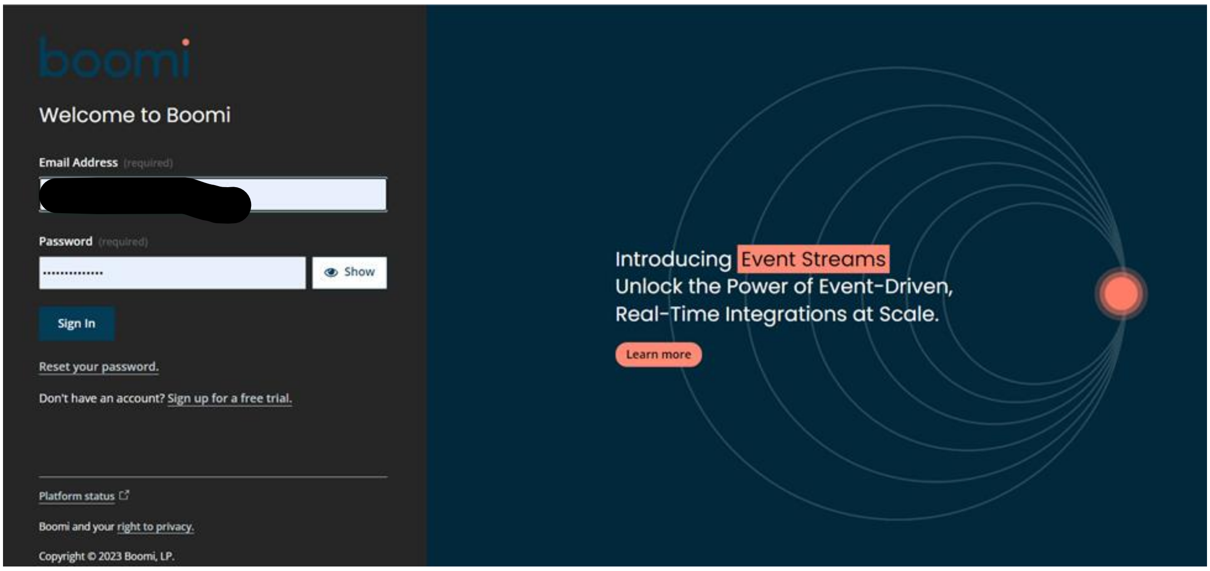
To configure a connector to communicate with an HTTP-enabled server, set up two components:

- REST Client connection
- REST Client operation

Now go to the Boomi Atomsphere Platform to start integrating Salesforce with Boomi Using Rest Client Connector.

Step 1: Go to platform.boomi.com

Use Email Address and Password to log into the Boomi platform



Welcome to Boomi

Email Address (required)

[redacted]

Password (required)

Reset your password.

Don't have an account? Sign up for a free trial.

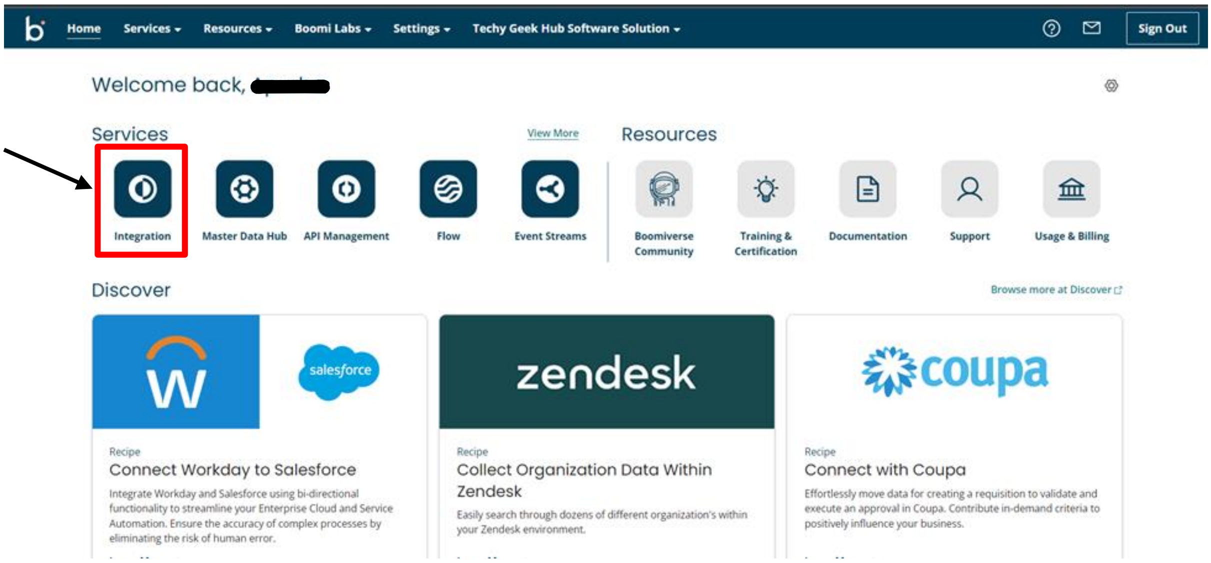
Platform status

Boomi and your right to privacy.

Copyright © 2023 Boomi, LP.

Introducing Event Streams
Unlock the Power of Event-Driven, Real-Time Integrations at Scale.

Step 2: Click on Integration



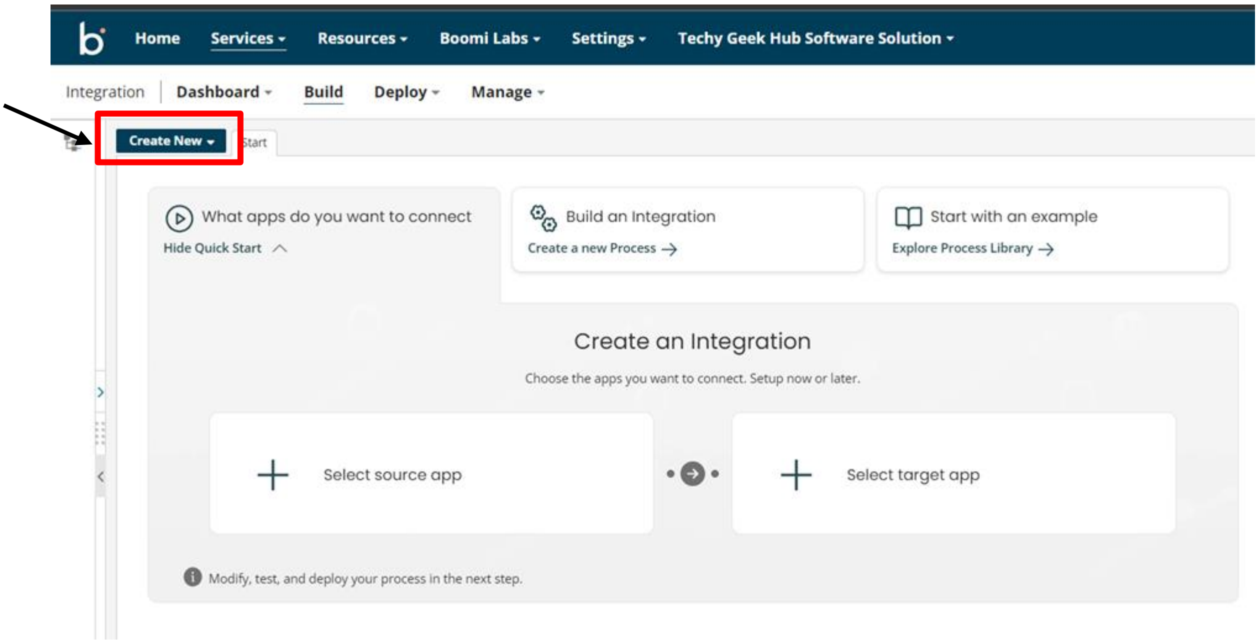
Welcome back, [redacted]

Recipe
Connect Workday to Salesforce
Integrate Workday and Salesforce using bi-directional functionality to streamline your Enterprise Cloud and Service Automation. Ensure the accuracy of complex processes by eliminating the risk of human error.

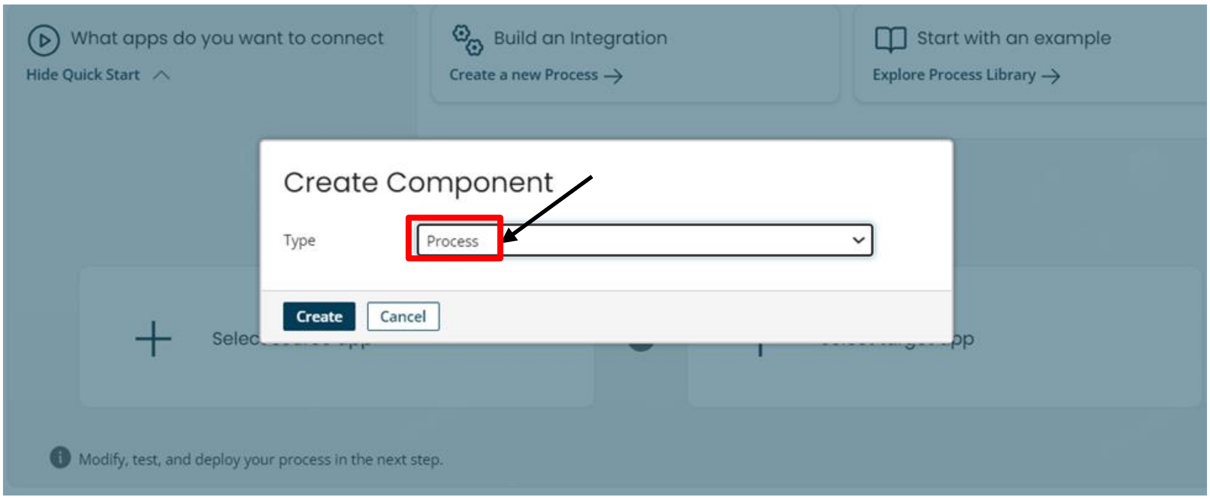
Recipe
Collect Organization Data Within Zendesk
Easily search through dozens of different organization's within your Zendesk environment.

Recipe
Connect with Coupa
Effortlessly move data for creating a requisition to validate and execute an approval in Coupa. Contribute in-demand criteria to positively influence your business.

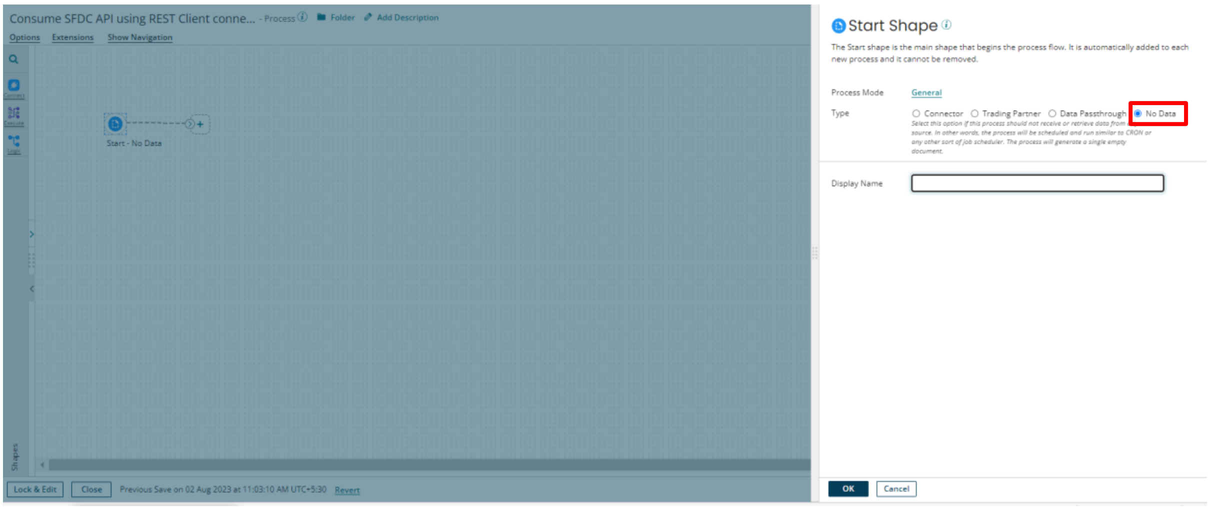
Step 3: Click on Create New



Step 4: Create a Process Component



Step 5: Configure Start Shape with No Data type



Consume SFDC API using REST Client conne... - Process

Options Extensions Show Navigation

Start - No Data

Lock & Edit | Close | Previous Save on 02 Aug 2023 at 11:03:10 AM UTC+5:30 Revert

Start Shape

The Start shape is the main shape that begins the process flow. It is automatically added to each new process and it cannot be removed.

Process Mode: General

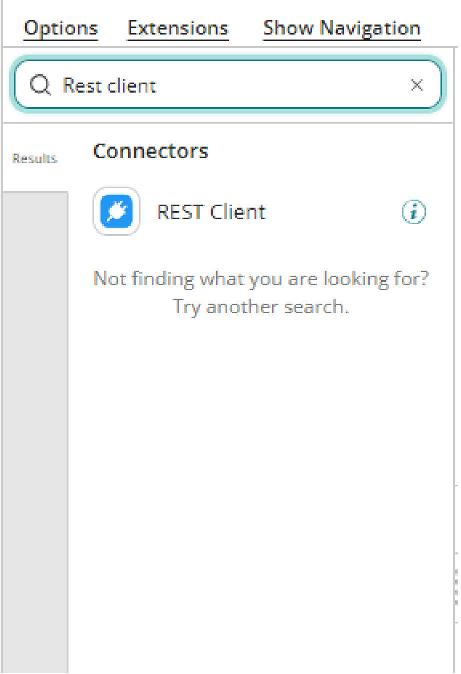
Type: ○ Connector ○ Trading Partner ○ Data Passthrough ● No Data

Select this option if this process should not receive or retrieve data from any source. In other words, the process will be scheduled and run similar to CRON or any other sort of job scheduler. The process will generate a single empty document.

Display Name:

OK | Cancel

Step 6: Search Rest Client in the Shape palette



Options Extensions Show Navigation

Rest client

Results

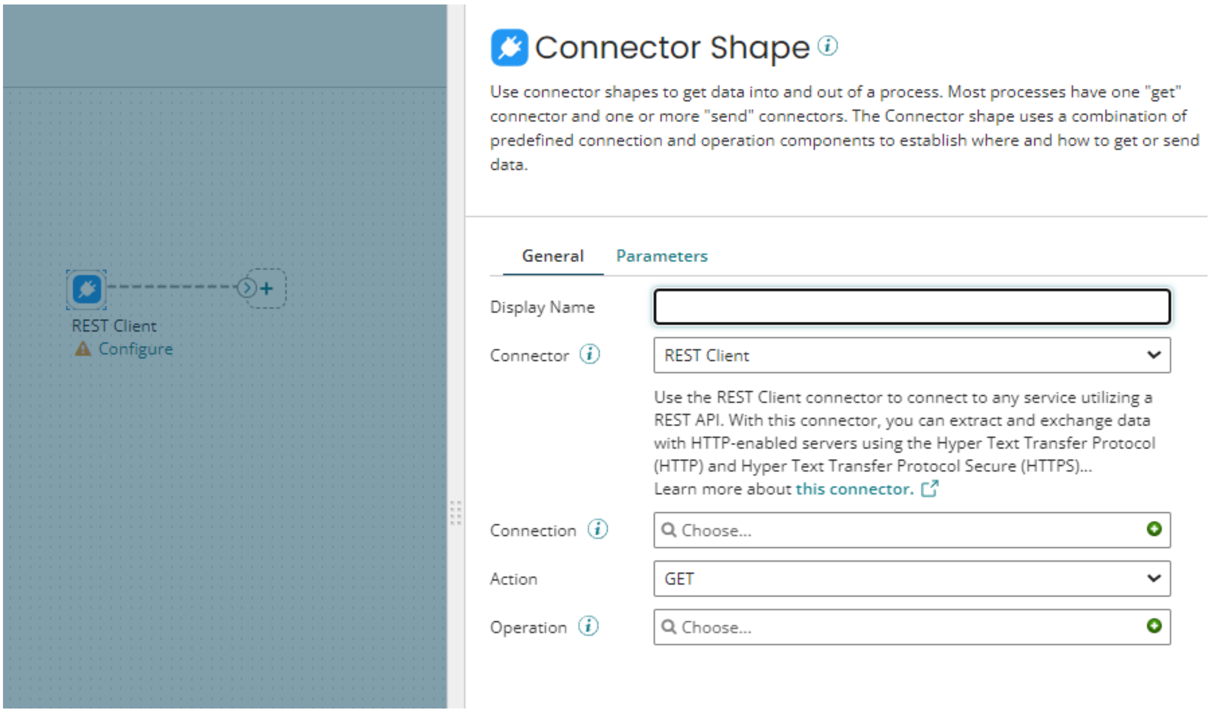
Connectors

REST Client

Not finding what you are looking for? Try another search.

Drag and drop the Rest Client Connector in the process canvas.

Step 7: Drag and drop the Rest Client Connector



Connector Shape

Use connector shapes to get data into and out of a process. Most processes have one "get" connector and one or more "send" connectors. The Connector shape uses a combination of predefined connection and operation components to establish where and how to get or send data.

General Parameters

Display Name

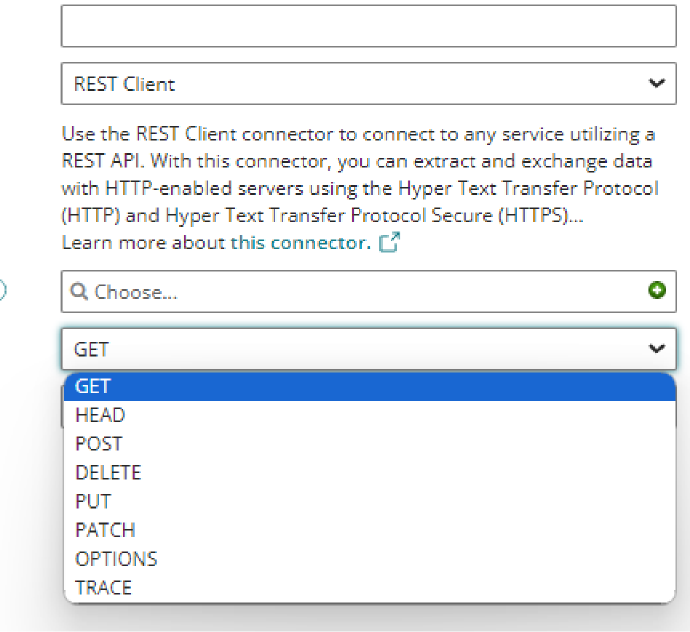
Connector: REST Client

Use the REST Client connector to connect to any service utilizing a REST API. With this connector, you can extract and exchange data with HTTP-enabled servers using the Hyper Text Transfer Protocol (HTTP) and Hyper Text Transfer Protocol Secure (HTTPS)... Learn more about this connector.

Connection: Choose...

Action: GET

Operation: Choose...



REST Client

Use the REST Client connector to connect to any service utilizing a REST API. With this connector, you can extract and exchange data with HTTP-enabled servers using the Hyper Text Transfer Protocol (HTTP) and Hyper Text Transfer Protocol Secure (HTTPS)... Learn more about this connector.

Choose...

GET

REST Client Connector Actions

GET — Retrieve information about an existing resource from the server.

POST — Request that a web server accept the data enclosed in the body of the request message.

PUT — Replace the resource at the current URL on the server with the resource contained within the request.

DELETE — Delete an existing resource from the server.

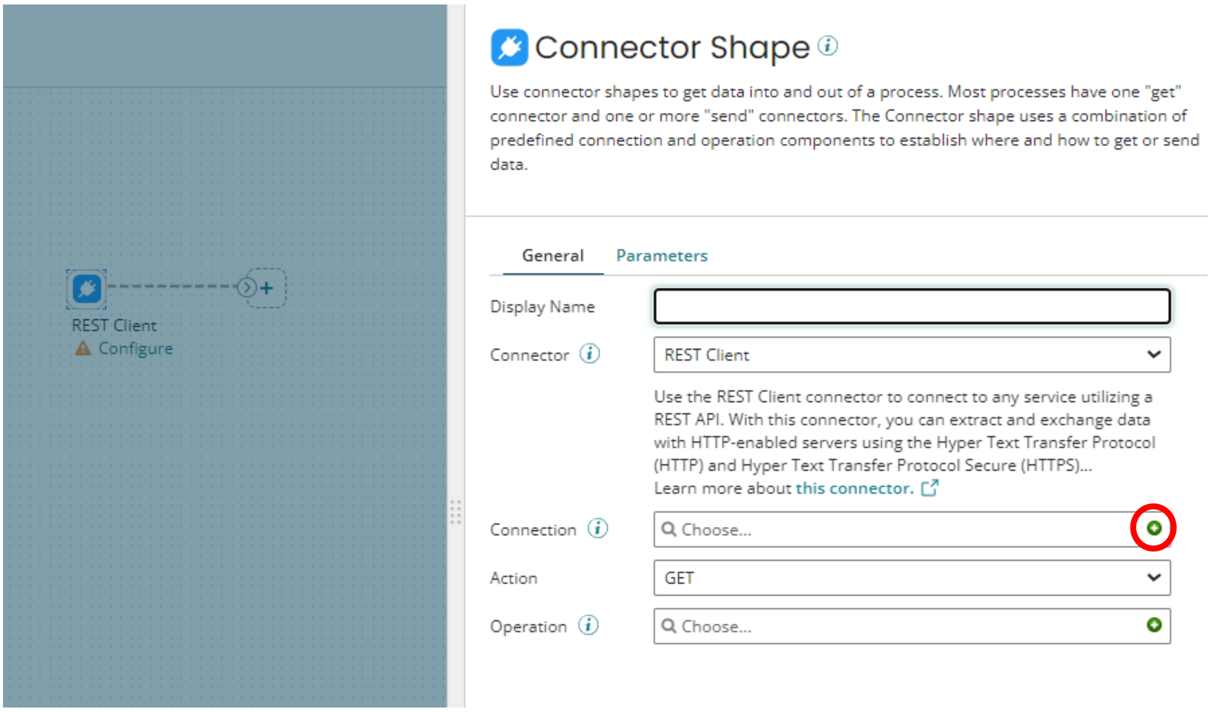
HEAD — Retrieve header information about an existing resource from the server, and not the actual resource itself.

OPTIONS — Retrieve information about the communication options available for an existing resource from the server.

PATCH — Update and make partial changes to an existing resource without replacing the original version of the resource.

TRACE — Perform a message loop-back test along the path to the target resource.

Step 8: Create a Connection Component



Connector Shape

Use connector shapes to get data into and out of a process. Most processes have one "get" connector and one or more "send" connectors. The Connector shape uses a combination of predefined connection and operation components to establish where and how to get or send data.

Display Name:

Connector: REST Client

Use the REST Client connector to connect to any service utilizing a REST API. With this connector, you can extract and exchange data with HTTP-enabled servers using the Hyper Text Transfer Protocol (HTTP) and Hyper Text Transfer Protocol Secure (HTTPS)...
Learn more about this connector.

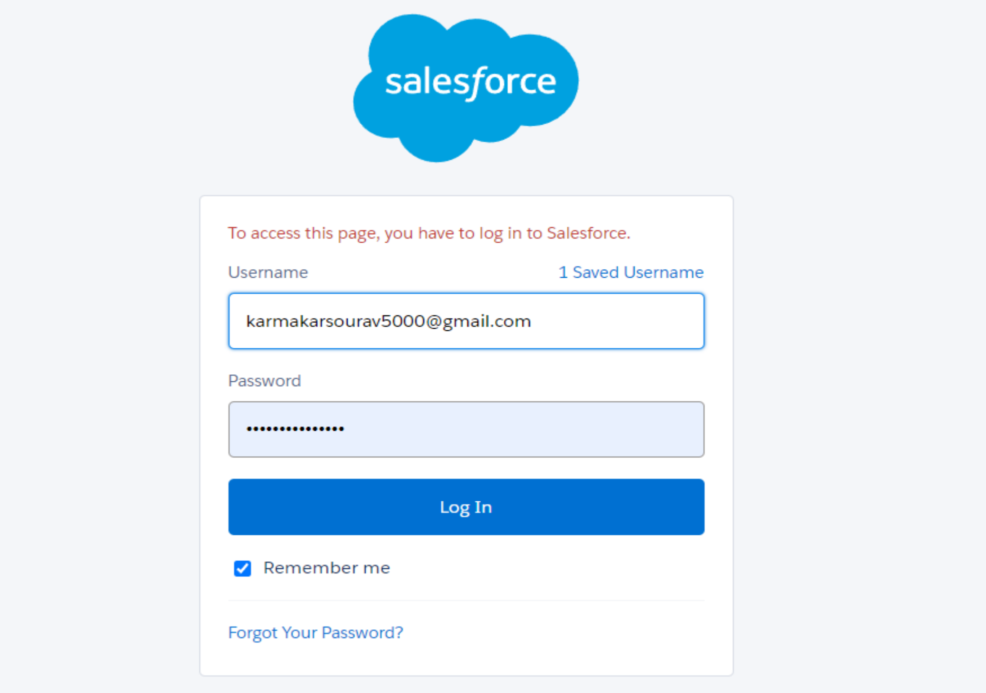
Connection: Choose...

Action: GET

Operation: Choose...

To establish a connection with Salesforce APIs we need to know about Salesforce

Step 9: Log in to Salesforce https://login.salesforce.com



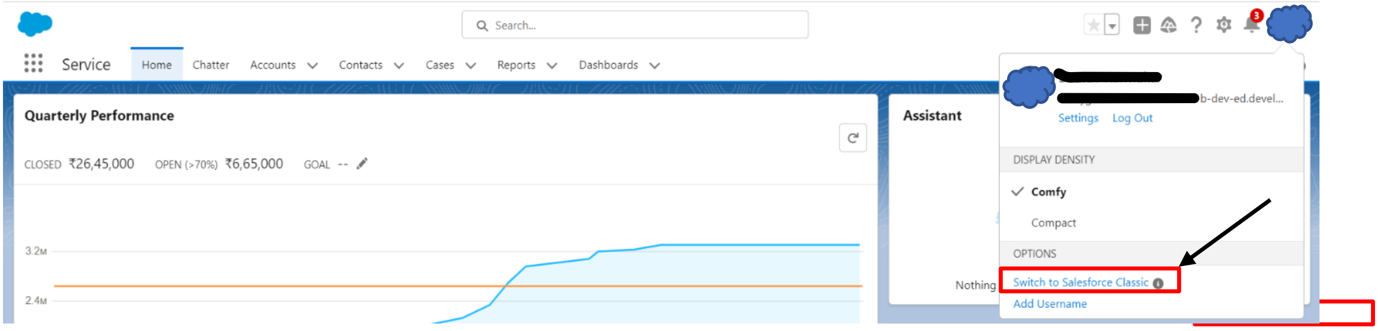
To access this page, you have to log in to Salesforce.

Username: karmakarsourav5000@gmail.com

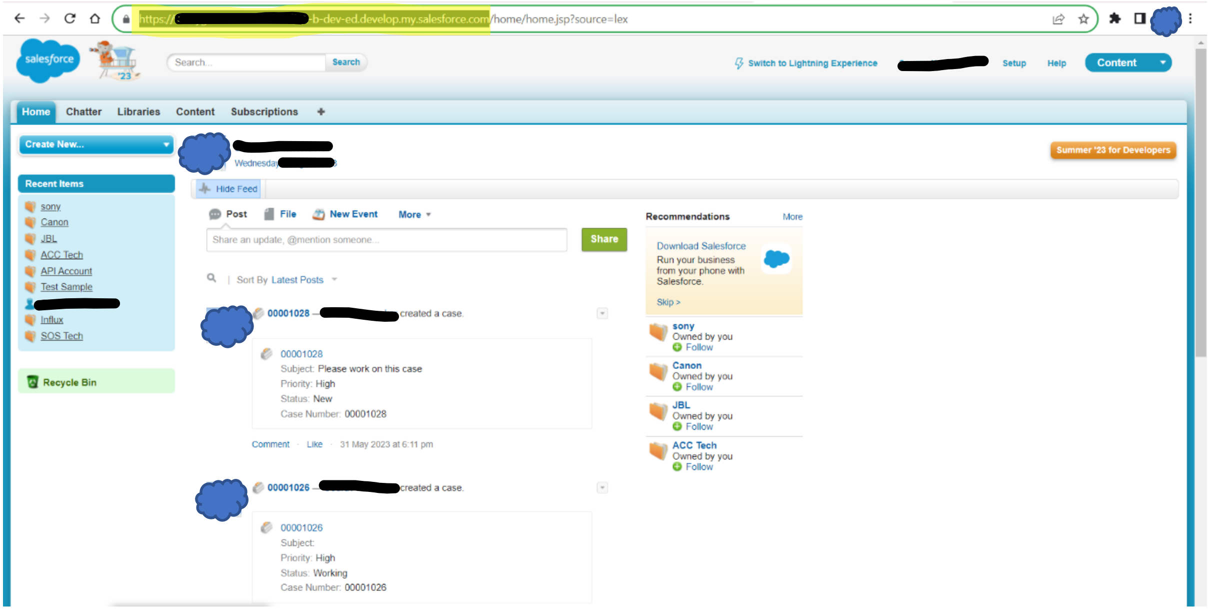
Password: •••••••••••••••

☑ Remember me

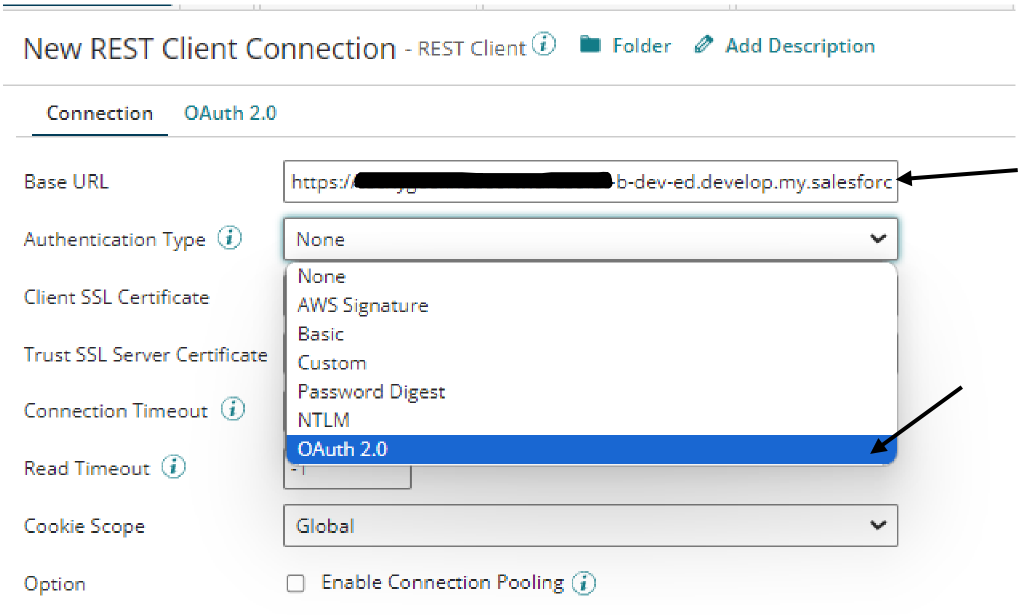
Step 10: Click Switch to Salesforce Classic



Step 11: Copy the Base URL of Salesforce



Step 12: Paste the Base URL in the Connection Component Base URL box and select the Authentication Type OAuth 2.0



New REST Client Connection - REST Client ⓘ Folder Add Description

Connection | OAuth 2.0

Base URL: https://[redacted]b-dev-ed.develop.my.salesforc

Authentication Type ⓘ: None

Client SSL Certificate

Trust SSL Server Certificate

Connection Timeout ⓘ

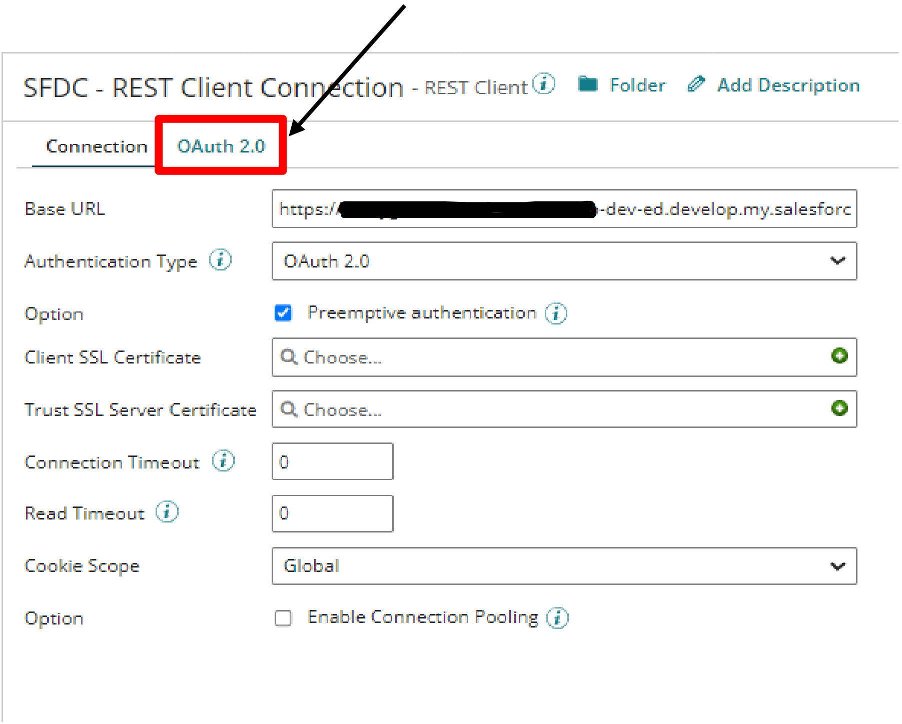
Read Timeout ⓘ: -1

Cookie Scope: Global

Option: ☐ Enable Connection Pooling ⓘ

Base URL from Salesforce

Step 13: Click on the OAuth 2.0 tab



SFDC - REST Client Connection - REST Client ⓘ Folder Add Description

Connection | OAuth 2.0

Base URL: https://[redacted]-dev-ed.develop.my.salesforc

Authentication Type ⓘ: OAuth 2.0

Option: ☑ Preemptive authentication ⓘ

Client SSL Certificate: Choose...

Trust SSL Server Certificate: Choose...

Connection Timeout ⓘ: 0

Read Timeout ⓘ: 0

Cookie Scope: Global

Option: ☐ Enable Connection Pooling ⓘ

- **Connection Timeout**

The maximum wait is in milliseconds to establish a connection to a service. A value less than or equal to zero means that the process will wait indefinitely until a connection to the service is made.

- **Read Timeout**

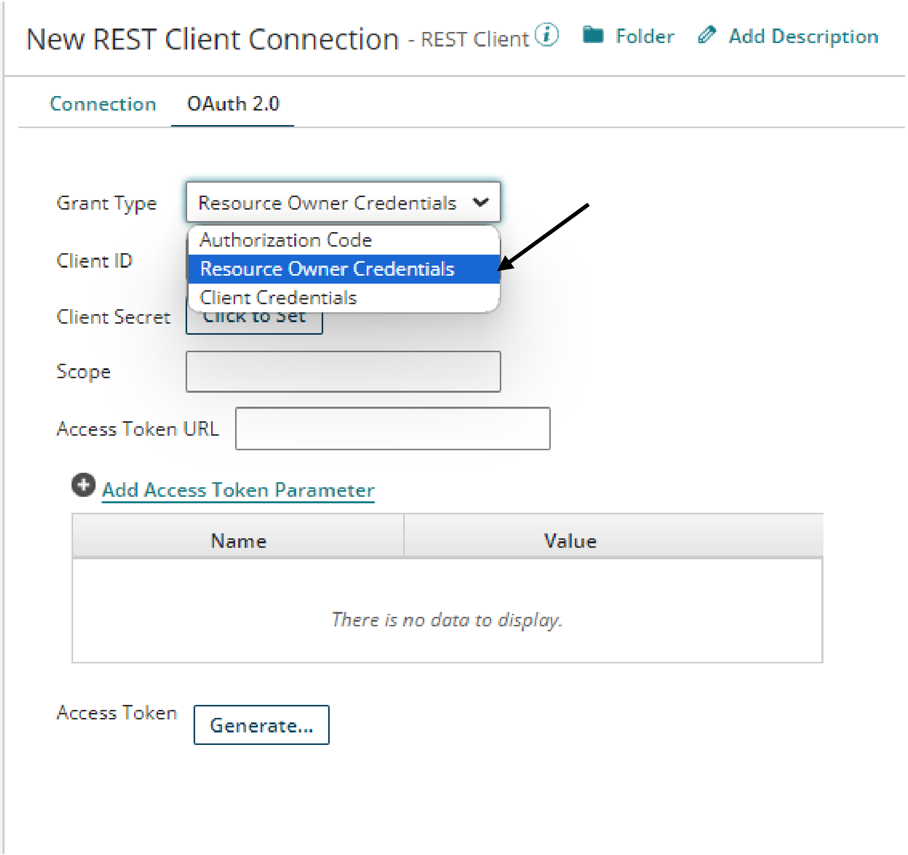
The maximum wait in milliseconds to return all of the data from a service. A value less than or equal to zero means that the process will wait indefinitely until all of the data has been returned from the service.

- **Enable Connection Pooling**

If selected, connection pooling is enabled and connections can be reused when future requests are required. Connection pools may increase performance by using resources more efficiently.

- **Cookie Scope**
  - **Global** — cookies are shared in all processes throughout the Atom JVM life cycle. This is the default.
  - **Connector Shape** — cookies are isolated and shared only for the duration of the Connector Shape execution.
  - **Ignored** — cookies are discarded, even if they are requested by the server.

Step 14: Select the Grant type Resource Owner Credentials



New REST Client Connection - REST Client

Grant Type: Resource Owner Credentials

Name	Value
There is no data to display.	

- **Grant Type**
  - **Authorization Code** — The standard, 3–Legged OAuth2 authorization where you grant the client an authorization code that can be exchanged for an access token.
  - **Resource Owner Credentials** — Requires username and password
  - **Client Credentials** — Uses client credentials to retrieve an access token directly instead of asking for user authorization. This authorization is typically used for administration tasks specific to a client.

Client ID: The client ID is obtained from the application.

Client Secret: The client secret is obtained from the application.

Authorization Token URL (Authorization Code): The endpoint URL to use to obtain an authorization token.

Scope: Add one or more permissions, case-sensitive and separated by a space, to application request URLs. If you change scope permissions, you need to re-authenticate to ensure that all of the requested permissions are granted.

Add Authorization Parameter (Authorization Code): The name and value of extensible endpoint parameters (Optional).

Access Token URL: The endpoint address provided by the application to make API requests.

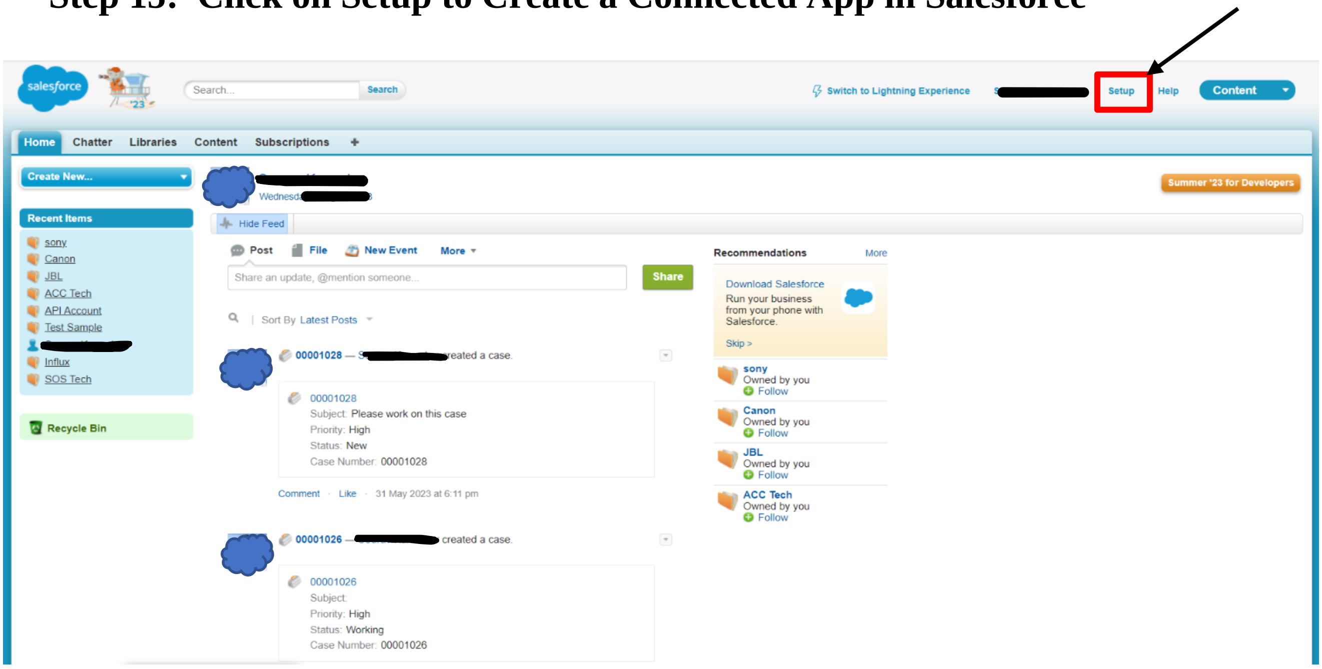
Add Access Token Parameter: The name and value of additional or custom token parameters required by your application (Optional).

Access Token (Authorization Code, Resource Owner Credentials): The encrypted access token retrieved from the application that is used to access protected resources.

Generate (Authorization Code, Resource Owner Credentials): Click to generate the access and refresh tokens. When a new window opens asking you to confirm offline access, click Accept. A second message indicates the access token was received and successfully generated.

To use Salesforce APIs, we need to generate a Client ID and Client Secret

Step 15: Click on Setup to Create a Connected App in Salesforce



00001028 — [redacted] created a case.

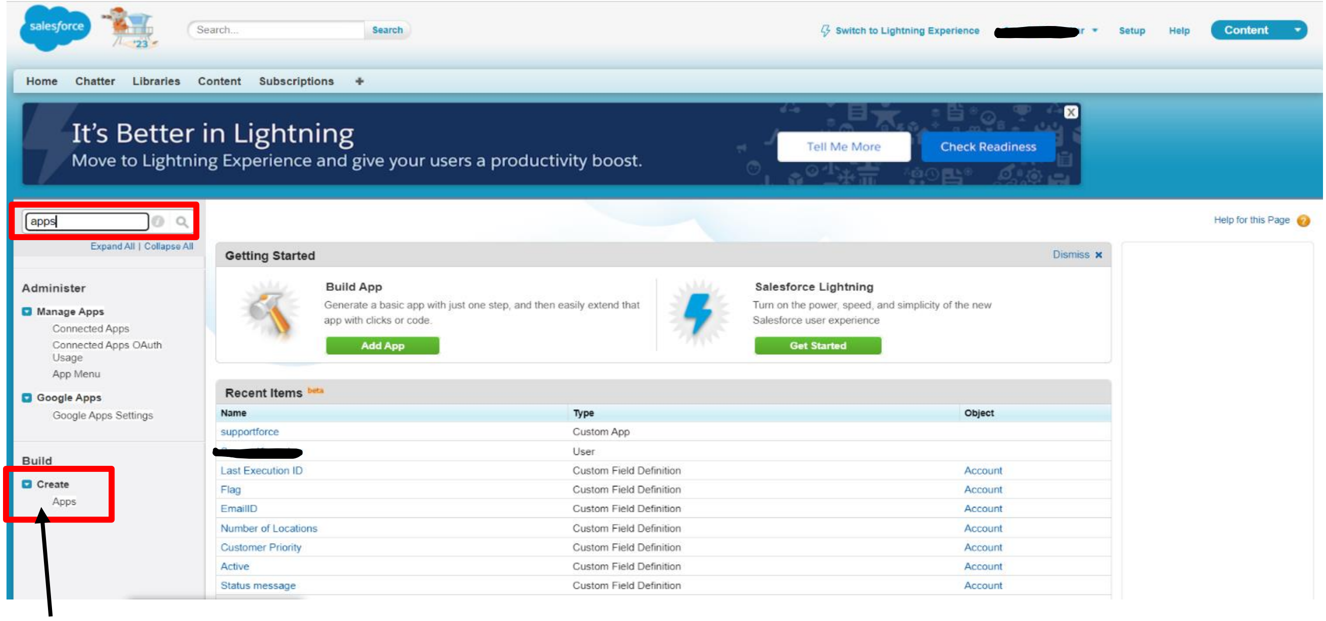
00001028
Subject: Please work on this case
Priority: High
Status: New
Case Number: 00001028

Comment · Like · 31 May 2023 at 6:11 pm

00001026 — [redacted] created a case.

00001026
Subject:
Priority: High
Status: Working
Case Number: 00001026

Step 16: Search Apps in the quick search box

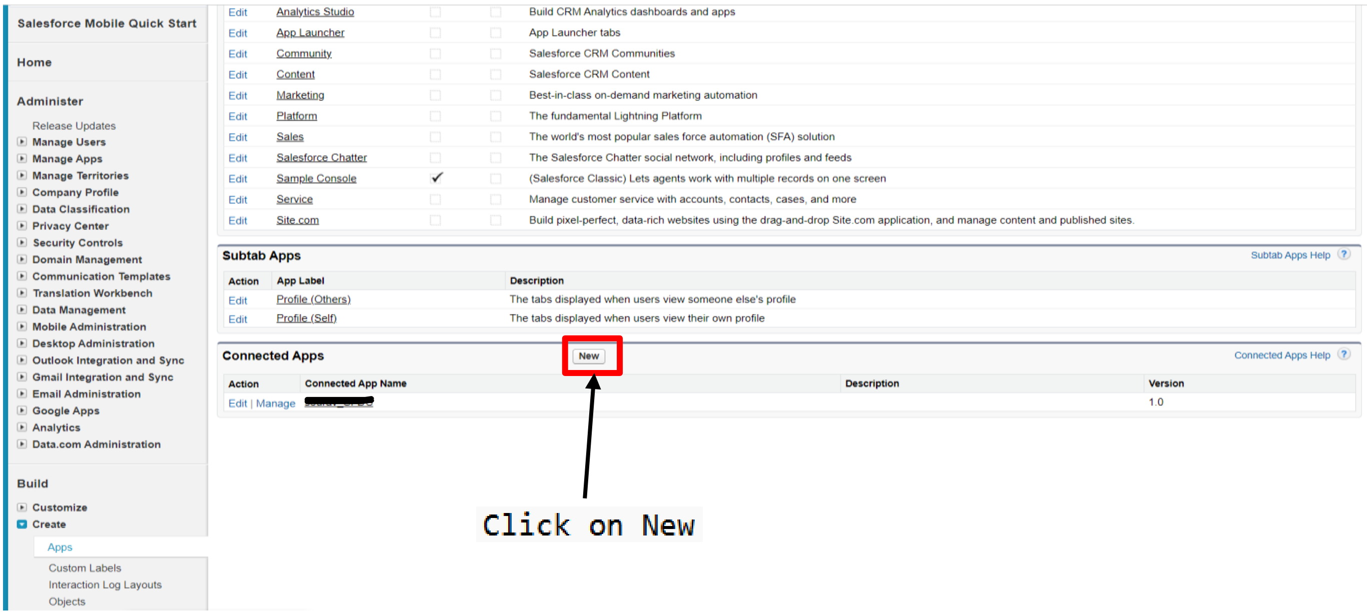


Recent Items beta

Name	Type	Object
supportforce	Custom App	
[redacted]	User	
Last Execution ID	Custom Field Definition	Account
Flag	Custom Field Definition	Account
EmailID	Custom Field Definition	Account
Number of Locations	Custom Field Definition	Account
Customer Priority	Custom Field Definition	Account
Active	Custom Field Definition	Account
Status message	Custom Field Definition	Account

Click on Apps

Step 17: Create a new Connected App



Edit	Analytics Studio	□	□	Build CRM Analytics dashboards and apps
Edit	App Launcher	□	□	App Launcher tabs
Edit	Community	□	□	Salesforce CRM Communities
Edit	Content	□	□	Salesforce CRM Content
Edit	Marketing	□	□	Best-in-class on-demand marketing automation
Edit	Platform	□	□	The fundamental Lightning Platform
Edit	Sales	□	□	The world's most popular sales force automation (SFA) solution
Edit	Salesforce Chatter	□	□	The Salesforce Chatter social network, including profiles and feeds
Edit	Sample Console	✓	□	(Salesforce Classic) Lets agents work with multiple records on one screen
Edit	Service	□	□	Manage customer service with accounts, contacts, cases, and more
Edit	Site.com	□	□	Build pixel-perfect, data-rich websites using the drag-and-drop Site.com application, and manage content and published sites.

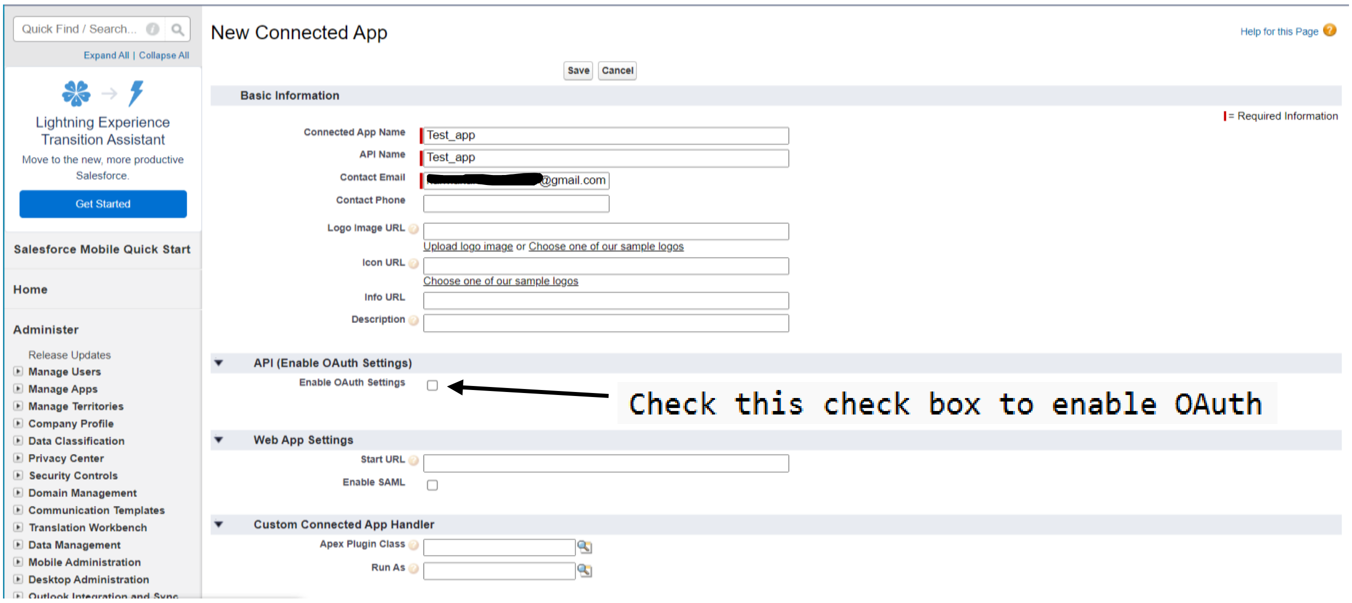
Subtab Apps

Action	App Label	Description
Edit	Profile (Others)	The tabs displayed when users view someone else's profile
Edit	Profile (Self)	The tabs displayed when users view their own profile

Connected Apps

Action	Connected App Name	Description	Version
Edit | Manage	[redacted]		1.0

Step 18: Fill the required fields to Configure the Connected App



Quick Find / Search...

Expand All | Collapse All

Lightning Experience Transition Assistant

Move to the new, more productive Salesforce.

Get Started

Salesforce Mobile Quick Start

Home

Administer

- Release Updates
- Manage Users
- Manage Apps
- Manage Territories
- Company Profile
- Data Classification
- Privacy Center
- Security Controls
- Domain Management
- Communication Templates
- Translation Workbench
- Data Management
- Mobile Administration
- Desktop Administration
- Outlook Integration and Sync

New Connected App

Help for this Page

Save Cancel

Basic Information

| = Required Information

Connected App Name: Test_app

API Name: Test_app

Contact Email: [redacted]@gmail.com

Contact Phone:

Logo Image URL:

Upload logo image or Choose one of our sample logos

Icon URL:

Choose one of our sample logos

Info URL:

Description:

API (Enable OAuth Settings)

Enable OAuth Settings ☐

Check this check box to enable OAuth

Web App Settings

Start URL:

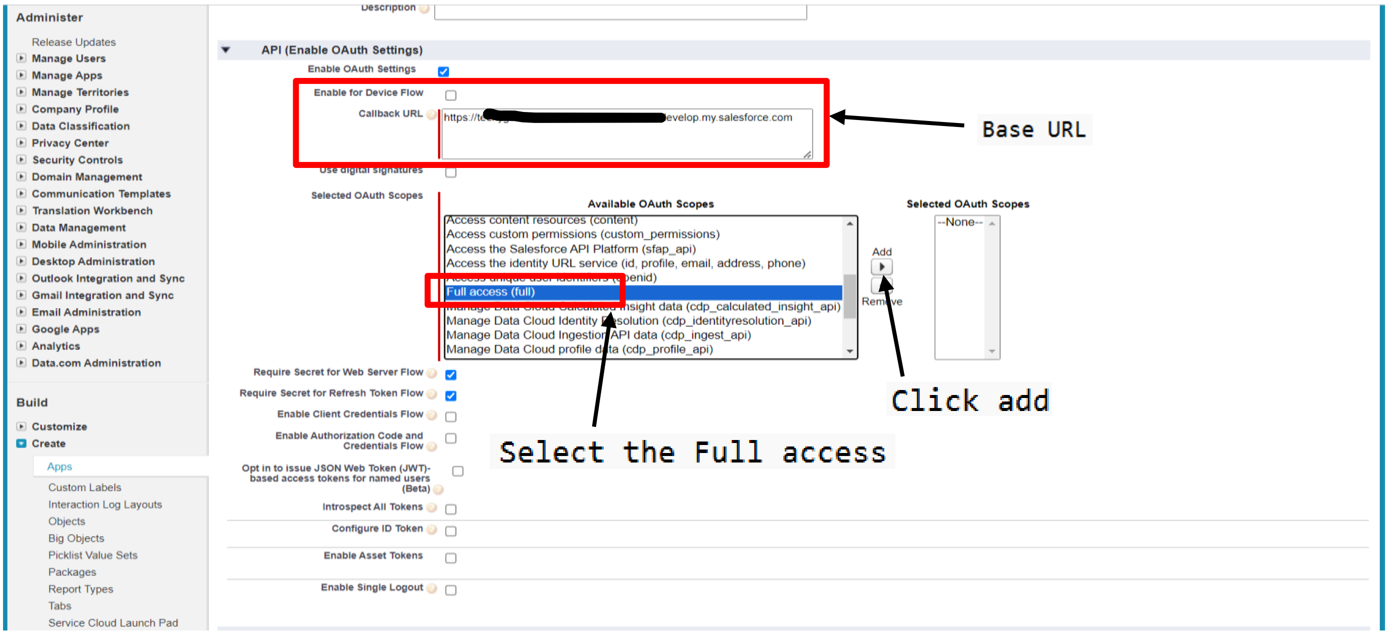
Enable SAML ☐

Custom Connected App Handler

Apex Plugin Class:

Run As:

Step 19: Give the Callback URL and OAuth Scopes



Administer

- Release Updates
- Manage Users
- Manage Apps
- Manage Territories
- Company Profile
- Data Classification
- Privacy Center
- Security Controls
- Domain Management
- Communication Templates
- Translation Workbench
- Data Management
- Mobile Administration
- Desktop Administration
- Outlook Integration and Sync
- Gmail Integration and Sync
- Email Administration
- Google Apps
- Analytics
- Data.com Administration

Build

- Customize
- Create
  - Apps
  - Custom Labels
  - Interaction Log Layouts
  - Objects
  - Big Objects
  - Picklist Value Sets
  - Packages
  - Report Types
  - Tabs
  - Service Cloud Launch Pad

Description

API (Enable OAuth Settings)

Enable OAuth Settings ☑

Enable for Device Flow ☐

Callback URL: https://te[redacted]evelop.my.salesforce.com

Base URL

Use digital signatures ☐

Selected OAuth Scopes

Available OAuth Scopes:

- Access content resources (content)
- Access custom permissions (custom_permissions)
- Access the Salesforce API Platform (sfap_api)
- Access the identity URL service (id, profile, email, address, phone)
- Access unique user identifiers (openid)
- Full access (full)
- Manage Data Cloud Calculated Insight data (cdp_calculated_insight_api)
- Manage Data Cloud Identity Resolution (cdp_identityresolution_api)
- Manage Data Cloud Ingestion API data (cdp_ingest_api)
- Manage Data Cloud profile data (cdp_profile_api)

Add

Remove

Selected OAuth Scopes: --None--

Select the Full access

Click add

Require Secret for Web Server Flow ☑

Require Secret for Refresh Token Flow ☑

Enable Client Credentials Flow ☐

Enable Authorization Code and Credentials Flow ☐

Opt in to issue JSON Web Token (JWT)-based access tokens for named users (Beta) ☐

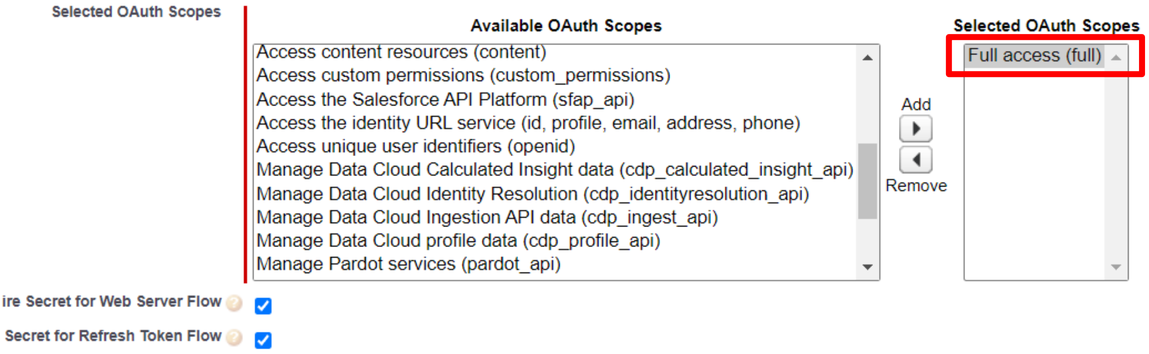
Introspect All Tokens ☐

Configure ID Token ☐

Enable Asset Tokens ☐

Enable Single Logout ☐

Step 20: Add the Scope



Selected OAuth Scopes

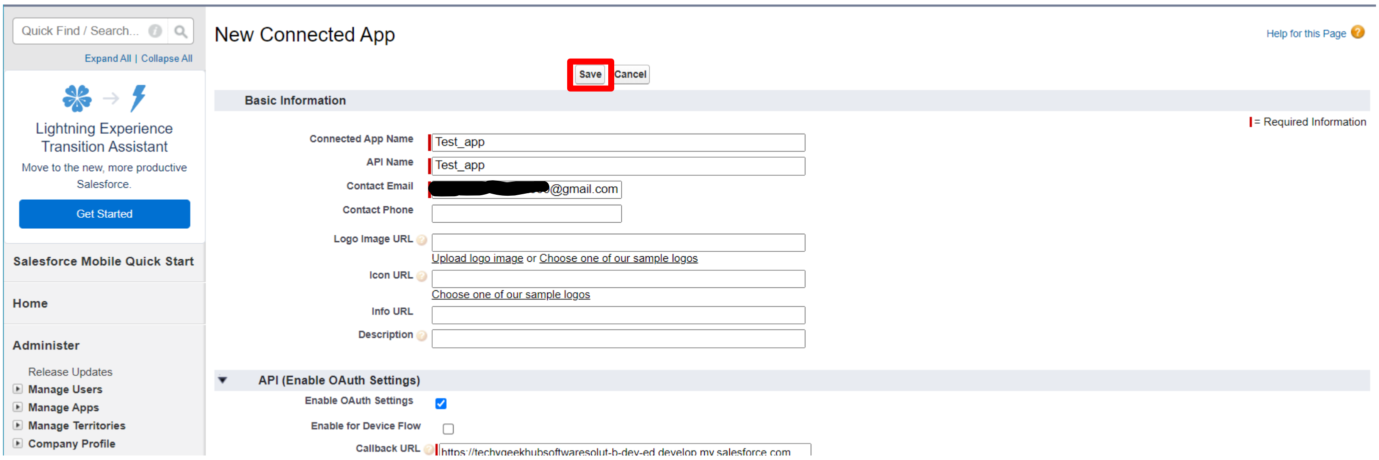
Available OAuth Scopes
- Access content resources (content)
- Access custom permissions (custom_permissions)
- Access the Salesforce API Platform (sfap_api)
- Access the identity URL service (id, profile, email, address, phone)
- Access unique user identifiers (openid)
- Manage Data Cloud Calculated Insight data (cdp_calculated_insight_api)
- Manage Data Cloud Identity Resolution (cdp_identityresolution_api)
- Manage Data Cloud Ingestion API data (cdp_ingest_api)
- Manage Data Cloud profile data (cdp_profile_api)
- Manage Pardot services (pardot_api)

Selected OAuth Scopes
- Full access (full)

ire Secret for Web Server Flow ☑

Secret for Refresh Token Flow ☑

Step 21: Click on save



New Connected App

Basic Information

| = Required Information

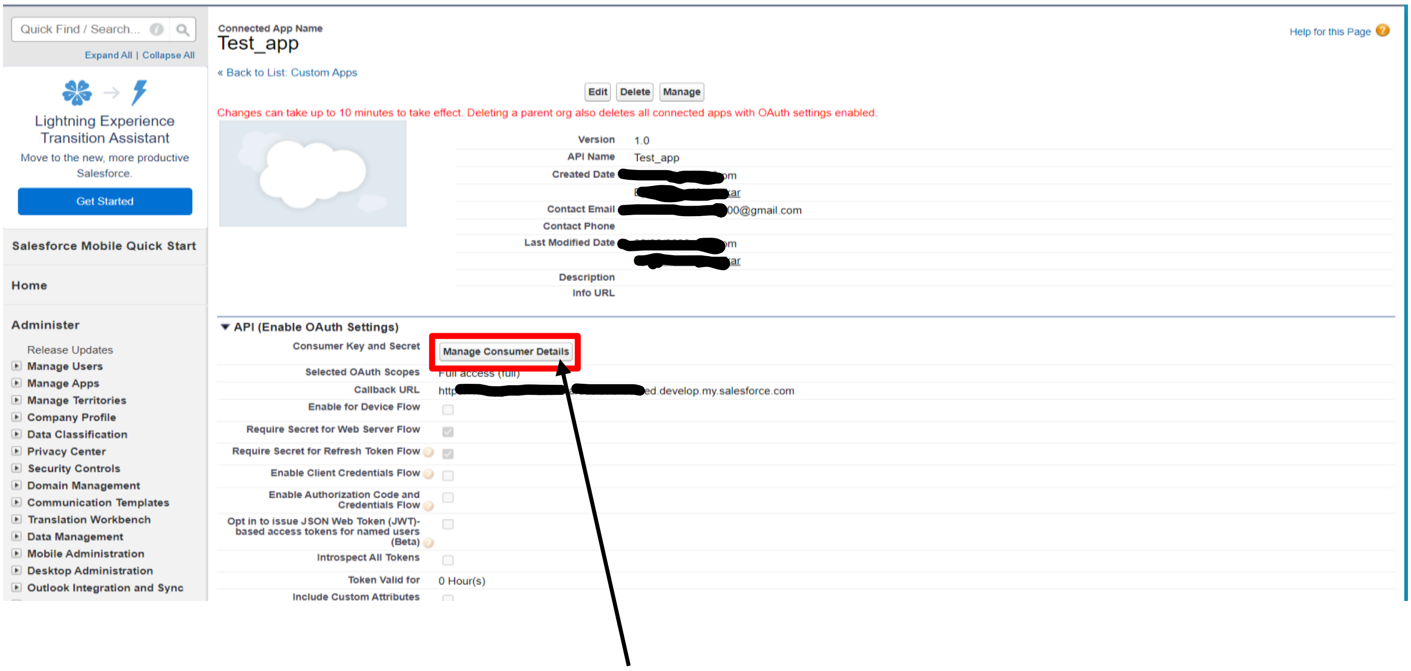
Connected App Name	Test_app
API Name	Test_app
Contact Email	[redacted]@gmail.com
Contact Phone	
Logo Image URL	 Upload logo image or Choose one of our sample logos
Icon URL	 Choose one of our sample logos
Info URL	
Description	

API (Enable OAuth Settings)

Enable OAuth Settings	☑
Enable for Device Flow	☐
Callback URL	https://techygeekhubsoftwaresolut-b-dev-ed.develop.my.salesforce.com

Note: Some time it takes 10 minutes to take effect

Step 22: Click on Manage Customer Details



Connected App Name
Test_app

« Back to List: Custom Apps

Changes can take up to 10 minutes to take effect. Deleting a parent org also deletes all connected apps with OAuth settings enabled.

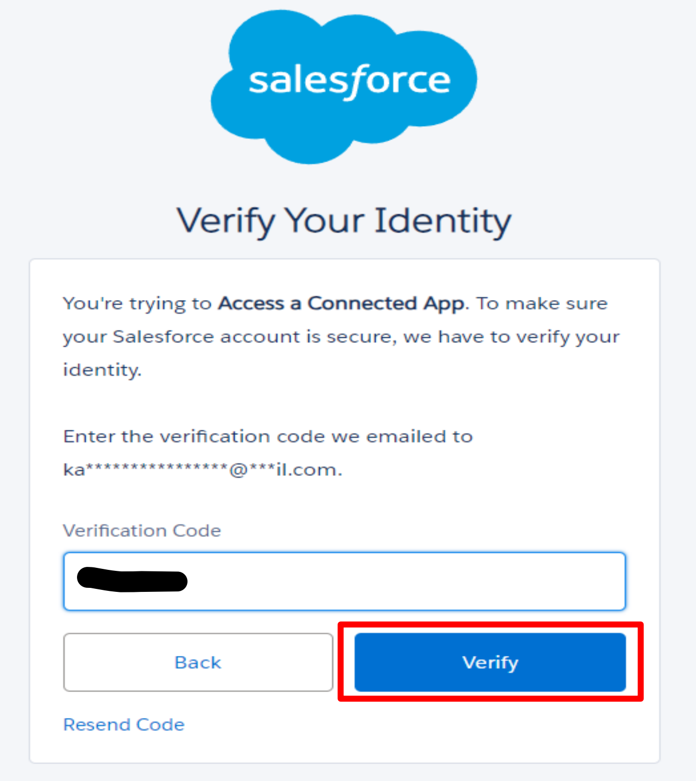
Version	1.0
API Name	Test_app
Created Date	[redacted]
	[redacted]
Contact Email	[redacted]00@gmail.com
Contact Phone	
Last Modified Date	[redacted]
	[redacted]
Description	
Info URL	

▼ API (Enable OAuth Settings)

Consumer Key and Secret	Manage Consumer Details
Selected OAuth Scopes	Full access (full)
Callback URL	http[redacted]ed.develop.my.salesforce.com
Enable for Device Flow	□
Require Secret for Web Server Flow	☑
Require Secret for Refresh Token Flow	☑
Enable Client Credentials Flow	□
Enable Authorization Code and Credentials Flow	□
Opt in to issue JSON Web Token (JWT)-based access tokens for named users (Beta)	□
Introspect All Tokens	□
Token Valid for	0 Hour(s)
Include Custom Attributes	□

- Click this Manage Customer Details Button to generate a Client ID and Client secret.
- One mail will be sent by Salesforce with the verification code to the mail ID that we give at the time of configuring the connected app.

Step 23: Give the 4-digit Verification Code and click Verify



Verify Your Identity

You're trying to **Access a Connected App**. To make sure your Salesforce account is secure, we have to verify your identity.

Enter the verification code we emailed to ka****************@***il.com.

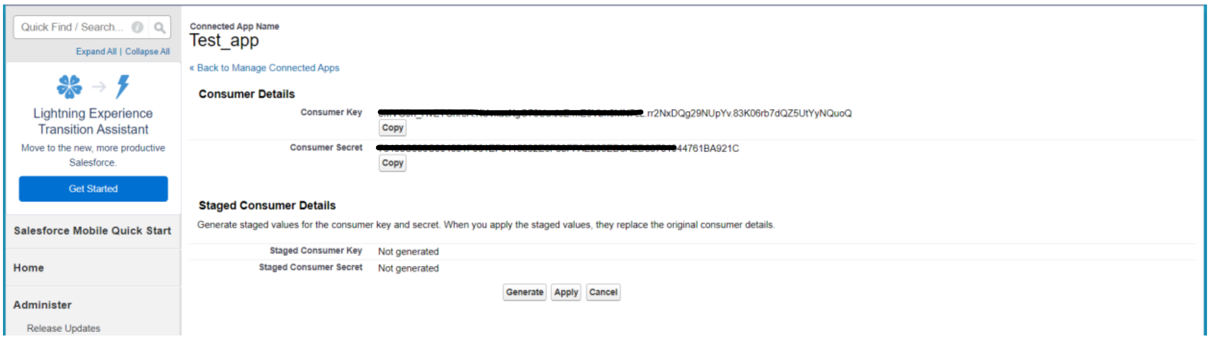
Verification Code

[redacted]

Back | Verify

Resend Code

Step 24: Copy the Consumer Key and Consumer Secret



Quick Find / Search...
Expand All | Collapse All
Lightning Experience Transition Assistant
Move to the new, more productive Salesforce.
Get Started
Salesforce Mobile Quick Start
Home
Administer
Release Updates

Connected App Name
Test_app

« Back to Manage Connected Apps

Consumer Details

Consumer Key [redacted].rr2NxDQg29NUpYv.83K06rb7dQZ5UtYyNQuoQ

Copy

Consumer Secret [redacted]44761BA921C

Copy

Staged Consumer Details

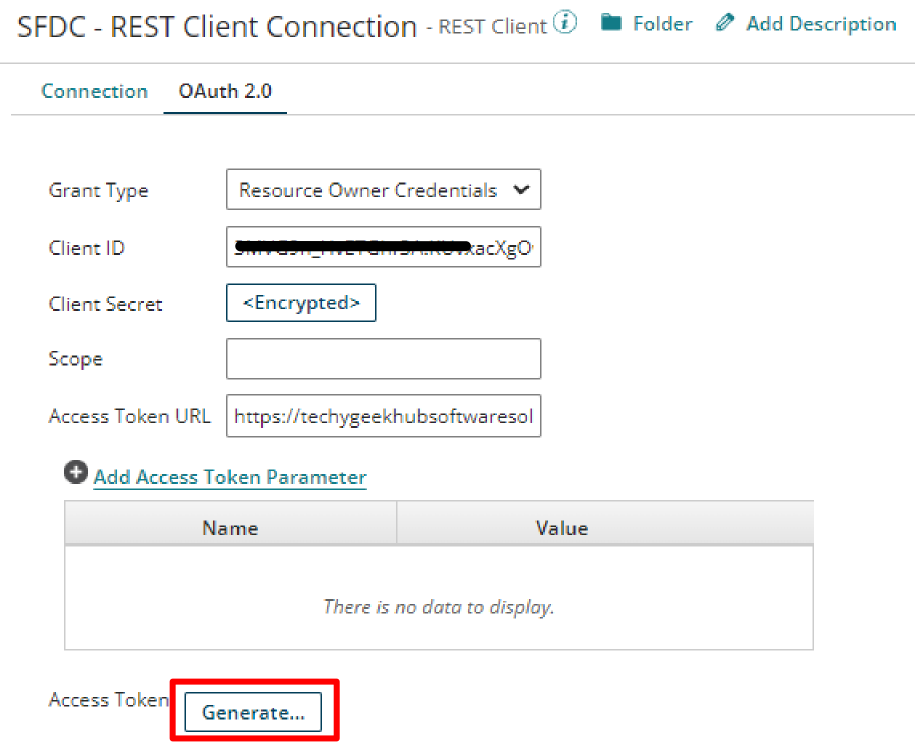
Generate staged values for the consumer key and secret. When you apply the staged values, they replace the original consumer details.

Staged Consumer Key Not generated

Staged Consumer Secret Not generated

- Consumer Key = Client ID
- Consumer Secret = Client Secret

Step 25: Give the Client ID, Client Secret, and Access Token URL



SFDC - REST Client Connection - REST Client
Folder
Add Description
Connection
OAuth 2.0

Grant Type: Resource Owner Credentials

Client ID: [redacted]xacXgO

Client Secret: <Encrypted>

Scope:

Access Token URL: https://techygeekhubsoftwaresol

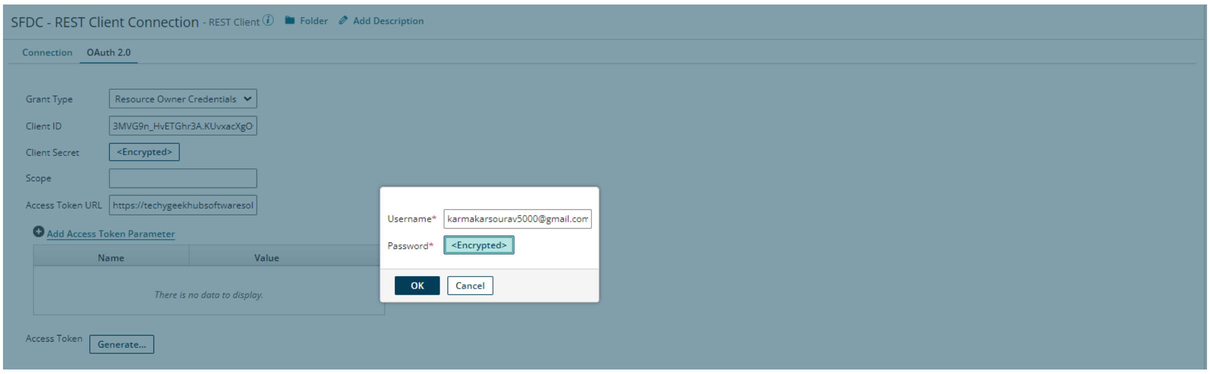
Add Access Token Parameter

Name	Value
There is no data to display.	

`After filling in all the details click this Generate button`

Access Token URL: Base URL + /services/oauth2/token

Step 26: Give Salesforce Username and Password (SFDC Password+token)



SFDC - REST Client Connection - REST Client

Grant Type: Resource Owner Credentials

Client ID: 3MVG9n_HvETGhr3A.KUvxacXgO

Client Secret: <Encrypted>

Scope:

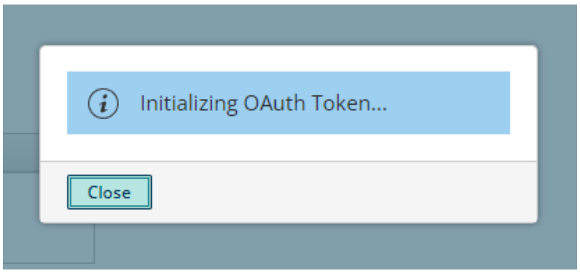
Access Token URL: https://techygeekhubsoftwaresol

There is no data to display.

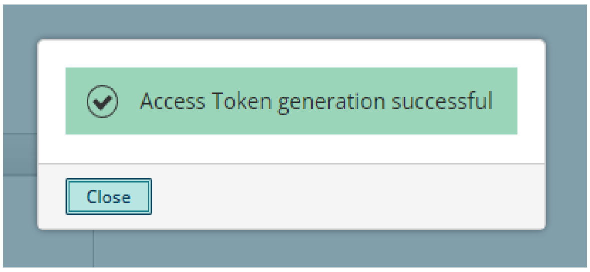
Username*: karmakarsourav5000@gmail.com

Password*: <Encrypted>

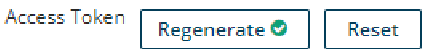
If every detail is right then Access Token will get generate



Initializing OAuth Token...

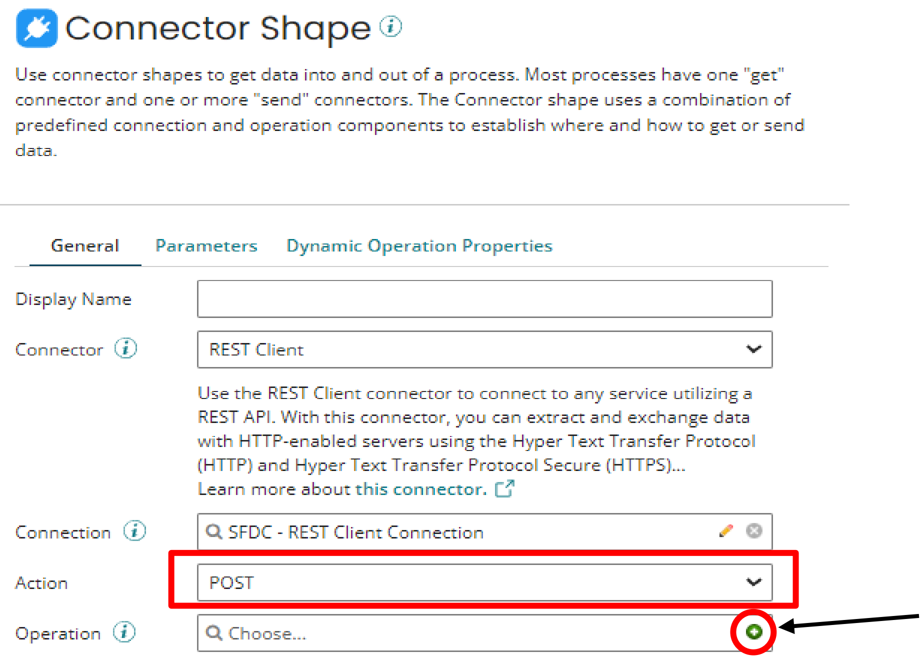


Access Token generation successful



We are able to establish a connection with Salesforce now we need an operation

Step 27: Create an operation and choose the action as POST



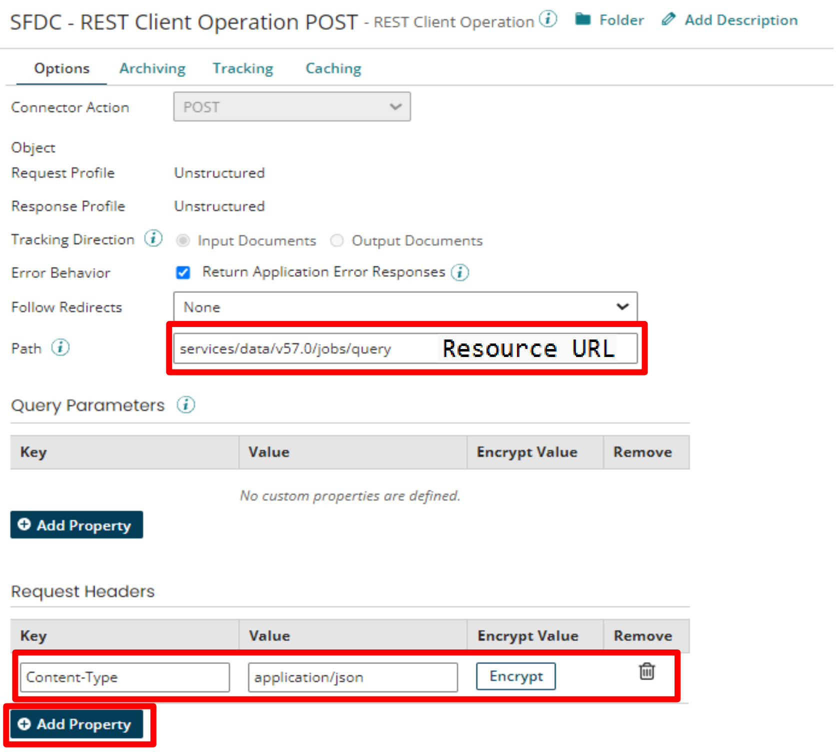
Connector Shape

Use connector shapes to get data into and out of a process. Most processes have one "get" connector and one or more "send" connectors. The Connector shape uses a combination of predefined connection and operation components to establish where and how to get or send data.

General | Parameters | Dynamic Operation Properties

Display Name	
Connector	REST Client
	Use the REST Client connector to connect to any service utilizing a REST API. With this connector, you can extract and exchange data with HTTP-enabled servers using the Hyper Text Transfer Protocol (HTTP) and Hyper Text Transfer Protocol Secure (HTTPS)... Learn more about this connector.
Connection	SFDC - REST Client Connection
Action	POST
Operation	Choose...

Step 28: Give the Resource URL and Content-Type



SFDC - REST Client Operation POST - REST Client Operation   Folder   Add Description

Options | Archiving | Tracking | Caching

Connector Action	POST
Object	
Request Profile	Unstructured
Response Profile	Unstructured
Tracking Direction	● Input Documents ○ Output Documents
Error Behavior	☑ Return Application Error Responses
Follow Redirects	None
Path	services/data/v57.0/jobs/query

Query Parameters

Key	Value	Encrypt Value	Remove
	No custom properties are defined.		

Add Property

Request Headers

Key	Value	Encrypt Value	Remove
Content-Type	application/json	Encrypt	🗑

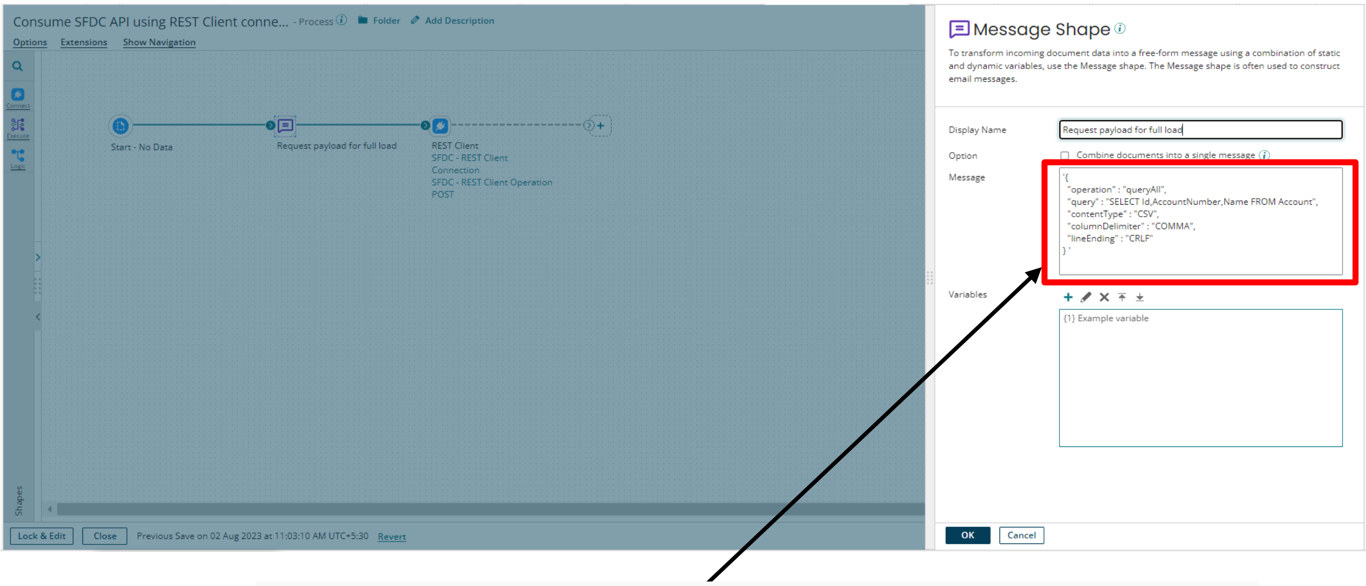
Add Property

- **First, we need to create a job to get the records from Salesforce. And after this API call, we will get a job ID, we need that ID to fetch records from Salesforce.**

Path: services/data/v57.0/jobs/query

We need to add a property to give a Content-Type as application/json. The values are static.

Step 29: Use a Message shape to pass a request file to the API


Consume SFDC API using REST Client conne... - Process
Folder
Add Description
Options
Extensions
Show Navigation
Start - No Data
Request payload for full load
REST Client
SFDC - REST Client Connection
SFDC - REST Client Operation
POST
Shapes
Lock & Edit
Close
Previous Save on 02 Aug 2023 at 11:03:10 AM UTC+5:30
Revert
OK
Cancel

Message Shape

To transform incoming document data into a free-form message using a combination of static and dynamic variables, use the Message shape. The Message shape is often used to construct email messages.

Display Name: Request payload for full load

Option: □ Combine documents into a single message

Message:

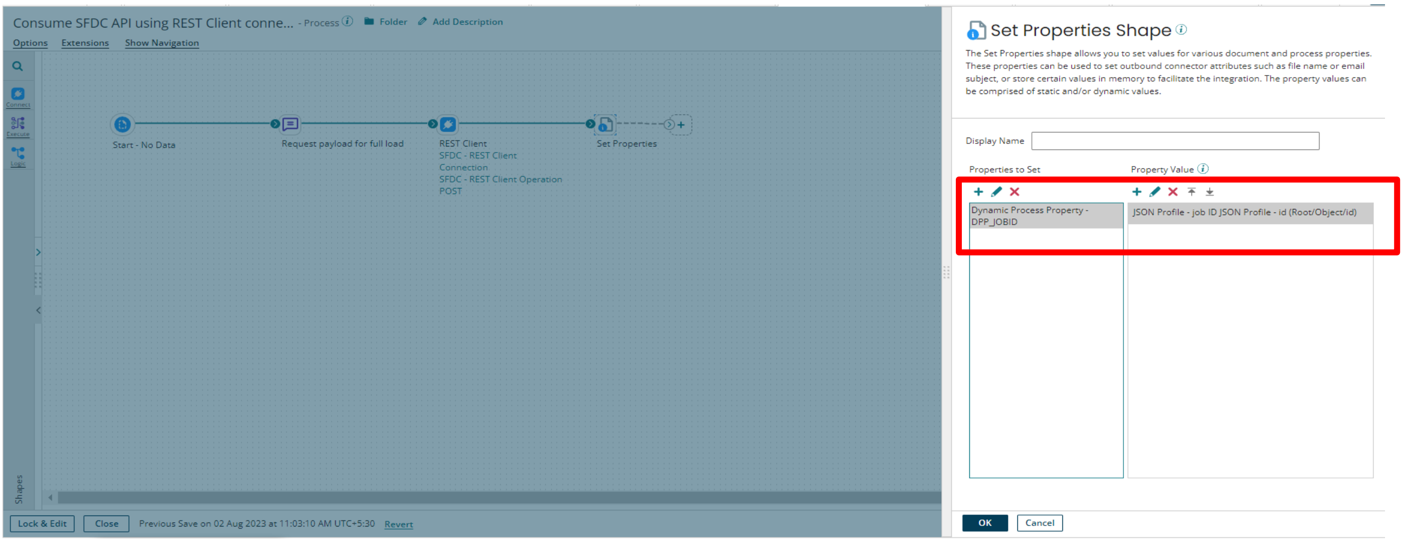
```
'{
 "operation" : "queryAll",
 "query" : "SELECT Id,AccountNumber,Name FROM Account",
 "contentType" : "CSV",
 "columnDelimiter" : "COMMA",
 "lineEnding" : "CRLF"
} '
```

Variables: {1} Example variable

```
'{
  "operation" : "queryAll",
  "query" : "SELECT Id,AccountNumber,Name FROM Account",
  "contentType" : "CSV",
  "columnDelimiter" : "COMMA",
  "lineEnding" : "CRLF"
} '
```

We need ID, Account Number, and Name. If we need more fields, we can mention them in the query.

Step 30: Use a set property shape to capture the job ID from the profile element and store it in a Dynamic Process Property


Consume SFDC API using REST Client conne... - Process
Folder
Add Description
Options
Extensions
Show Navigation
Start - No Data
Request payload for full load
REST Client
SFDC - REST Client Connection
SFDC - REST Client Operation
POST
Set Properties
Shapes
Lock & Edit
Close
Previous Save on 02 Aug 2023 at 11:03:10 AM UTC+5:30
Revert
OK
Cancel

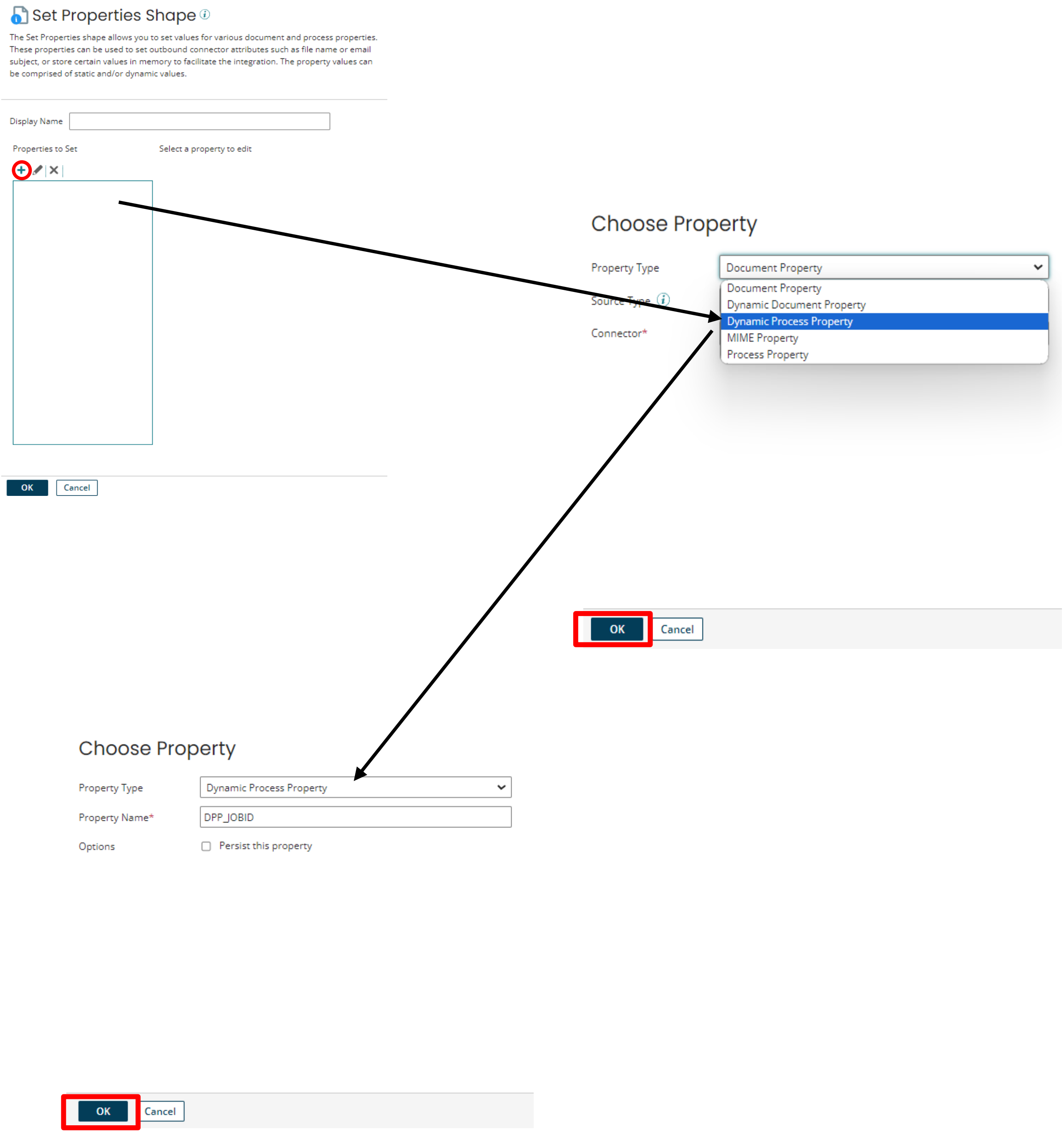
Set Properties Shape

The Set Properties shape allows you to set values for various document and process properties. These properties can be used to set outbound connector attributes such as file name or email subject, or store certain values in memory to facilitate the integration. The property values can be comprised of static and/or dynamic values.

Display Name:

Properties to Set	Property Value
Dynamic Process Property - DPP_JOBID	JSON Profile - job ID JSON Profile - id (Root/Object/id)

Step 31: Set a DPP to store the job ID



Set Properties Shape

The Set Properties shape allows you to set values for various document and process properties. These properties can be used to set outbound connector attributes such as file name or email subject, or store certain values in memory to facilitate the integration. The property values can be comprised of static and/or dynamic values.

Display Name

Properties to Set

Select a property to edit

Choose Property

Property Type: Document Property

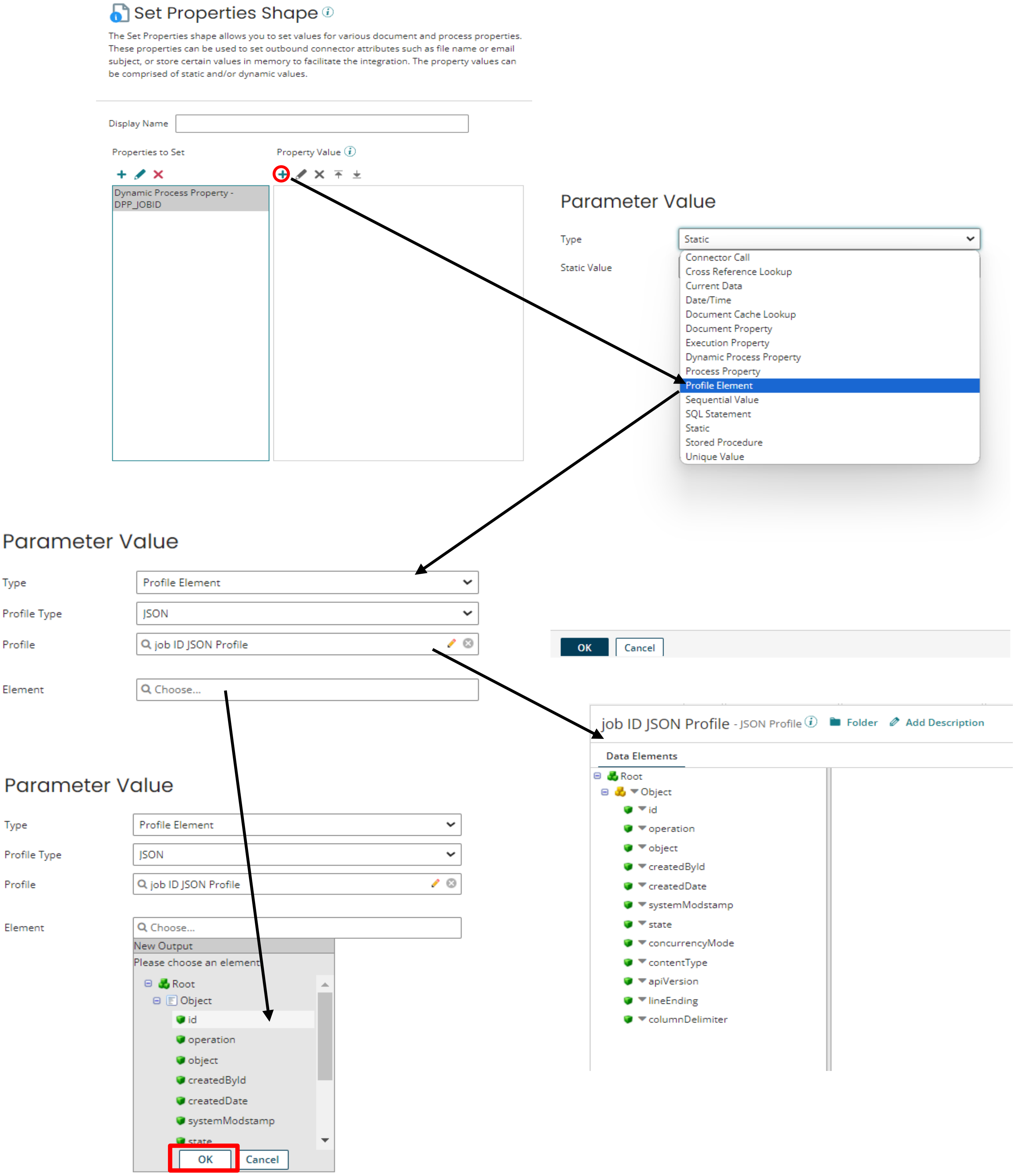
Choose Property

Property Type: Dynamic Process Property

Property Name*: DPP_JOBID

Options: □ Persist this property

Step 32: Set the Parameter Value



Set Properties Shape

The Set Properties shape allows you to set values for various document and process properties. These properties can be used to set outbound connector attributes such as file name or email subject, or store certain values in memory to facilitate the integration. The property values can be comprised of static and/or dynamic values.

Display Name

Properties to Set: Dynamic Process Property - DPP_JOBID

Property Value

Parameter Value

Type: Static

Static Value

Parameter Value

Type: Profile Element

Profile Type: JSON

Profile: job ID JSON Profile

Element: Choose...

job ID JSON Profile - JSON Profile

Parameter Value

Type: Profile Element

Profile Type: JSON

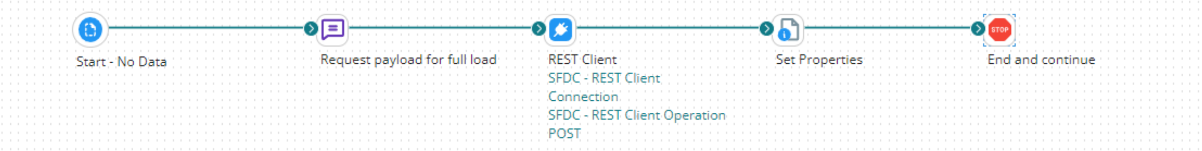
Profile: job ID JSON Profile

Element: Choose...

New Output

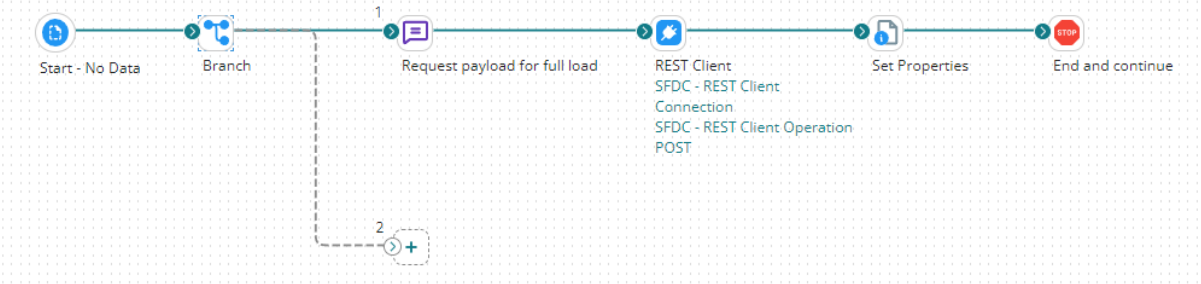
Please choose an element

Step 33: Use a stop shape after the set property shape

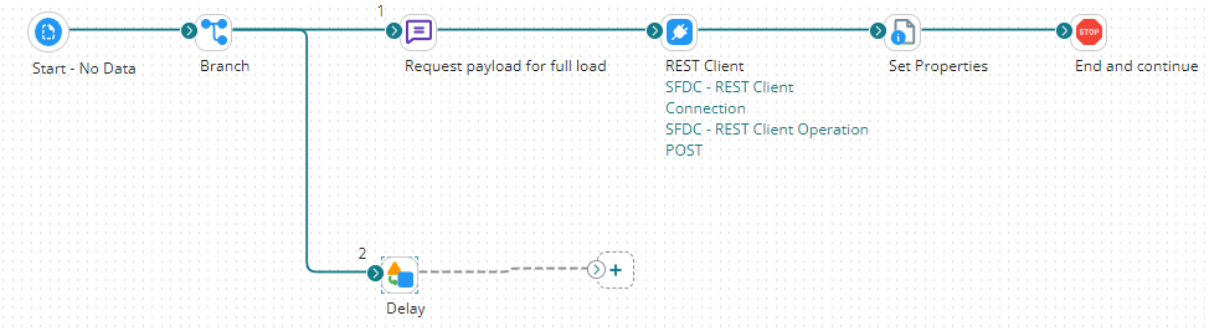

Start - No Data
Request payload for full load
REST Client
SFDC - REST Client Connection
SFDC - REST Client Operation
POST
Set Properties
End and continue

Now we need to get the Records from Salesforce

Step 34: Use a branch shape after the start shape

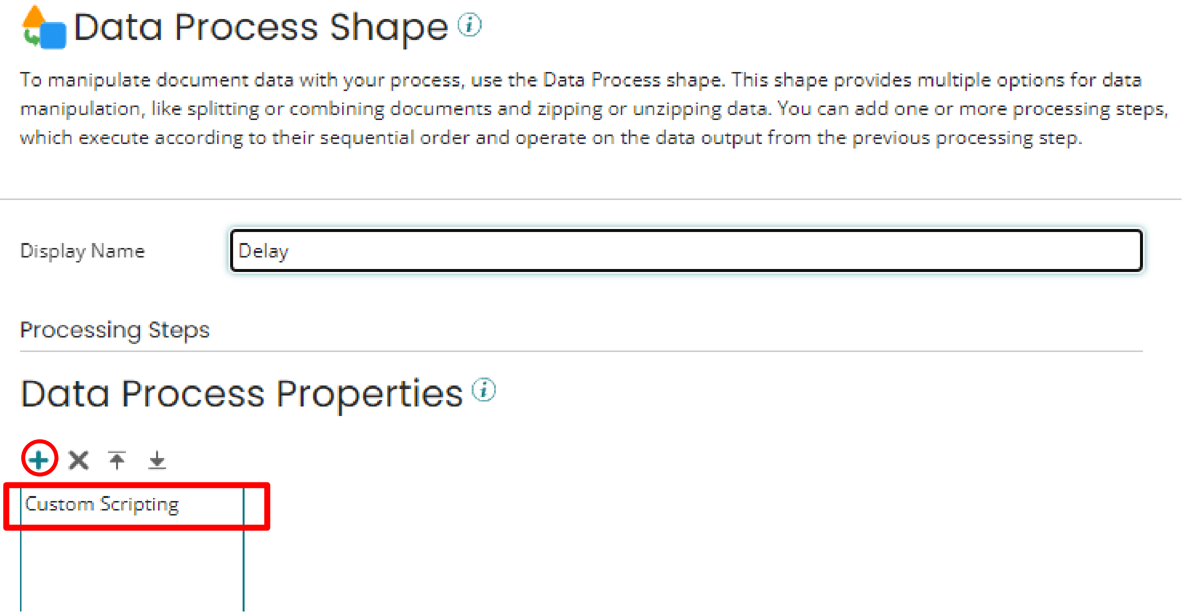

Start - No Data
Branch
1
Request payload for full load
REST Client
SFDC - REST Client Connection
SFDC - REST Client Operation
POST
Set Properties
End and continue
2

Step 35: Use a Data Process shape to give some delay to our process


Start - No Data
Branch
1
Request payload for full load
REST Client
SFDC - REST Client Connection
SFDC - REST Client Operation
POST
Set Properties
End and continue
2
Delay

Note: The job takes some time to complete so we use delay. If we don't use delay sometimes the job will not be complete and we won't get the desired output.

Step 36: Choose the Custom Scripting in Data Process Shape



Data Process Shape

To manipulate document data with your process, use the Data Process shape. This shape provides multiple options for data manipulation, like splitting or combining documents and zipping or unzipping data. You can add one or more processing steps, which execute according to their sequential order and operate on the data output from the previous processing step.

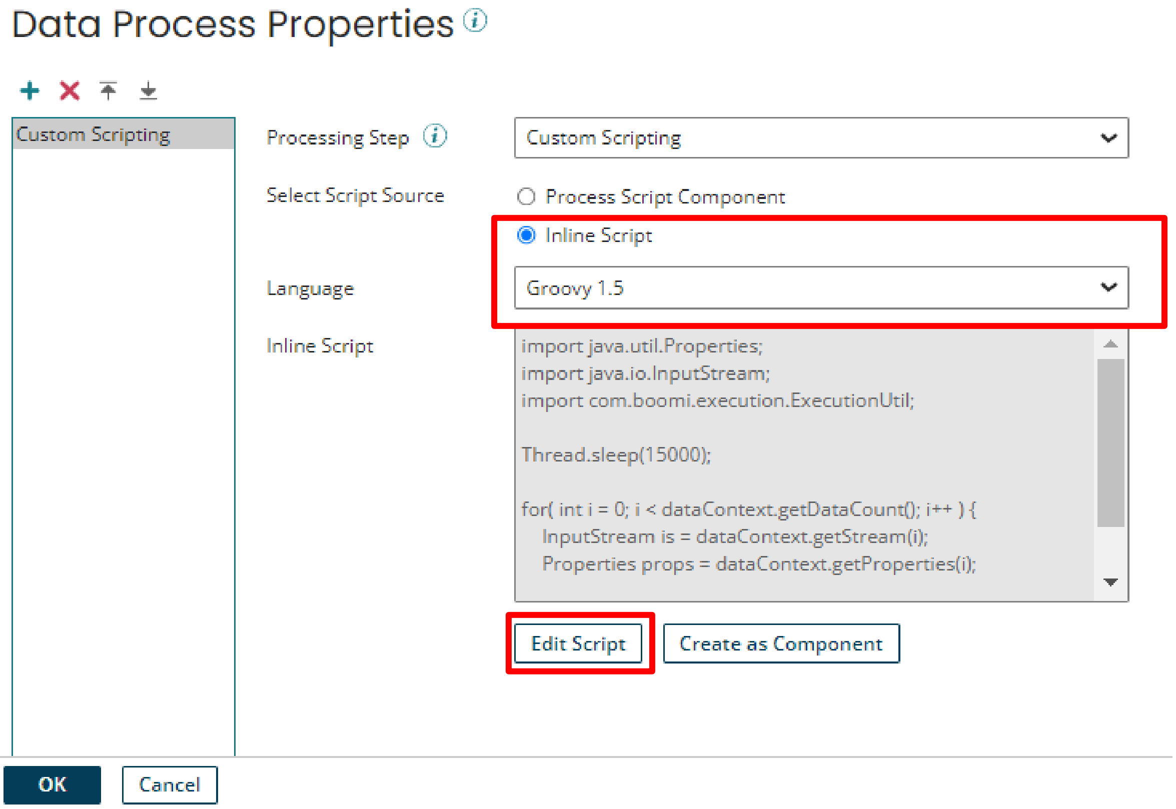
Display Name: Delay

Processing Steps

Data Process Properties

Custom Scripting

Step 37: Need to select Inline Script and Language as Groovy 1.5



Data Process Properties

Custom Scripting

Processing Step: Custom Scripting

Select Script Source: ○ Process Script Component ● Inline Script

Language: Groovy 1.5

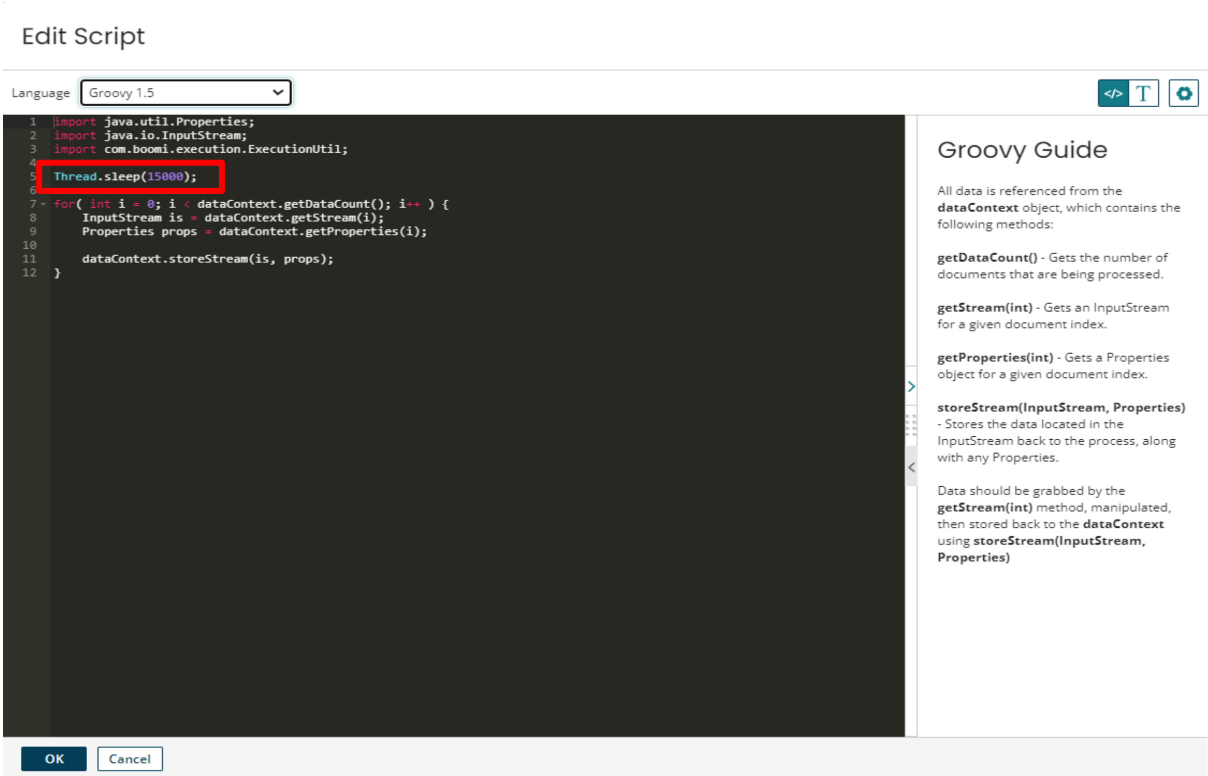
Inline Script:

```
import java.util.Properties;
import java.io.InputStream;
import com.boomi.execution.ExecutionUtil;

Thread.sleep(15000);

for( int i = 0; i < dataContext.getDataCount(); i++ ) {
    InputStream is = dataContext.getStream(i);
    Properties props = dataContext.getProperties(i);
```

Step 38: Script



Language Groovy 1.5

```
import java.util.Properties;
import java.io.InputStream;
import com.boomi.execution.ExecutionUtil;

Thread.sleep(15000);

for( int i = 0; i < dataContext.getDataCount(); i++ ) {
    InputStream is = dataContext.getStream(i);
    Properties props = dataContext.getProperties(i);

    dataContext.storeStream(is, props);
}
```

Groovy Guide

All data is referenced from the **dataContext** object, which contains the following methods:

getDataCount() - Gets the number of documents that are being processed.

getStream(int) - Gets an InputStream for a given document index.

getProperties(int) - Gets a Properties object for a given document index.

storeStream(InputStream, Properties) - Stores the data located in the InputStream back to the process, along with any Properties.

Data should be grabbed by the **getStream(int)** method, manipulated, then stored back to the **dataContext** using **storeStream(InputStream, Properties)**

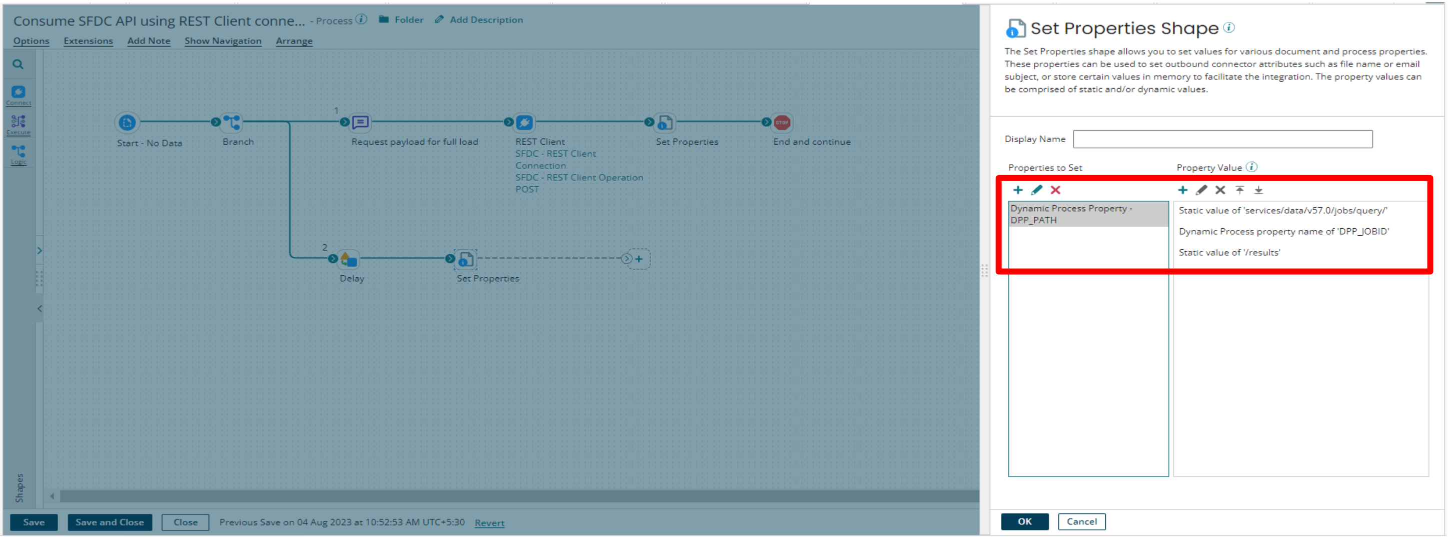
```
import java.util.Properties;
import java.io.InputStream;
import com.boomi.execution.ExecutionUtil;

Thread.sleep(15000);

for( int i = 0; i < dataContext.getDataCount(); i++ ) {
    InputStream is = dataContext.getStream(i);
    Properties props = dataContext.getProperties(i);

    dataContext.storeStream(is, props);
}
```

Step 39: Set a Dynamic process property to create the URL to get the records



Consume SFDC API using REST Client conne... - Process · Folder · Add Description

Previous Save on 04 Aug 2023 at 10:52:53 AM UTC+5:30 Revert

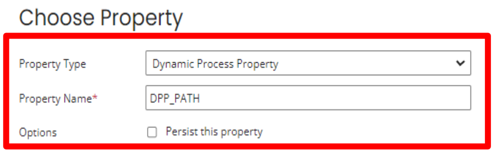
Set Properties Shape

The Set Properties shape allows you to set values for various document and process properties. These properties can be used to set outbound connector attributes such as file name or email subject, or store certain values in memory to facilitate the integration. The property values can be comprised of static and/or dynamic values.

Display Name:

Properties to Set	Property Value
Dynamic Process Property - DPP_PATH	Static value of 'services/data/v57.0/jobs/query/'
	Dynamic Process property name of 'DPP_JOBID'
	Static value of '/results'

Step 40: Set the DPP

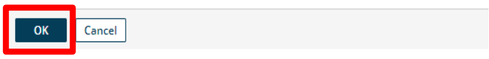


Choose Property

Property Type: Dynamic Process Property

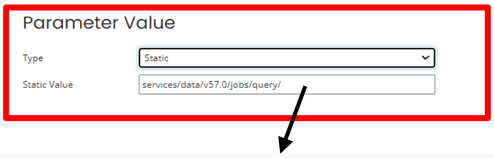
Property Name*: DPP_PATH

Options: ☐ Persist this property



Step 41: Set the Parameter value

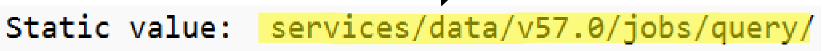
1



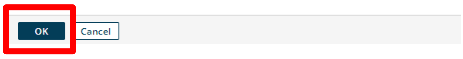
Parameter Value

Type: Static

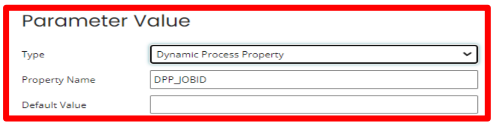
Static Value: services/data/v57.0/jobs/query/



Static value: services/data/v57.0/jobs/query/



2

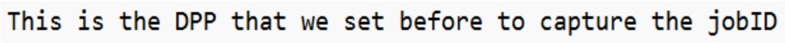


Parameter Value

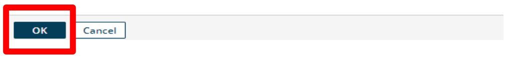
Type: Dynamic Process Property

Property Name: DPP_JOBID

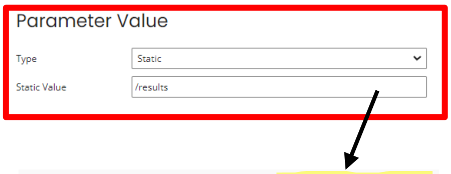
Default Value:



This is the DPP that we set before to capture the jobID



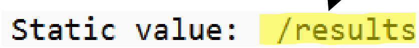
3



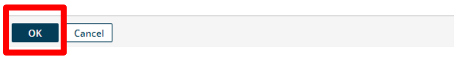
Parameter Value

Type: Static

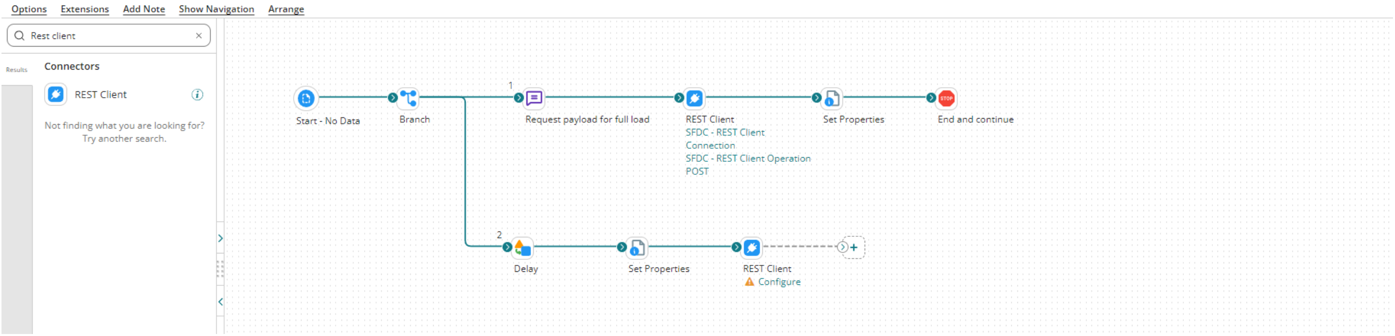
Static Value: /results



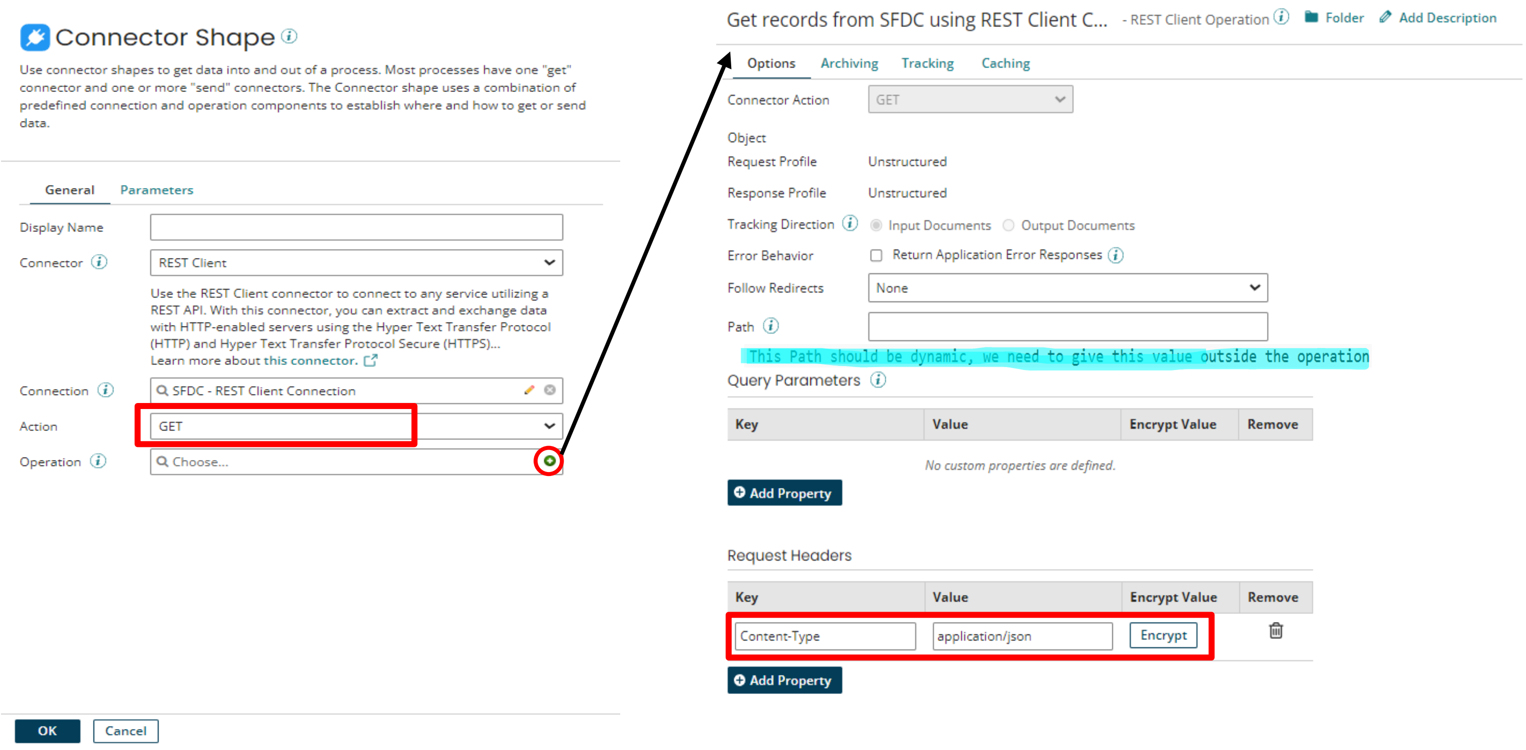
Static value: /results



Step 42: Drag one Rest Client Connector and set it after the Set Property shape



Step 43: Configure the Operation of the Rest Client Connector and the action should be Get also we need to reuse the Connection



Connector Shape

Use connector shapes to get data into and out of a process. Most processes have one "get" connector and one or more "send" connectors. The Connector shape uses a combination of predefined connection and operation components to establish where and how to get or send data.

General | Parameters

Display Name:

Connector: REST Client

Use the REST Client connector to connect to any service utilizing a REST API. With this connector, you can extract and exchange data with HTTP-enabled servers using the Hyper Text Transfer Protocol (HTTP) and Hyper Text Transfer Protocol Secure (HTTPS)... Learn more about this connector.

Connection: SFDC - REST Client Connection

Action: GET

Operation: Choose...

OK | Cancel

Get records from SFDC using REST Client C... - REST Client Operation | Folder | Add Description

Options | Archiving | Tracking | Caching

Connector Action: GET

Object

Request Profile: Unstructured

Response Profile: Unstructured

Tracking Direction: ● Input Documents ○ Output Documents

Error Behavior: □ Return Application Error Responses

Follow Redirects: None

Path:

This Path should be dynamic, we need to give this value outside the operation

Query Parameters

Key	Value	Encrypt Value	Remove

No custom properties are defined.

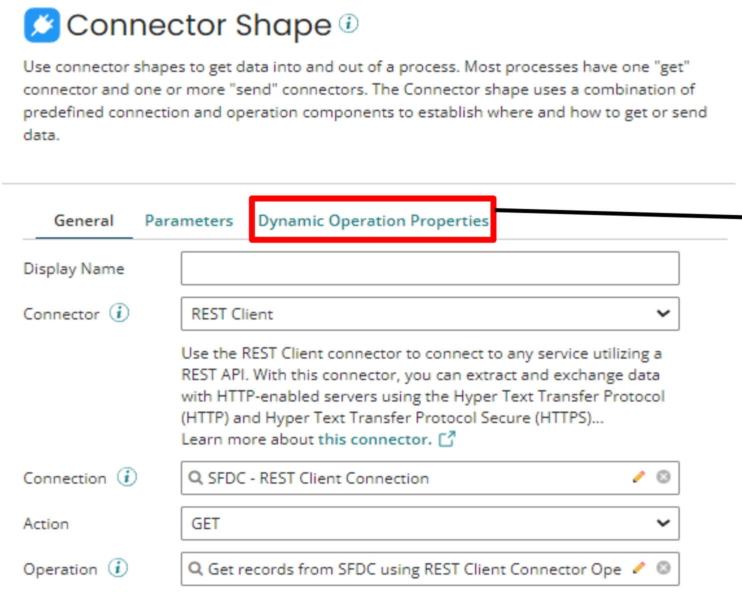
Add Property

Request Headers

Key	Value	Encrypt Value	Remove
Content-Type	application/json	Encrypt	🗑

Add Property

Step 44: Set the path in Dynamic Operation Property



Connector Shape

Use connector shapes to get data into and out of a process. Most processes have one "get" connector and one or more "send" connectors. The Connector shape uses a combination of predefined connection and operation components to establish where and how to get or send data.

General | Parameters | Dynamic Operation Properties

Display Name:

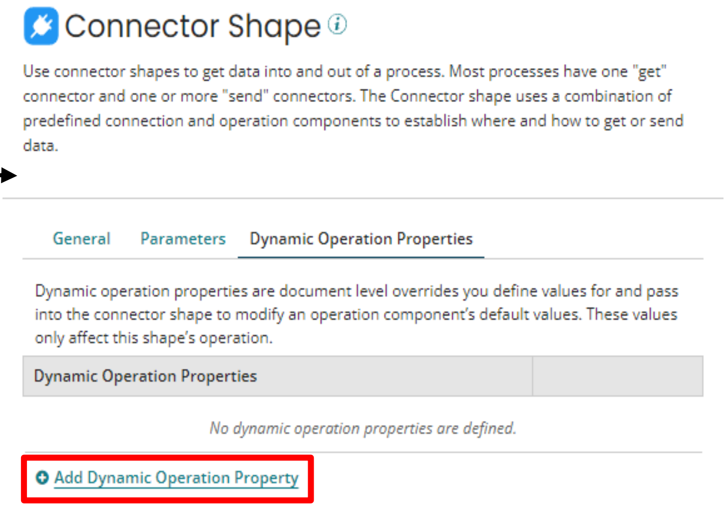
Connector: REST Client

Use the REST Client connector to connect to any service utilizing a REST API. With this connector, you can extract and exchange data with HTTP-enabled servers using the Hyper Text Transfer Protocol (HTTP) and Hyper Text Transfer Protocol Secure (HTTPS)...
Learn more about this connector.

Connection: SFDC - REST Client Connection

Action: GET

Operation: Get records from SFDC using REST Client Connector Ope



Connector Shape

Use connector shapes to get data into and out of a process. Most processes have one "get" connector and one or more "send" connectors. The Connector shape uses a combination of predefined connection and operation components to establish where and how to get or send data.

General | Parameters | Dynamic Operation Properties

Dynamic operation properties are document level overrides you define values for and pass into the connector shape to modify an operation component's default values. These values only affect this shape's operation.

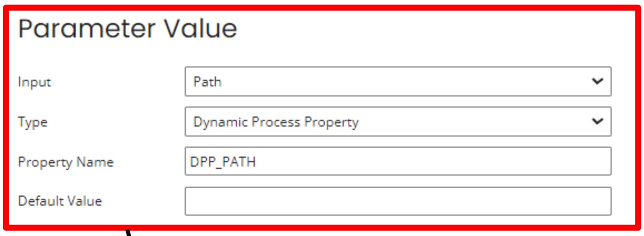
Dynamic Operation Properties

No dynamic operation properties are defined.

Add Dynamic Operation Property

Dynamic Operation Property

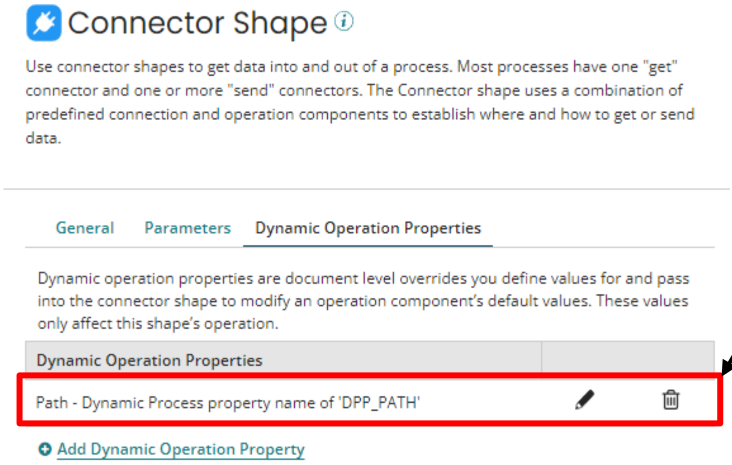
Some connectors enable you to define dynamic operation properties, which are document-level overrides that you define values for, and pass them into the Connector shape to override the operation component's default value. The Connector Shape dialog's Dynamic Operation Properties tab allows you to provide static values or document-level values (Document Property) for the parameters. You can add, edit, and delete the parameters.
The dynamic operation property values defined only affect the specific shape's operation that is selected on the General tab.



Parameter Value

Input	Path
Type	Dynamic Process Property
Property Name	DPP_PATH
Default Value	

Select Input as Path
Type as Dynamic Process Property
Property Name as DPP_PATH



Connector Shape

Use connector shapes to get data into and out of a process. Most processes have one "get" connector and one or more "send" connectors. The Connector shape uses a combination of predefined connection and operation components to establish where and how to get or send data.

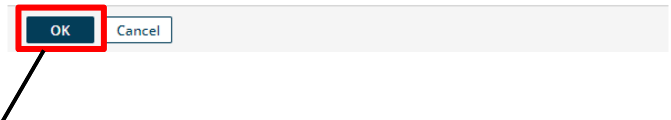
General | Parameters | Dynamic Operation Properties

Dynamic operation properties are document level overrides you define values for and pass into the connector shape to modify an operation component's default values. These values only affect this shape's operation.

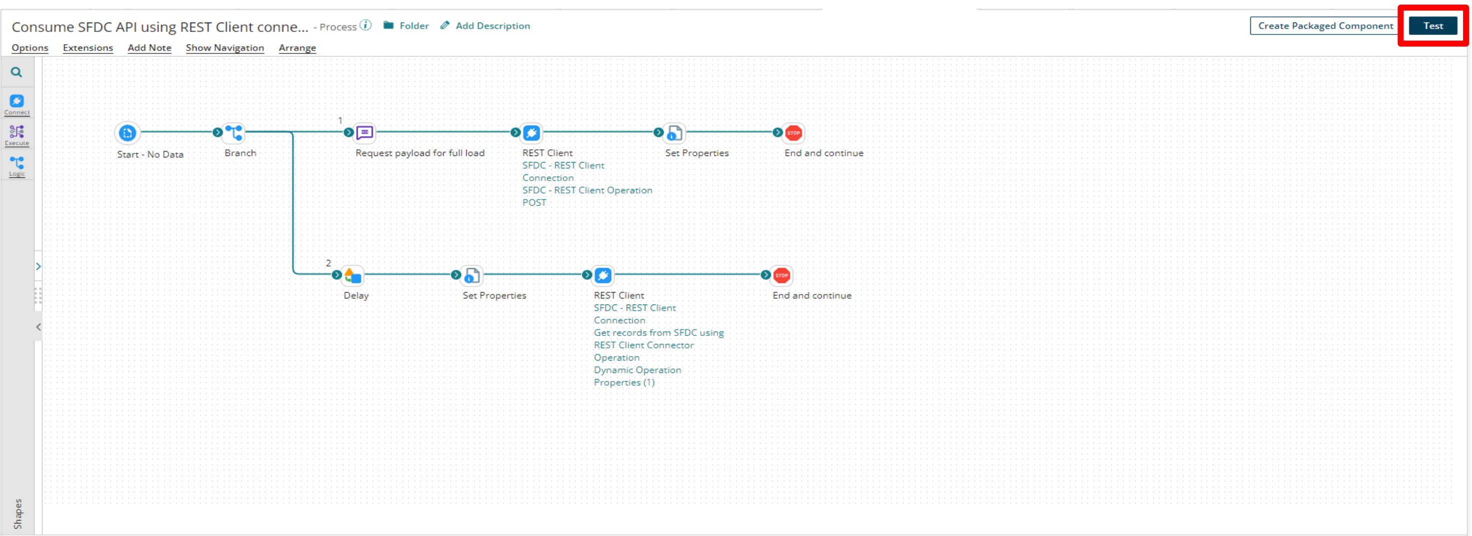
Dynamic Operation Properties

Path - Dynamic Process property name of 'DPP_PATH'

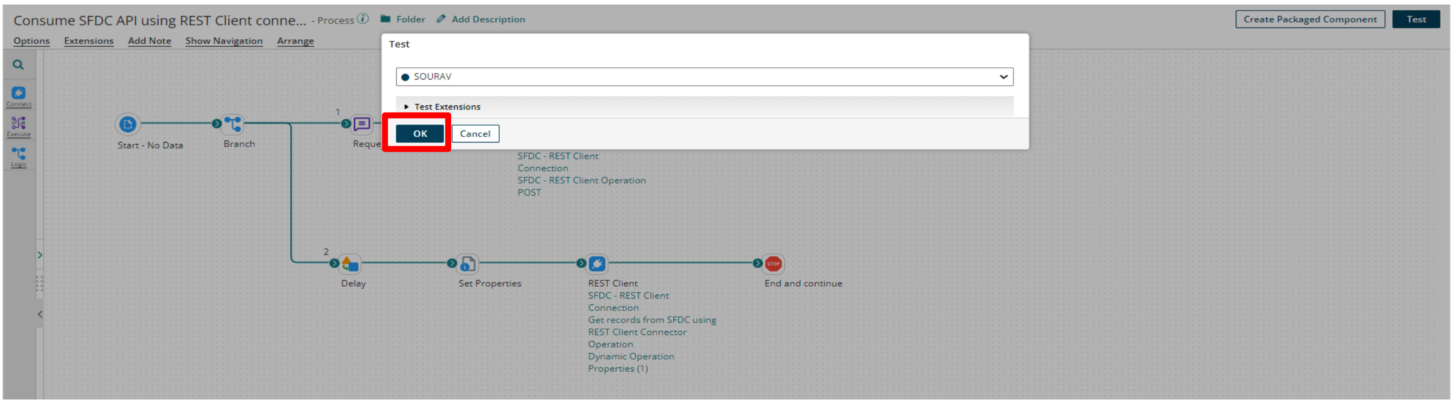
Add Dynamic Operation Property



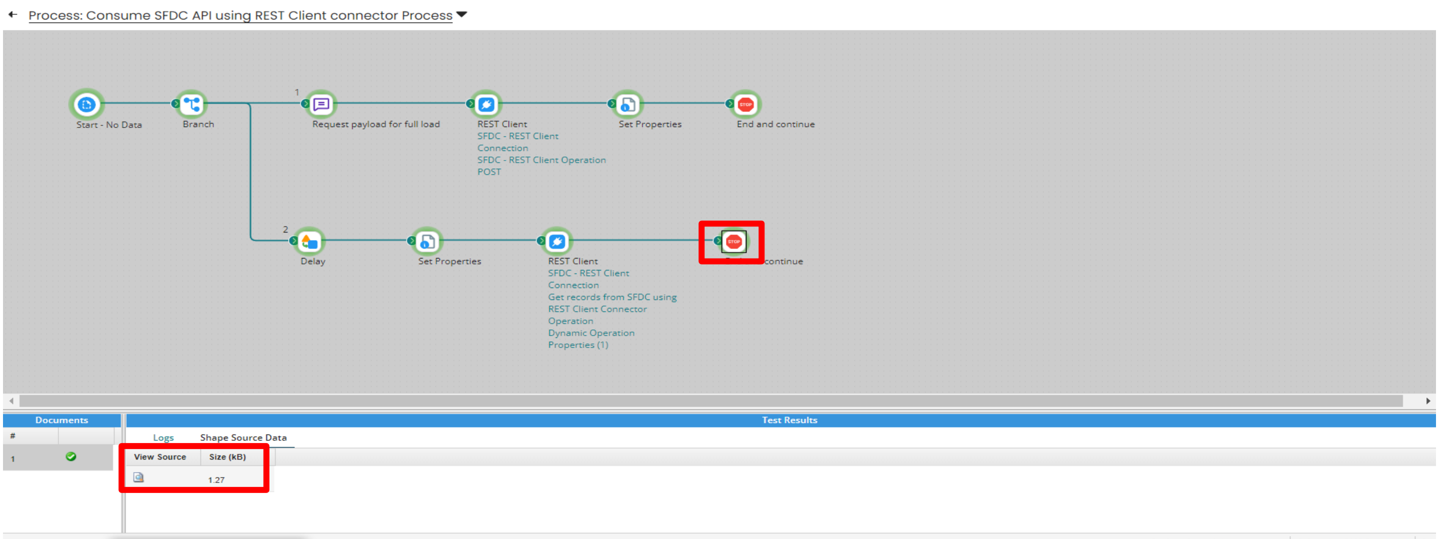
Step 45: Attach a stop shape at the end and click on the test button to test the process



Step 46: Select the Atom and click OK to test the process



Step 47: Go to the stop shape and Shape Source Data and see the response



TGH
Making Integrations Simpler

boomi Partner

TGH Software Solutions Pvt. Ltd.

www.techygeekhub.com

At TGH, we specialize in driving digital transformation through seamless Integration Technologies.

Operating as an INTEGRATION FACTORY, we serve as a one-stop shop for all your integration needs. Our expert team is well-versed in enterprise software and legacy system integration, along with leading iPaaS technologies like Boomi, MuleSoft, Workato, OIC, and more.

We're committed to enhancing business processes and solving problems through our integration expertise.

Email address
connect@techygeekhub.com

Phone number
+ 011-40071137
+ 91-8810610395

Our offices

Noida Office
iThum
Plot No -40, Tower A,
Office No: 712,
Sector-62, Noida,
Uttar Pradesh, 201301

Hyderabad Office
Plot no: 6/3, 5th Floor,
Techno Pearl Building,
HUDA Techno Enclave,
HITEC City, Hyderabad,
Telangana 500081