



# App Connect with RPA

## Hands-on Lab

*Version 1.1*

*15<sup>th</sup> September 2021*

*Authors:*

*Nigel Crowther – [ncrowther@uk.ibm.com](mailto:ncrowther@uk.ibm.com)*

## Table of Contents

1. Introduction .....	3
2. Use case .....	4
3. Prerequisites .....	5
4. Lab Instructions.....	6
Create a Cloudant database.....	6
Provision an App Connect Instance .....	11
Import the RPA Bot API .....	13
Create an App Connect Flow.....	14
Fix Errors in the Flow.....	15
Run the App Connect Flow .....	18
View App Connect Logs.....	18
View Processing Result in Cloudant .....	20

## 1. Introduction

This hands-on lab you will learn to build an App Connect flow to run a bot.

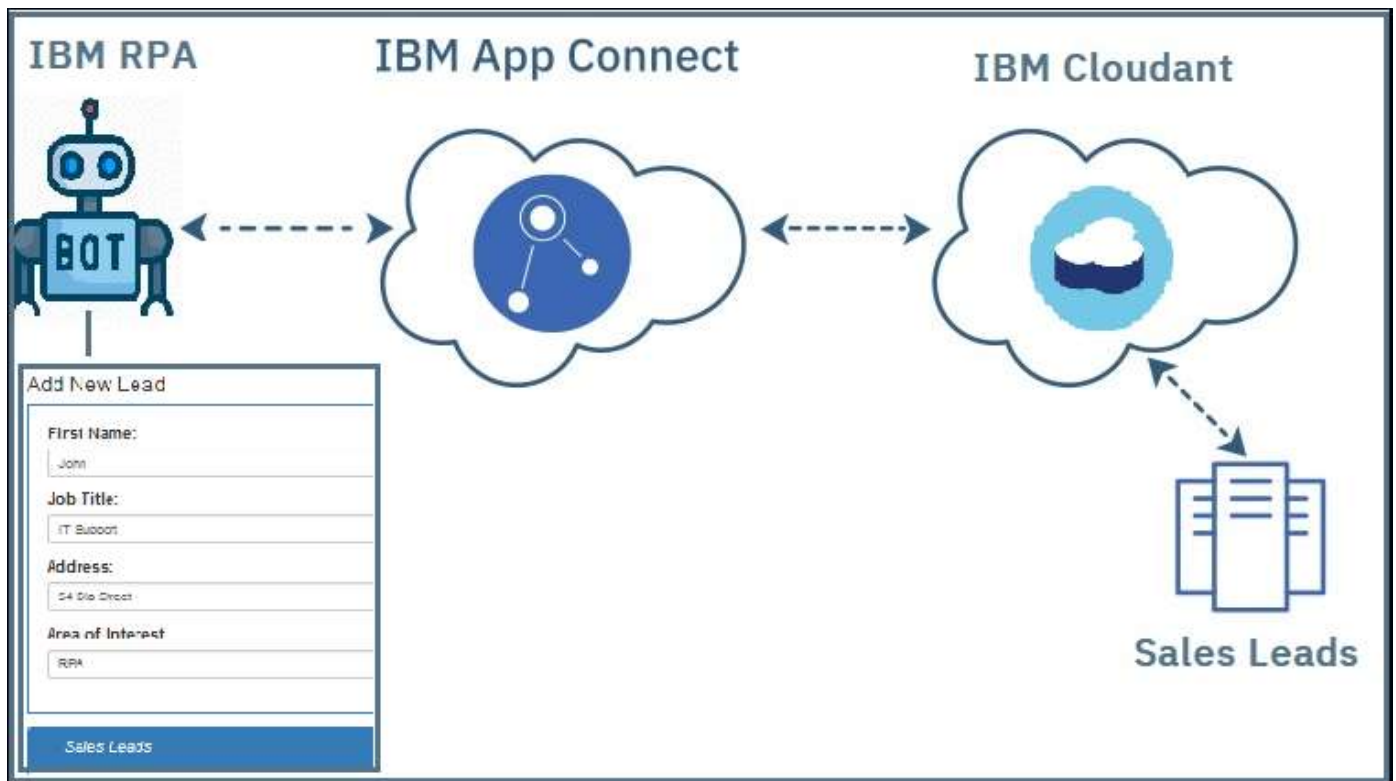


FIGURE 1. LAB ARCHITECTURE

## 2. Use case

In this exercise you will build a bot to automate sales leads for JK Automation. Sales leads are currently captured by the sales team and entered to a database. The sales leads are then manually copy/pasted into the online opportunity system. This task is error prone and the sales team repeatedly ask if this can be automated, but automation has been too difficult. Until now ...

### 3. Prerequisites

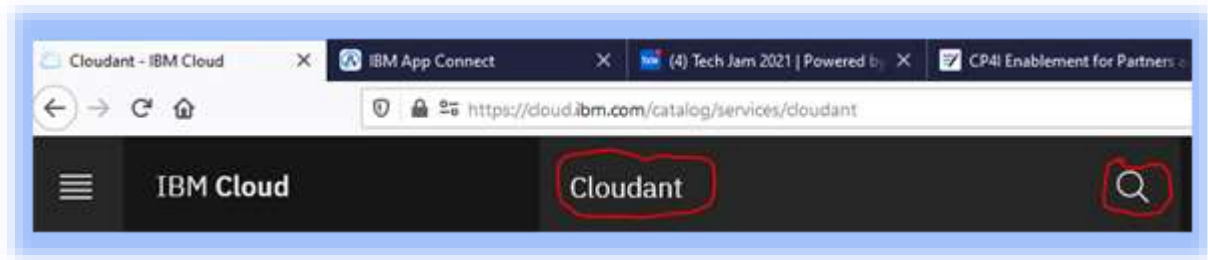
- An IBM Cloud Account. If you don't have one, visit: <https://cloud.ibm.com/registration>
- App Connect Lite (free)
- Cloudant Lite (free)

## 4. Lab Instructions

Create a Cloudant database

Log into IBM Cloud

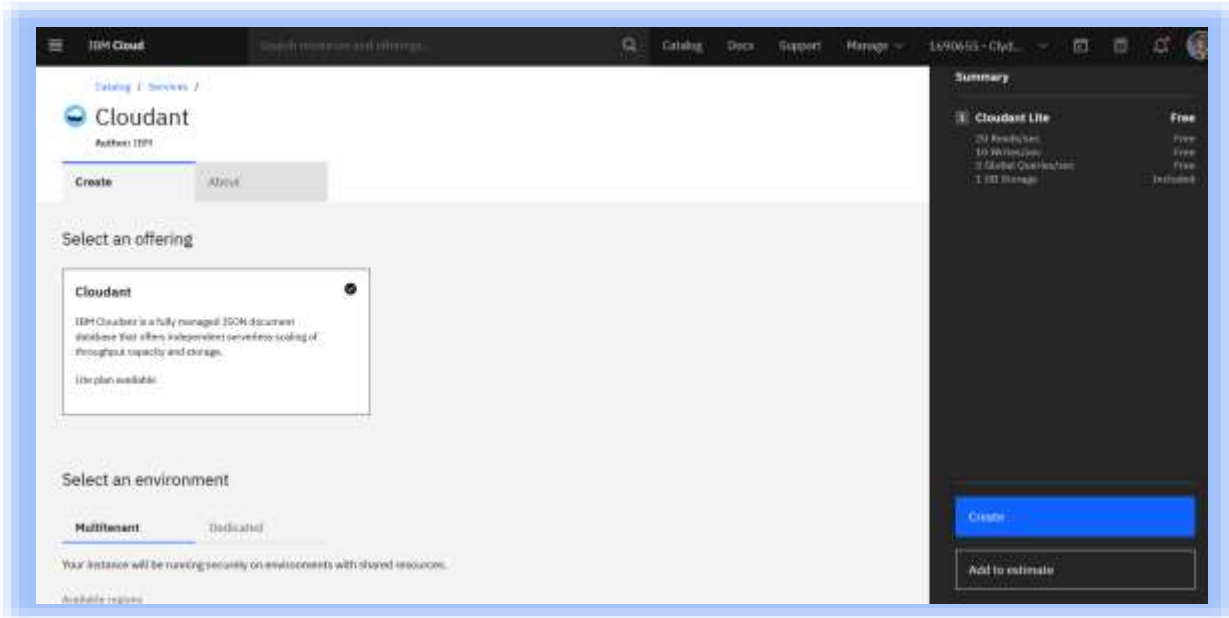
In the search bar, type Cloudant and then press the magnifying glass.



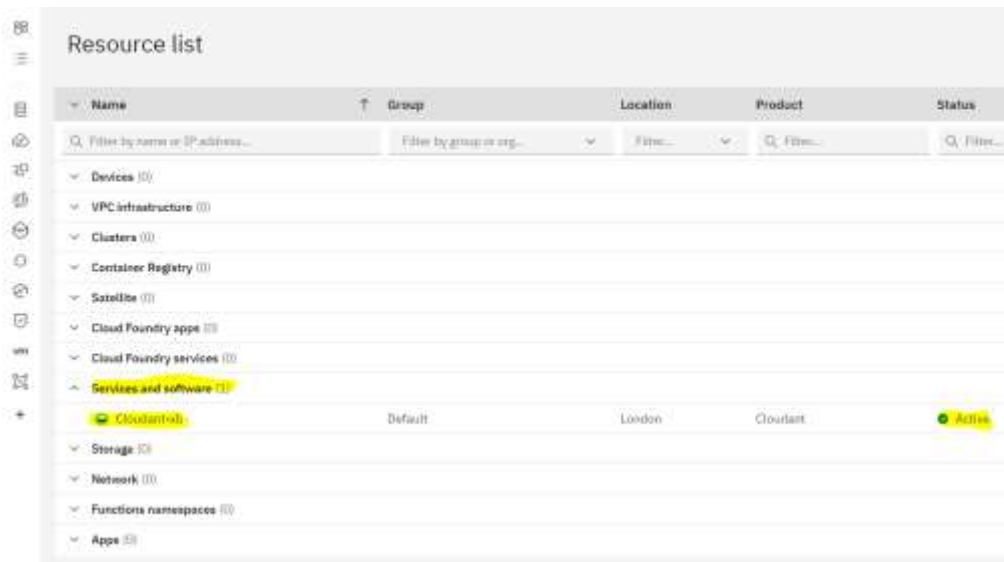
Select the Cloudant Service:



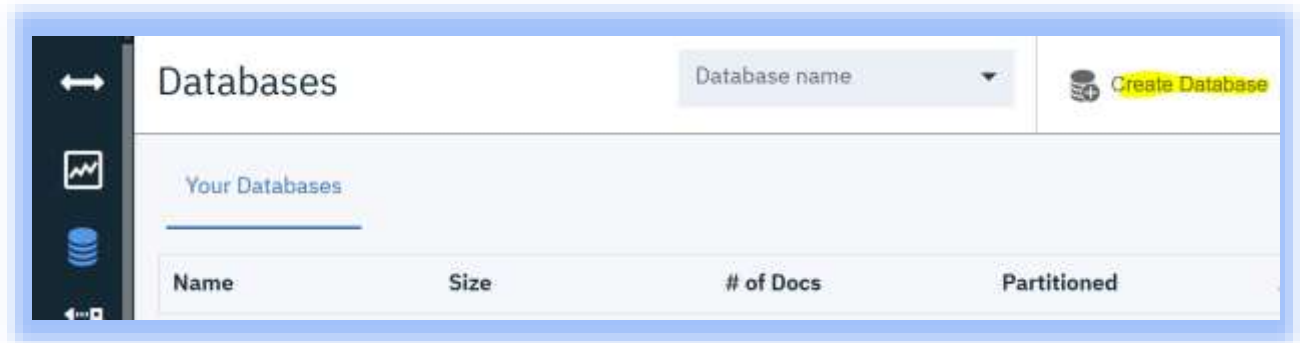
Select the **Cloudant Lite** service and then press “Create”



After a minute or two you should see the service has been provisioned under *Services and Software*. Make sure the service is in *Active* state.



Click on your provisioned Cloudbant instance and click *Launch Dashboard*. Select Databases and then *Create Database*:

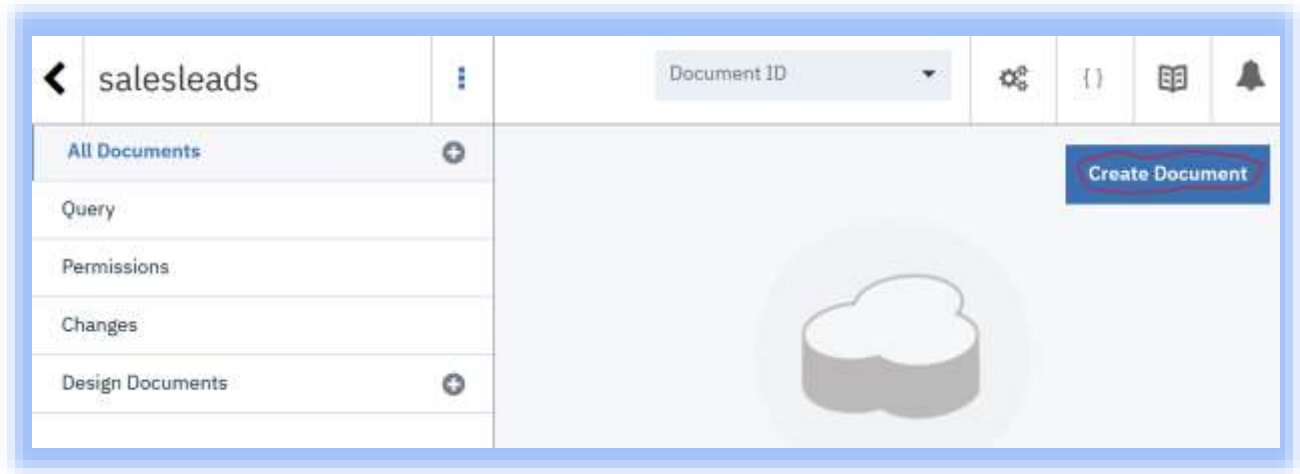


Give the database the name of *salesleads* and make it non-partitioned. Then press Create.

A screenshot of the 'Create Database' form. The title 'Create Database' is at the top. Below it is a 'Database name' label followed by a text input field containing 'salesleads'. Underneath is a 'Partitioning' section with two radio buttons: 'Partitioned' and 'Non-partitioned'. The 'Non-partitioned' option is selected. Below the radio buttons is a link that says 'What is a Partitioned Database?'. At the bottom of the form are two buttons: 'Cancel' and 'Create'.



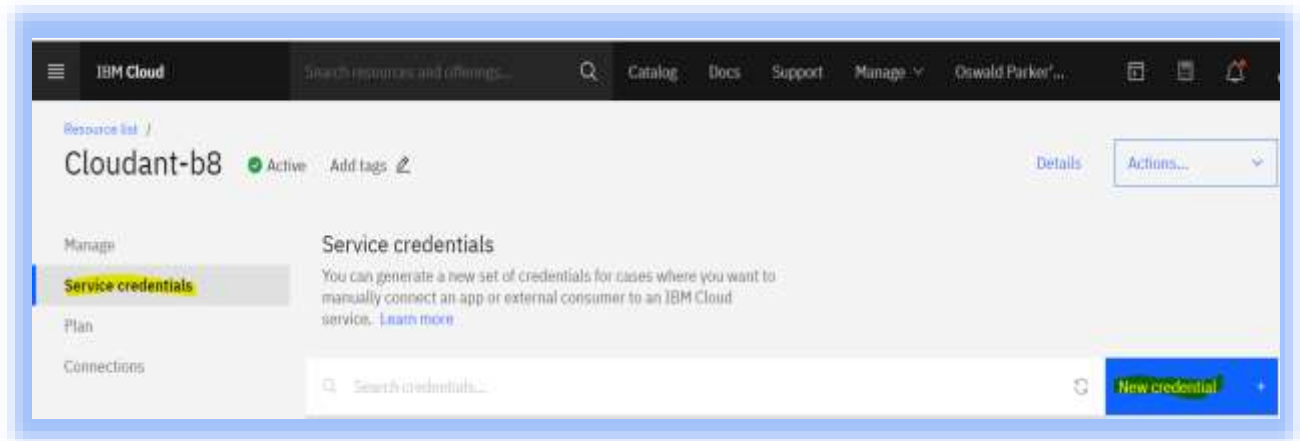
Select *Create Document*:



Paste the following JSON into the document and then press *Create Document*. You should see the message “*Document created successfully*”

```
{
  "_id": "28eb67f1e89deb5a058758433d36c446",
  "first_name": "Ned",
  "last_name": "Flanders",
  "job_title": "IT Support",
  "company": "IBM",
  "email": "ned@ibm.com",
  "phone": "87898977",
  "client_address": "101 Acasia Av",
  "client_city": "Springfield",
  "client_zipcode": "786786",
  "interest": "8",
  "followup": "Yes",
  "client_state": "North Carolina"
}
```

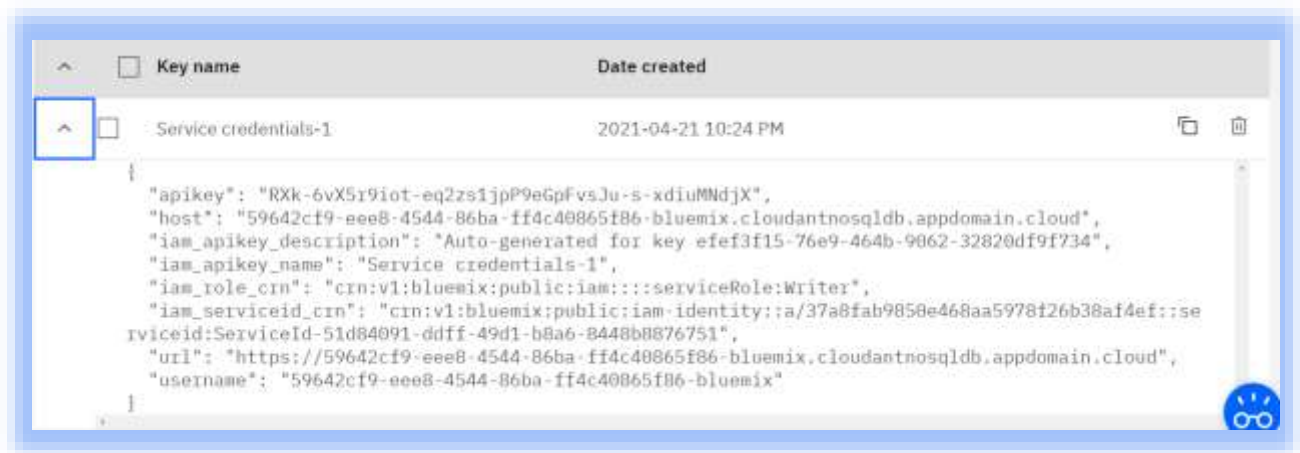
Back in IBM Cloud, click on your Cloudant instance and then click on Service Credentials. Select *New Credential*:



Give the credential name of *AppConnectCreds* and make the role *Writer*:

The screenshot shows a 'Create credential' dialog box. It has a title bar with a close button (X). The form contains two main fields: 'Name:' with a text input field containing 'AppConnectCreds', and 'Role:' with a dropdown menu showing 'Writer'. Below these fields is a button labeled 'Advanced options' with a downward arrow. At the bottom of the dialog are two buttons: 'Cancel' on the left and 'Add' on the right.

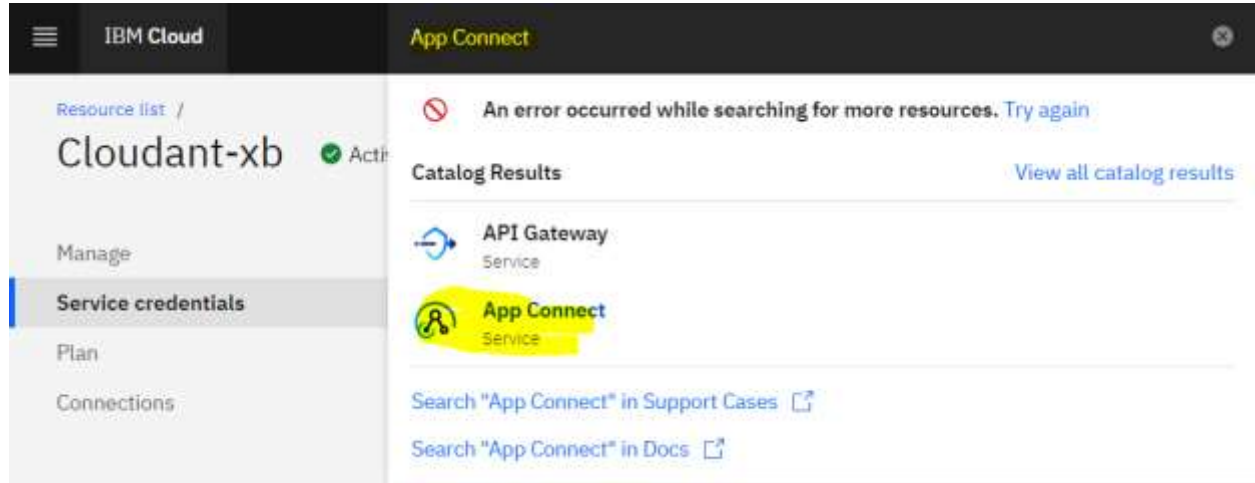
Press *Add*. Now expand the newly created service Credentials to view the details.



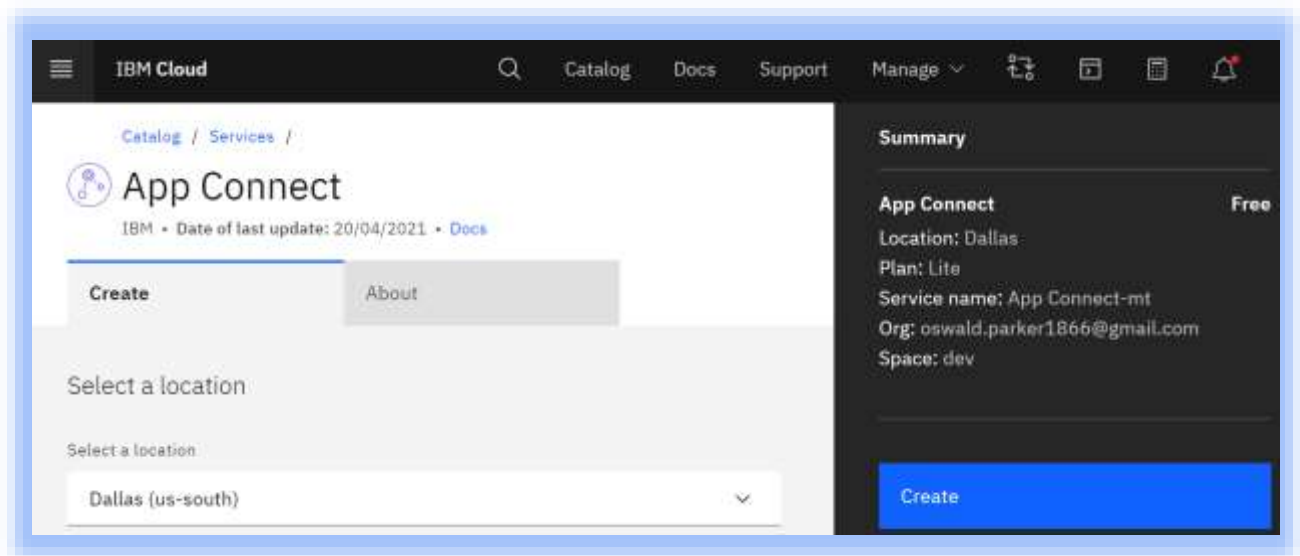
Copy these details into notepad so that you can use them in the next section.

## Provision an App Connect Instance

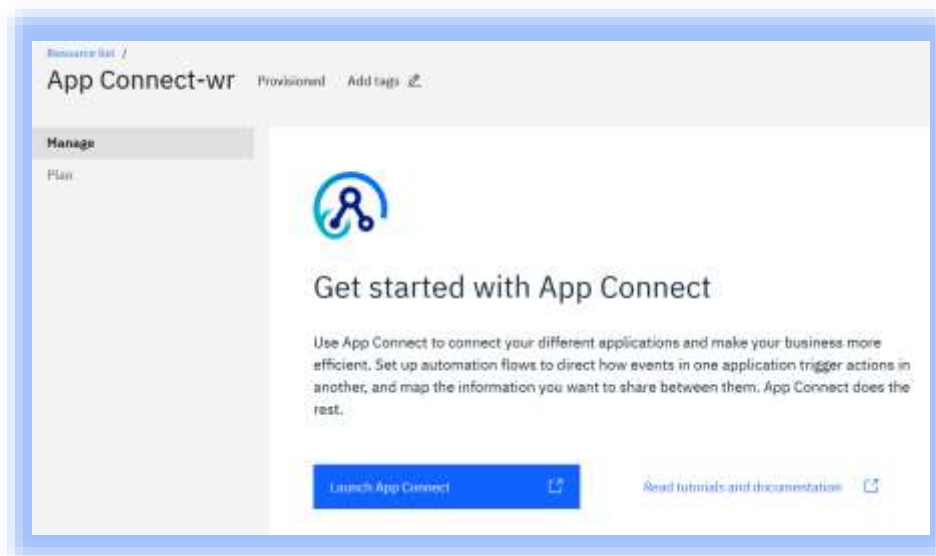
Still in IBM Cloud, in the search bar, type *App Connect*:




Select the *App Connect* service. Create an App Connect Lite (free plan) by pressing *Create*.

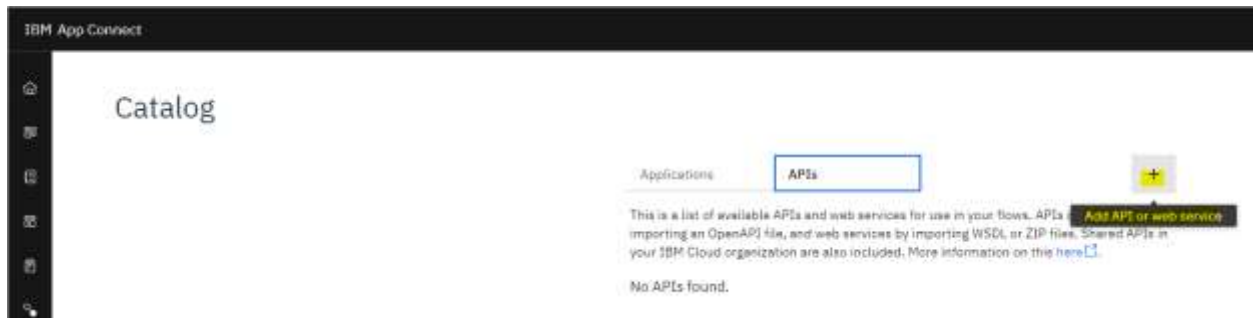


After a few minutes your App Connect instance will be provisioned. Select Launch App Connect.

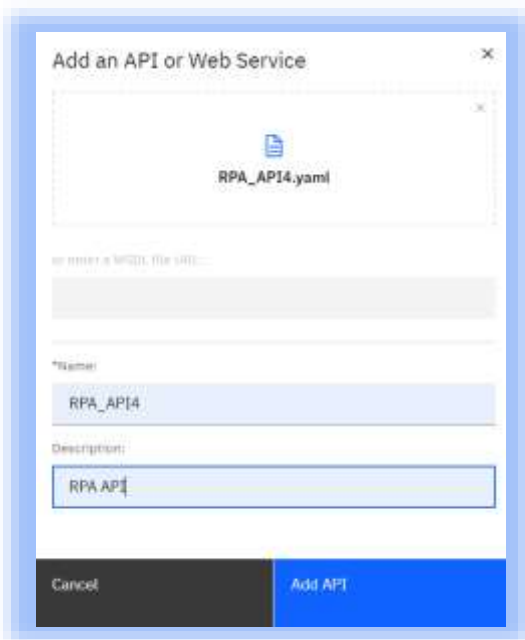


## Import the RPA Bot API

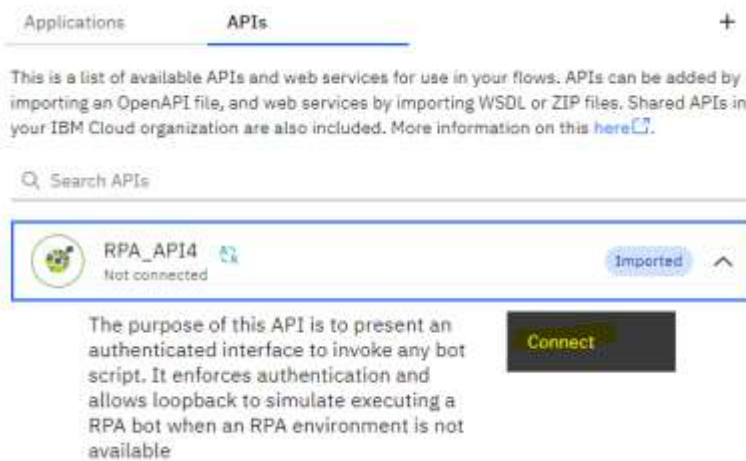
On the left-hand side, select *Catalog*  and then select the *APIs* tab and then press the + button



Drag [https://rpapi.eu-gb.mybluemix.net/RPA\\_API.yaml](https://rpapi.eu-gb.mybluemix.net/RPA_API.yaml) to the drop area and press *Next*. Enter the name *RPA\_API4* and description *RPA API*. Note the name *RPA\_API4* must be exact. Press *Add API*. The API should be created without errors.




Select the drop-down arrow on the newly created *RPA API* and press *Connect*

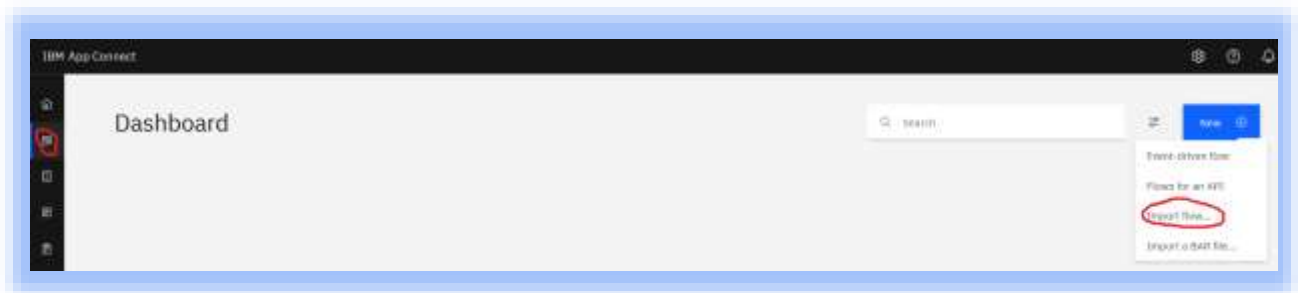


Leave all fields as default and press *Connect* again.

An API Account should now have been created, which we will use in the next step.

## Create an App Connect Flow

On the left-hand side, select *Dashboard*  and then on the right-hand side, press *New->Import Flow...*



Drag <https://rpapi.eu-gb.mybluemix.net/SalesLeadsApi.yaml> to the drop area and press *Import*. The API Flow should be imported. You should see the flow with several errors that we will rectify in the next step.



## Fix Errors in the Flow

Click on the Scheduler inside the flow. Create a schedule as shown below

A screenshot of the 'Schedule flow' dialog box. The dialog has a title bar 'Schedule flow' and a close button. Below the title bar, there is a section '\*Select schedule type' with a dropdown menu showing 'Repeating interval'. Below this, there is a section 'Run every' with a dropdown menu showing '23' and a unit dropdown showing 'Hour'. Below this, there is a section 'on' with checkboxes for 'Mon', 'Tue', 'Wed', 'Thu', 'Fri', 'Sat', and 'Sun'. The 'Sun' checkbox is checked. Below this, there is a section 'Timezone' with a dropdown menu showing 'UTC'. At the bottom, there is a checkbox labeled 'Also run the flow when it's first switched on' which is checked.

The error should disappear.

Click on IBM Cloudant. Click on connect. Enter the credentials of the Cloudant instance you created earlier. The Cloudant password is not required:

Retrieve documents

Connect to IBM Cloudant

Cloudant user name: (optional)

59642cf9-eee8-4544-86ba-ff4c40865f86-bluemix

If the service instance uses legacy credentials for authentication, provide your user name.

Cloudant password: (optional)

|

If the service instance uses legacy credentials for authentication, provide the associated password.

Cloudant host: (optional)

59642cf9-eee8-4544-86ba-ff4c40865f86-bluemix.cloudantnosqldb.appdomain.cloud

If the service instance supports Identity and Access Management (IAM) authentication, provide the host name of the instance.

Cloudant API key: (optional)

.....

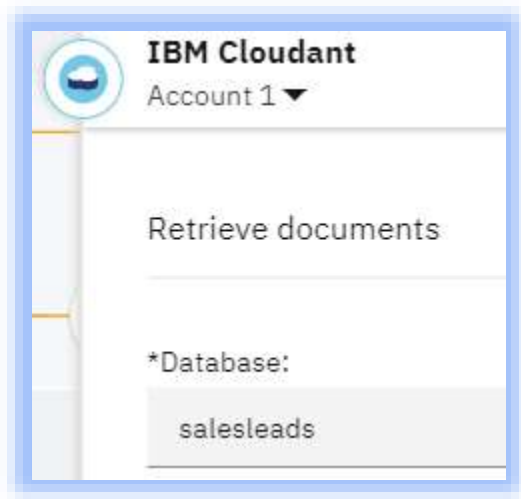
If the service instance supports IAM authentication, provide your API key.

Cancel

Connect

If you have more than one Cloudant database, select *salesleads*





Click *IBM Cloudant*, select the drop down and select *salesleads* under *Database*. The database should have no errors.

Back in the flow, click on RPA\_API4. You should see the following:

runScript 📄 ✎ ⋮

---

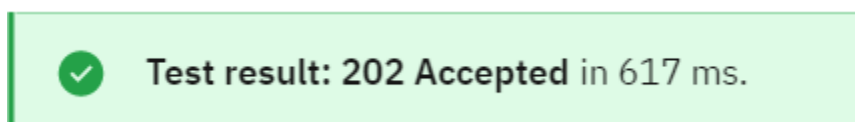
Populate the target fields in RPA\_API4 ⓘ

script	<div>abc ▼</div>	sales_lead_automation_API
host	<div>abc ▼</div>	LOOPBACK
unlockMachine	<div>abc ▼</div>	False
RequestBody	<div>{ }</div>	<div>(json) Parsed JSON</div>

Press the *play* button and then press *continue*:



After a moment you should see:



This verifies that the API is working.

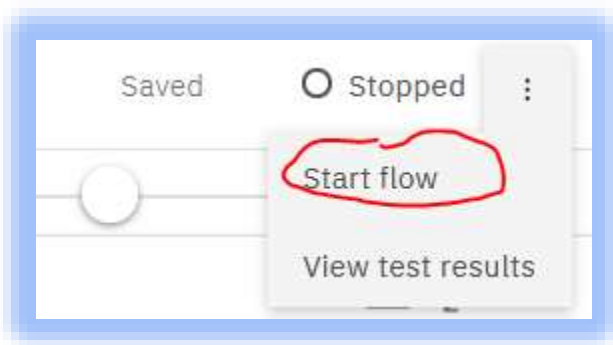
Note that for this lab we are using an emulated RPA *host* called LOOPBACK. To see the API invoked using a real RPA server host, view the following video:

[https://youtu.be/1K\\_Zja-okQg](https://youtu.be/1K_Zja-okQg)

All the errors in the flow have now been fixed, so we can now run the entire flow.

### Run the App Connect Flow


Choose *Start flow* under the vertical ellipsis at the top right of the flow



The flow will run. It will perform the following:

1. Read a document from Cloudant
2. Invoke the RPA bot using the document as bot parameters
3. The bot fills in a web form using the passed in parameters.
4. Once executed, the bots pass back the results where they are written to Cloudant
5. If there are several Cloudant documents, the process repeats from step 1

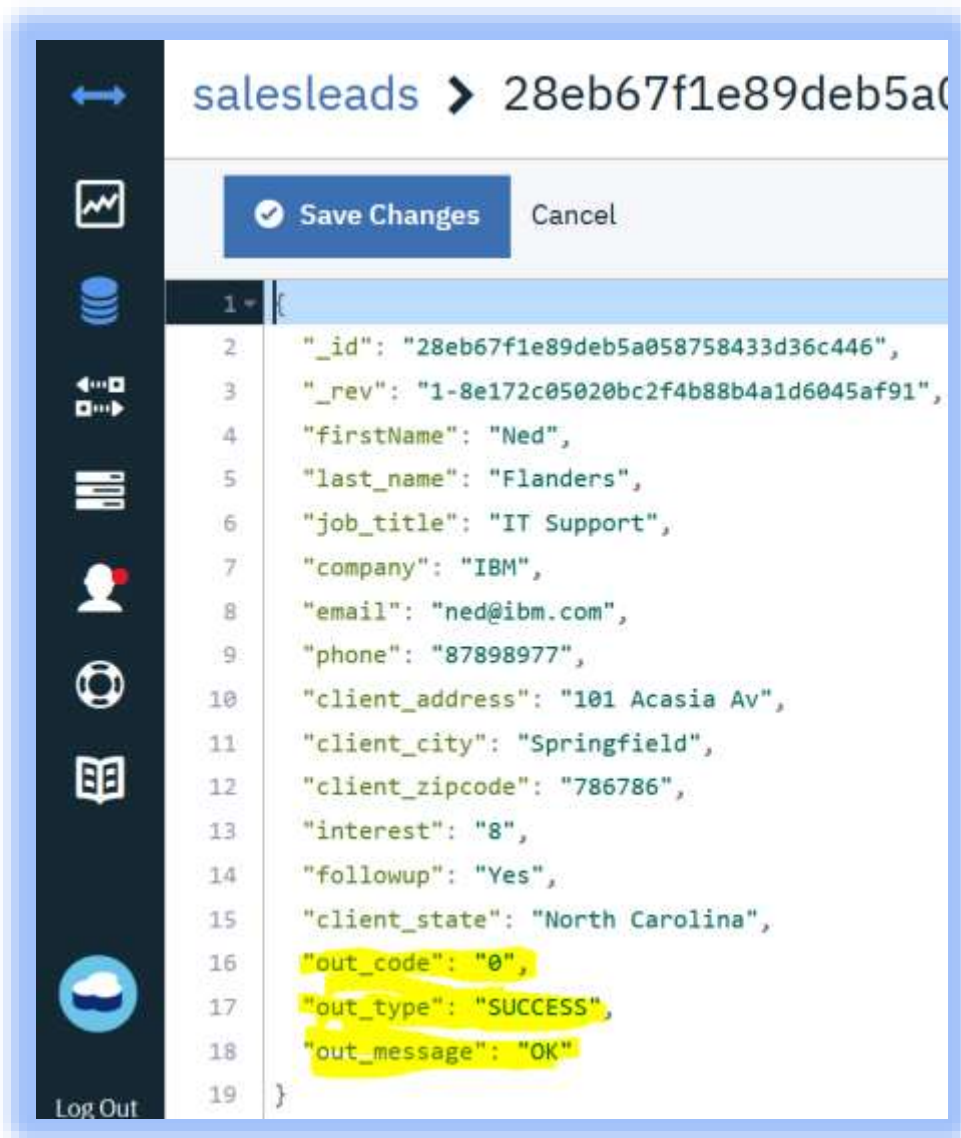
### View App Connect Logs


To view the result, select the Logs  icon. After a few minutes you should see `Flow SalesLeadsRPA completed successfully` in the log

Event time (UTC)	Message
<div data-bbox="228 275 245 296">▼</div> 2021-09-15 16:32:03.852	Flow SalesLeadsRPA completed successfully.
<div data-bbox="228 350 245 371">▼</div> 2021-09-15 16:32:03.800	Processing complete
<div data-bbox="228 426 245 447">▼</div> 2021-09-15 16:16:41.382	Successfully created API "RPA_API4".
<div data-bbox="228 501 245 522">▼</div> 2021-09-15 16:05:27.350	An instance of App Connect with ID fog7zo51h has been provisioned.

## View Processing Result in Cloudant

To view the processing results in Cloudant, go back to your Cloudant database and open the document you created earlier. The processing results from the bot have been added to the Cloudant document. This indicates that the bot added data to the JK Automation web site and returned successfully.



Go back into App Connect *Dashboard* , Within the SalesLeadsRPA vertical ellipses menu, select *Stop Flow*.

Congratulations. You have completed the RPA with App Connect lab!

**THIS COMPLETES THIS HANDS-ON STANDARD LAB**