Base Oauth2 Authorization User Manual

# INTRODUCTION

This two module will implement **Oauth2 Authorization server** in Odoo and its services and Oauth2 client service for odoo.

## FEATURES

* Implemented in Odoov13.
* Have the Grant Type Authorization Code with response type as “code” .
* Have the Grant Type Implicit with response type as “token”.
* Have the Grant Type Resource Owner Credentials.
* Have the Grant Type Refresh Token.
* Have the login with “access\_token”
* Can act an Odoo instance as client to Our Authorisatiion Server.

# INSTALLATION

Now simply follow these steps:

pip3 install oauthlib

For Oauth2 Authorization Server

1. Go to settings menu and click on Activate the Developer mode.
2. Now go to apps menu and Click on ‘Update Modules List’.
3. Remove the app’s filter and search for the the word ‘base\_oauth2\_authorization\_services’.
4. Now you will see the module then simply install it.

For Oauth2 Client for an Odoo client (Optional )

1. Go to settings menu and click on Activate the Developer mode.
2. Now go to apps menu and Click on ‘Update Modules List’.
3. Remove the app’s filter and search for the the word ‘oauth2\_auth\_redirect’’.
4. Now you will see the module then simply install it.

After installing you will be able to handle different functionality as mentioned in module’s work flows.

# CONFIGURATIONS FOR Oauth2 Authorization Server

**Go to Rest API Menus under Settings**

All configurations are assuming that **localhost:9879 as Oauth2 Authorization Server.**

All configurations are assuming that **127.0.0.1:9889 as Oauth2 Odoo Client.**

The endpoints should need to be same which means ipaddress, hostname, domain or port can change according to your implementations.

(for eg:

http://127.0.0.1:9889/auth\_oauth/oea

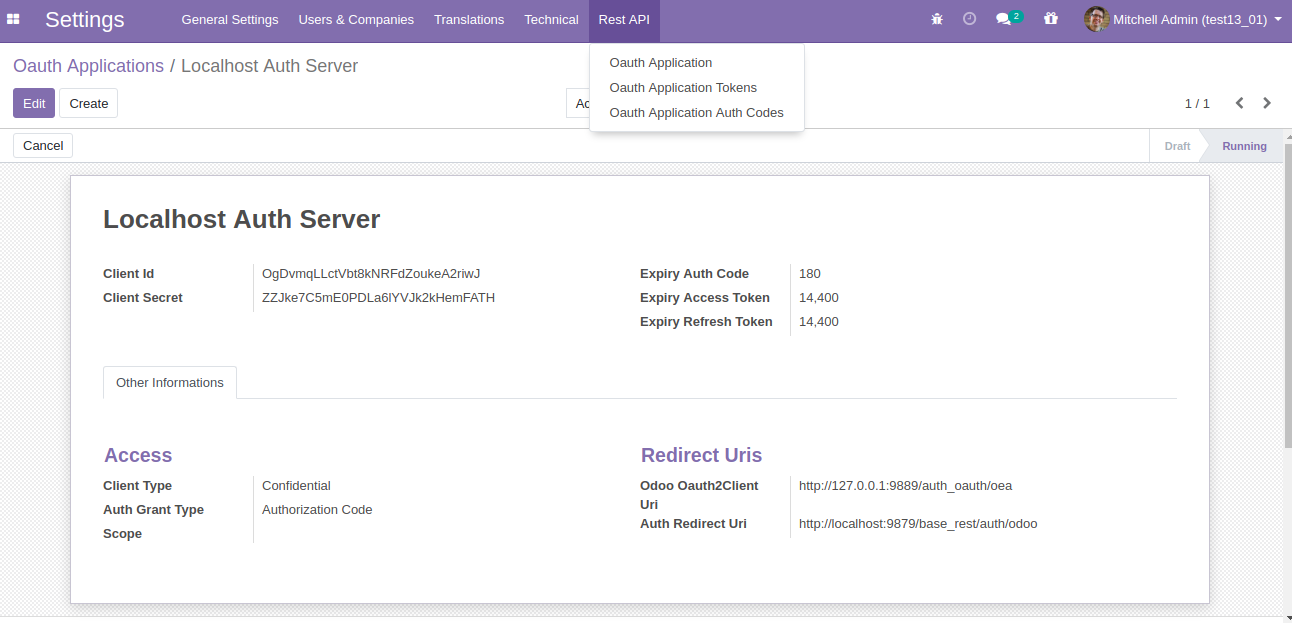
can change to

http://192.168.1.41:8879/auth\_oauth/oea

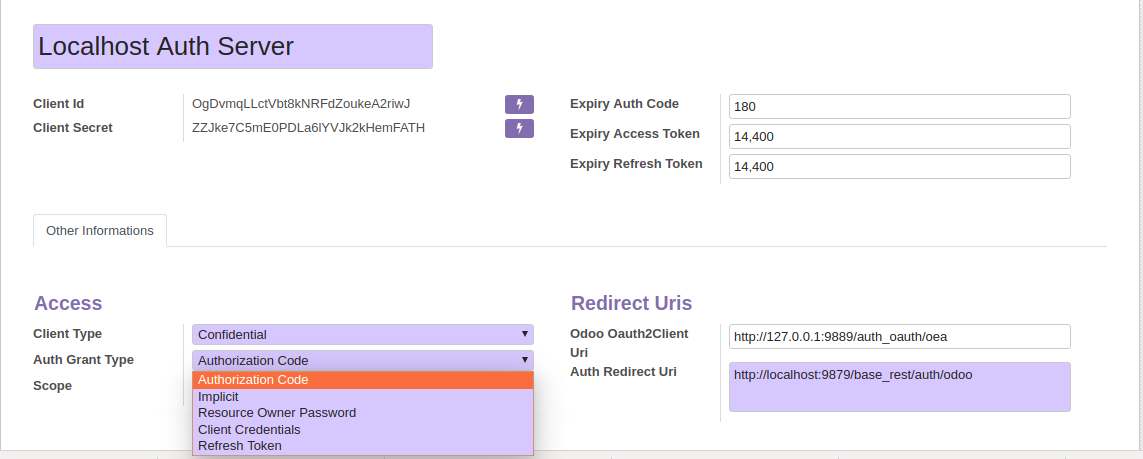
or http://localhost:9879/base\_rest/auth/odoo

can change to

http://10.0.0.11:8875/base\_rest/auth/odoo)

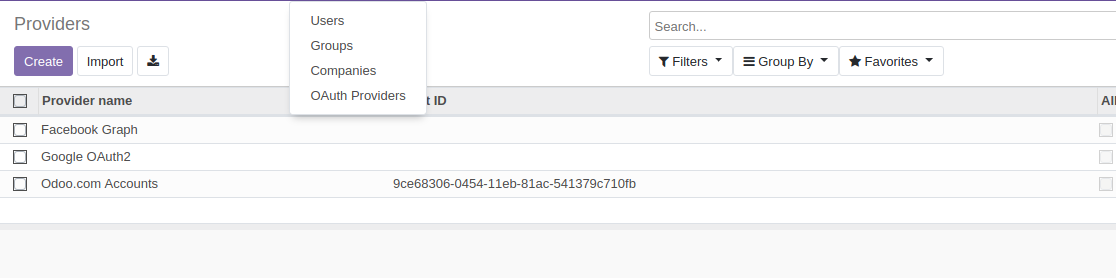
****

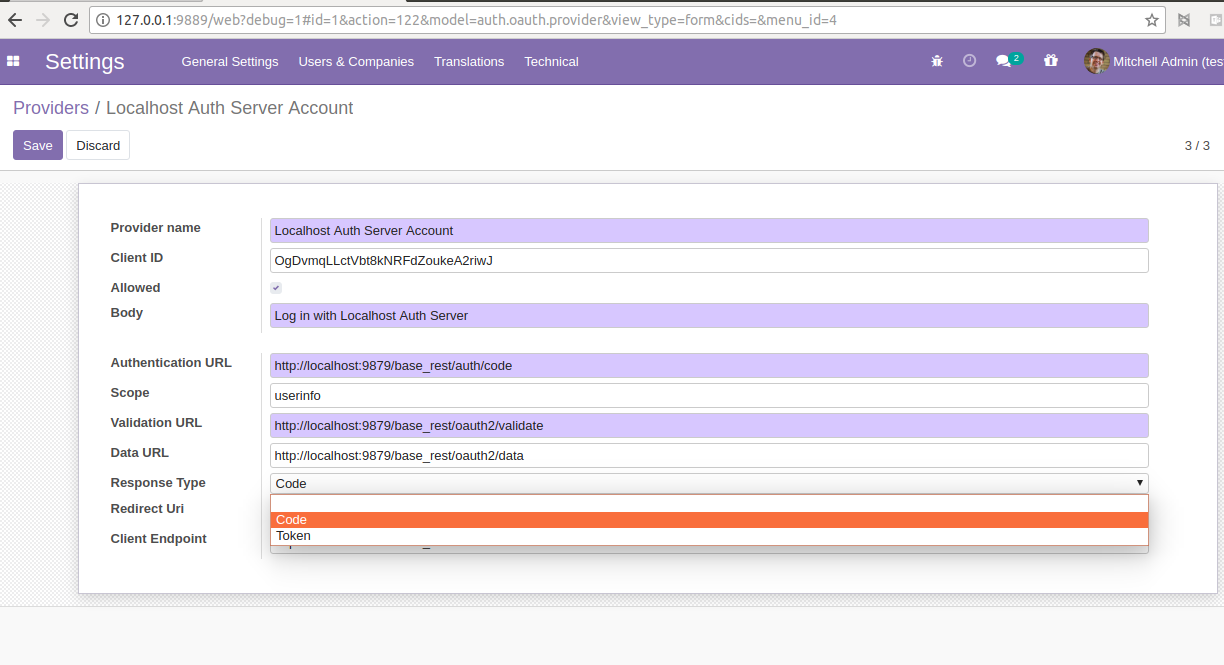
Configure the Grant type, redirect URL, Odoo , etc



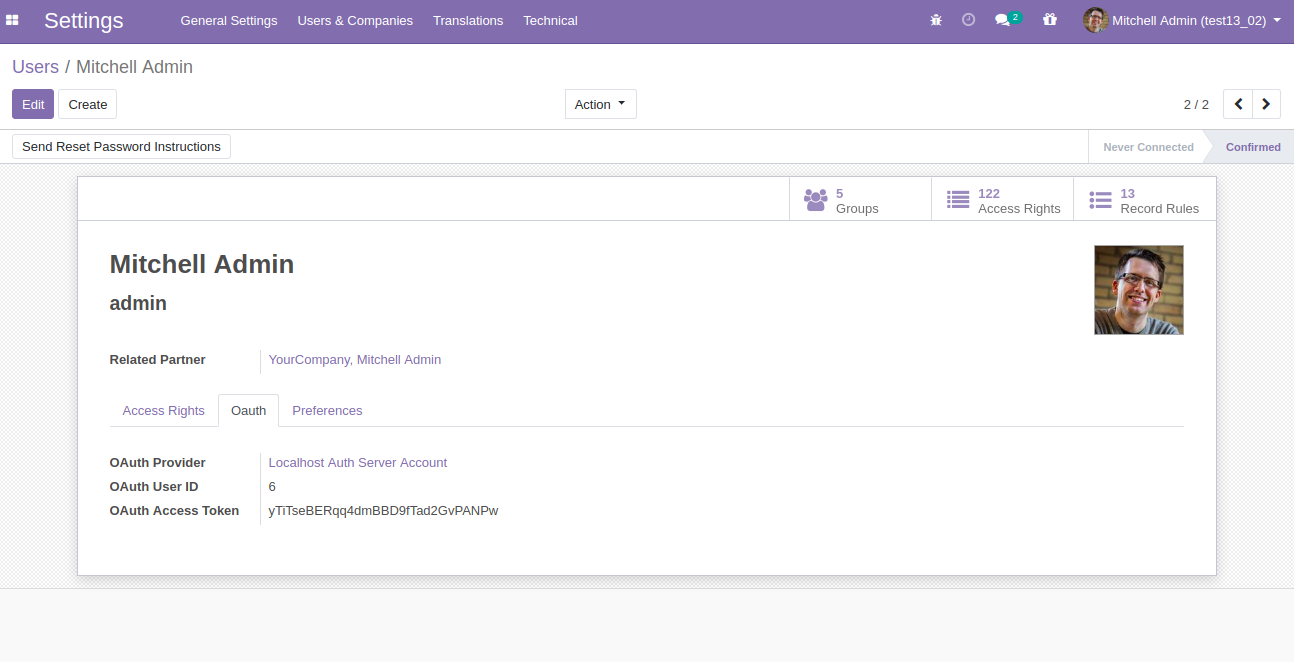
For using an Odoo client (enterly different server and db)

Do the following configuration

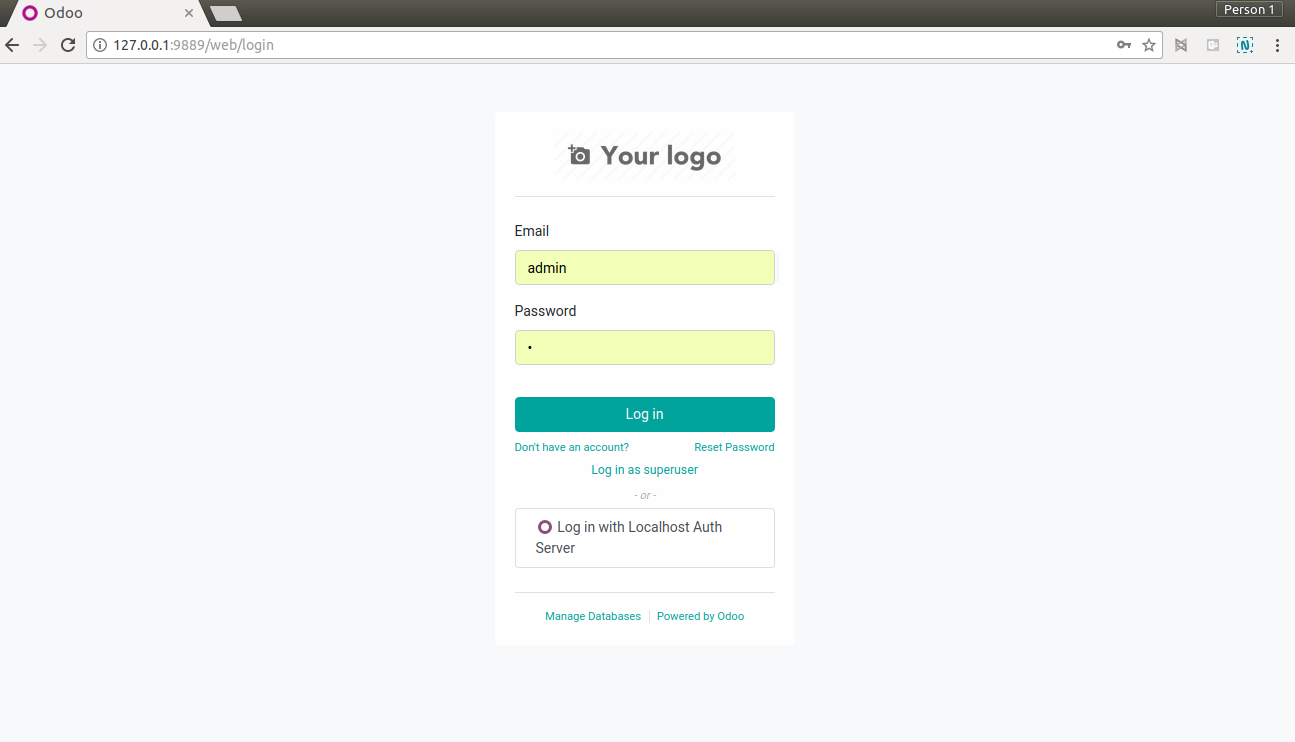
For time being choose Odoo.com Accounts and change the details.



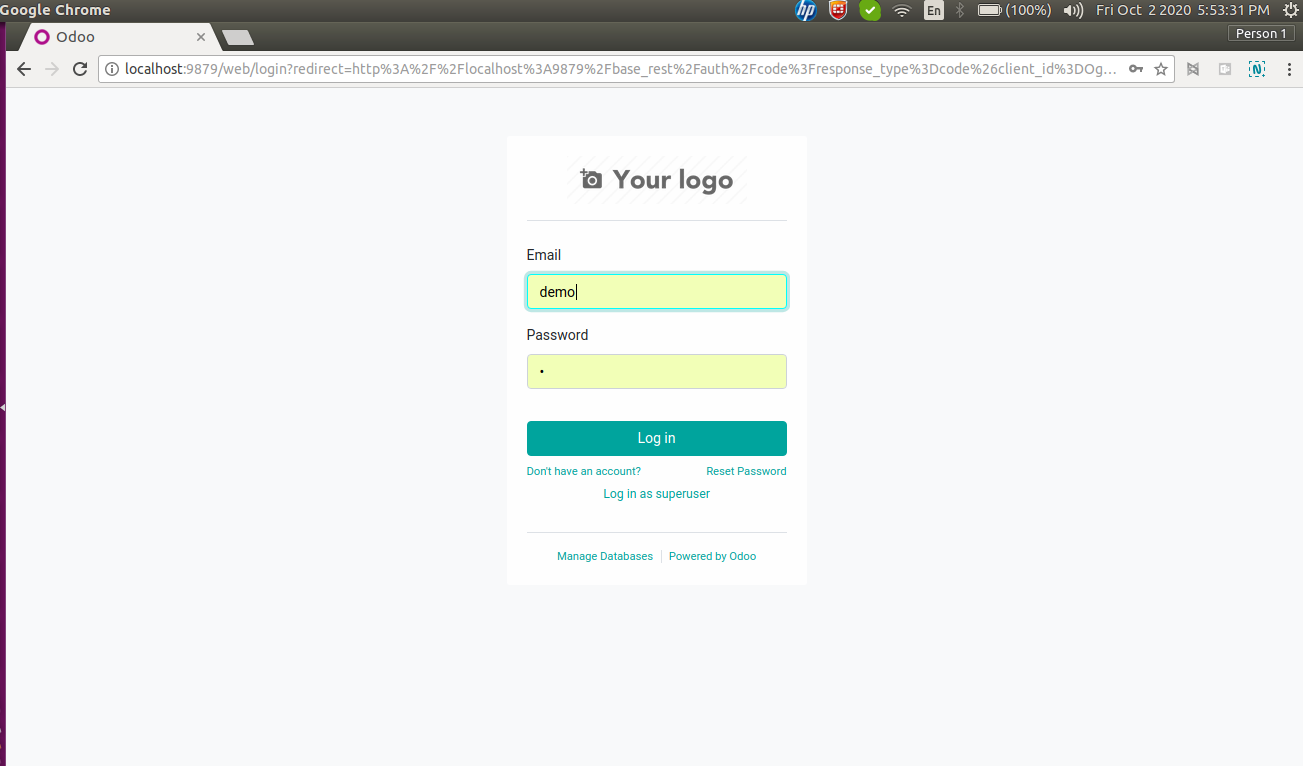
**For testing as Grant Type “Authorisation Code” or “ Implicit”**

First need to map the users with Oauth2 odoo client with the Authorisation server user reference

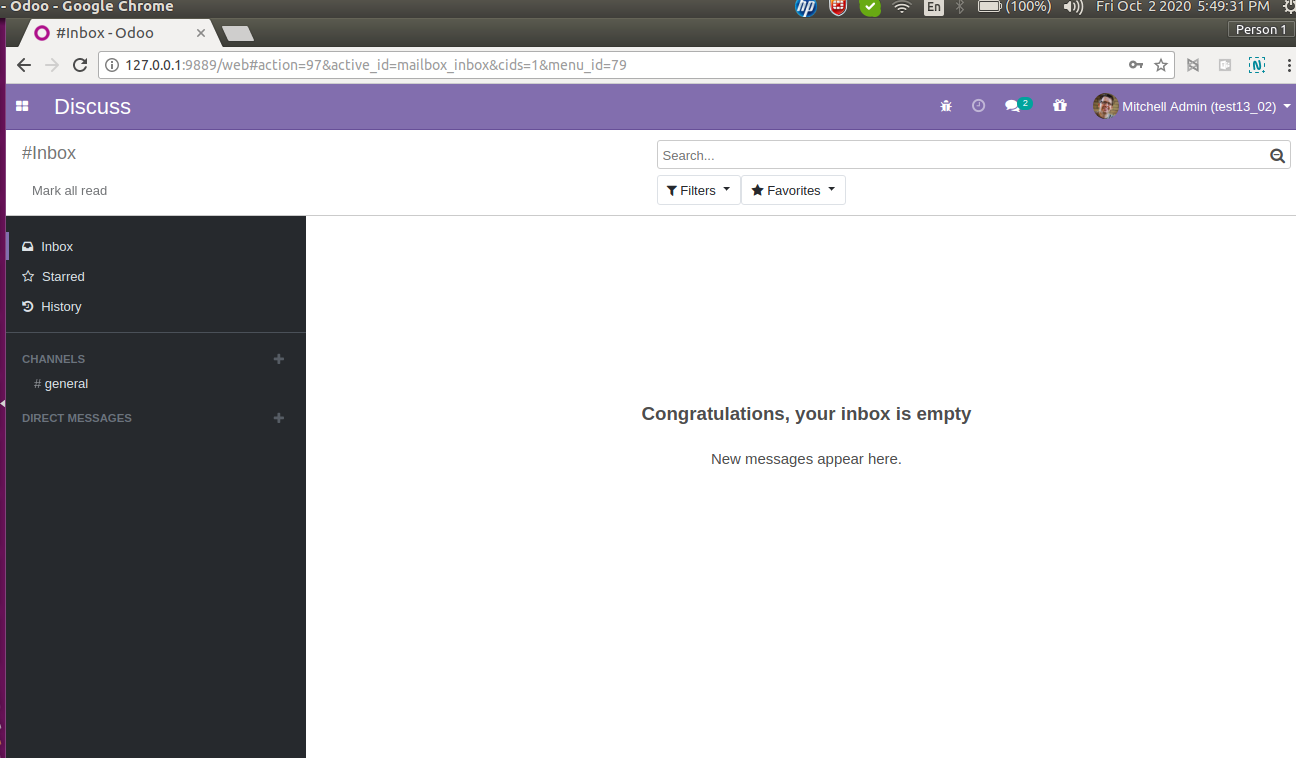
Then go to Oauth2 Odoo Client server and DB instance Login page



Choose Login in with Localhost Authorization Server (here means another Odoo instance and different DB running on localhost)



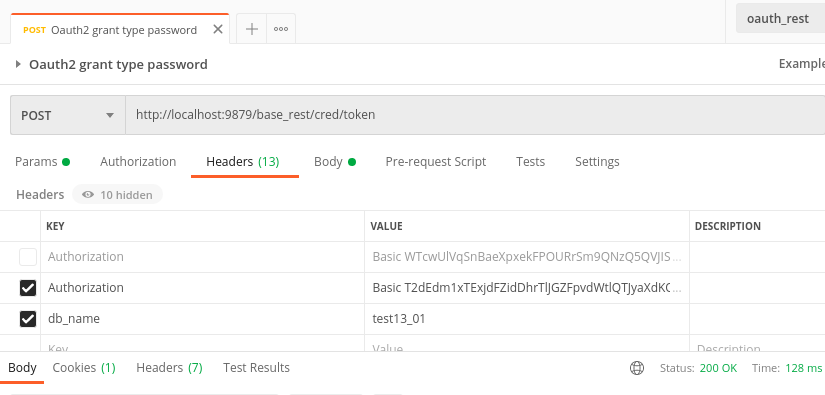
After clicking that button the browser will redirect to Oauth2 Authorization server. The user need to login to **“Oauth2 Authorisation server”** (here it is localhost) with the mapped user with Oauth2client (here user login called “demo” in Oauth2 server).



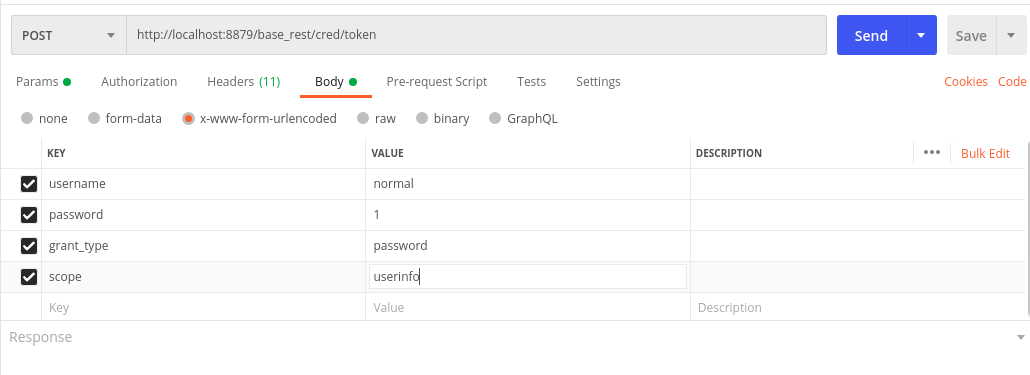
After successful login the OAuth2 Authorisation Server redirected back to Oauth2 Odoo Client and their it successfully login.

Some screen shots of other Grant Types.

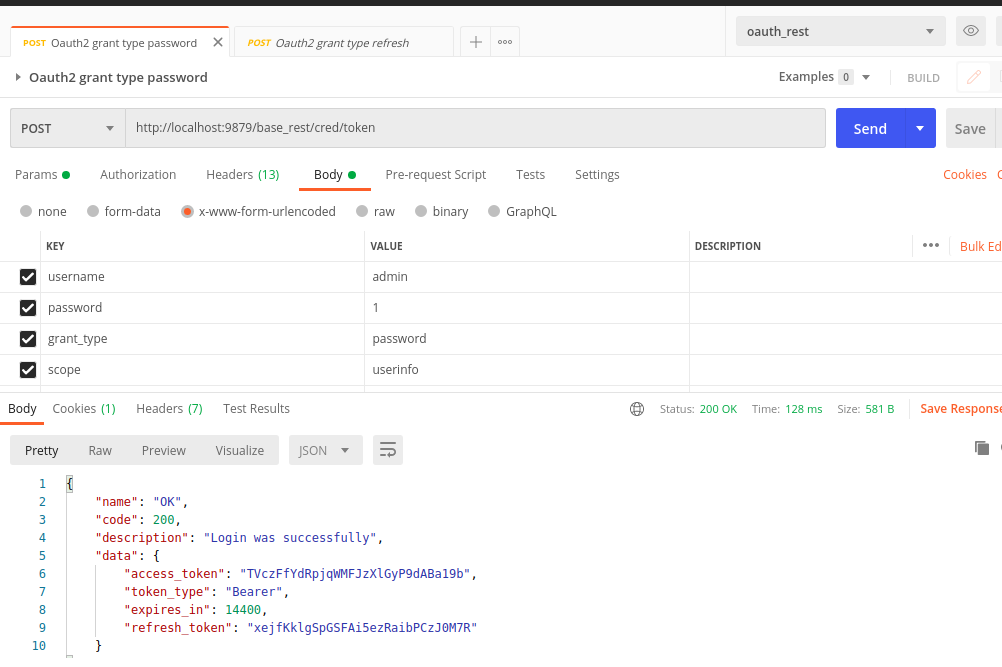
**Grant Type Resource Owner Credentials.**



Params needed in header. (Authorization Header should be the following format : Basic <BASE64({CLIENT ID}:{CLIENT SECRET})>

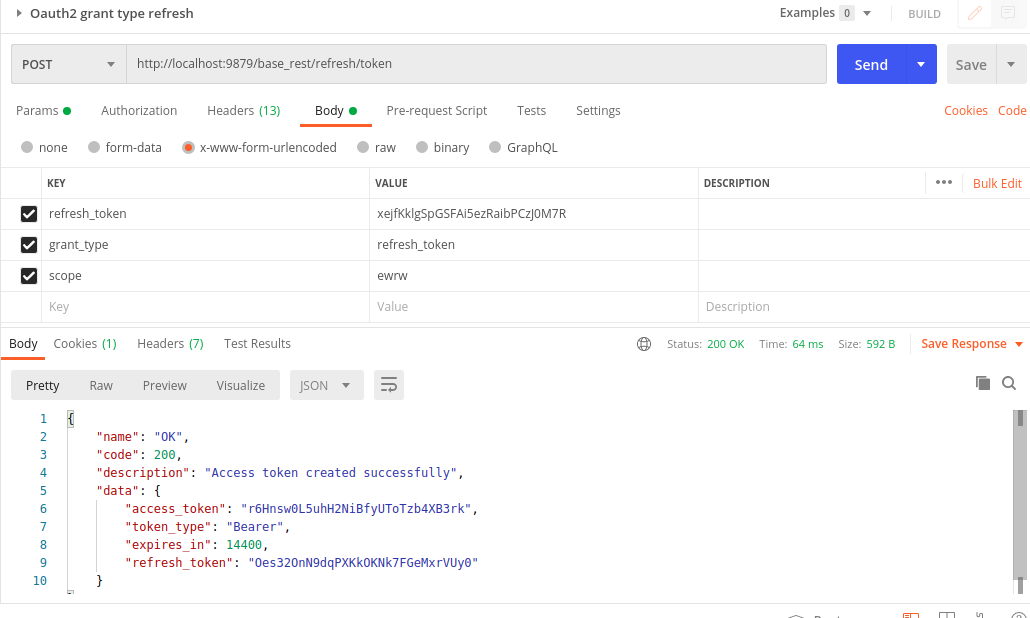


Params needed in form data.



Result Fetched.

**Grant Type Refresh Token**



Params in form data and result fetched. Header should be same as previous (resource owner grant type)

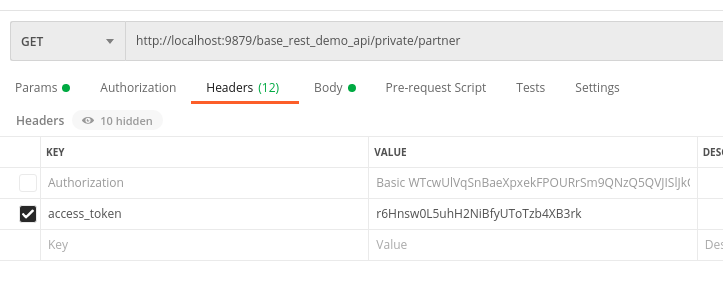
**Login with Access Token**

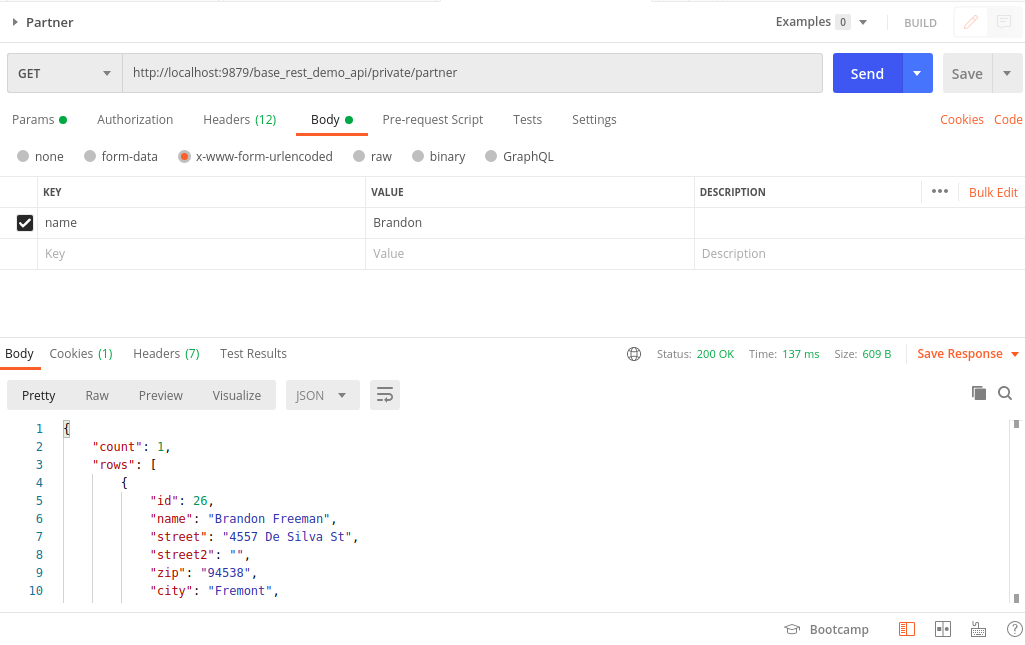
Sample taken from the module “base\_rest\_demo” private service.



The variable “default\_auth” needs to set to “access\_token”.

We noticed that the module “base\_rest” have some bugs. We rectified it and after that it works.

 params needed in header



params needed in form data and result fetched.

**Known Issues:**

* Need to write unit tests
* Didnt tested properly. Tested only during the development time.
* Sorry for the document as we prepared in a hurry

**Future Plans**

* Need to work on client type and scope
* A generic solution for fetching the data from models