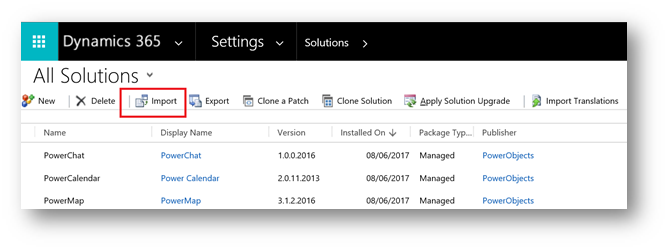
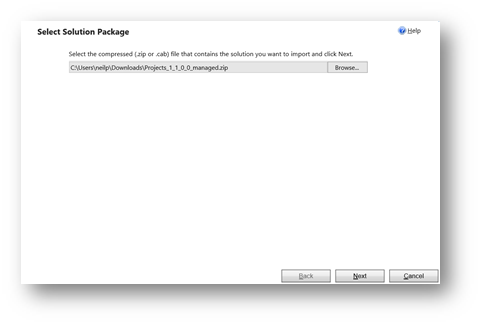
# Prerequisites for CICD Automation tool:

## Deploy Solution in Instance

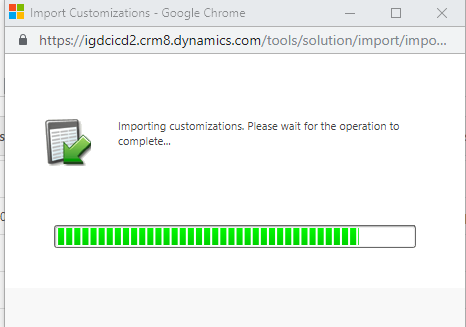
1. Download the attached zip file.



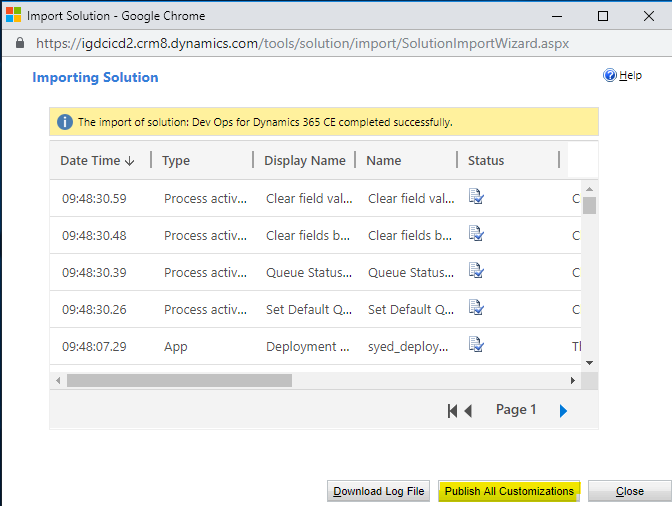
1. Login to Dynamics 365. Navigate to Settings-> Solutions.
2. Select the import option.
3. Now browse for a solutions file and click on Next



1. Click Import button. Then you will see below screen.



1. Once Import completed. Please click on Publish All Customization button.

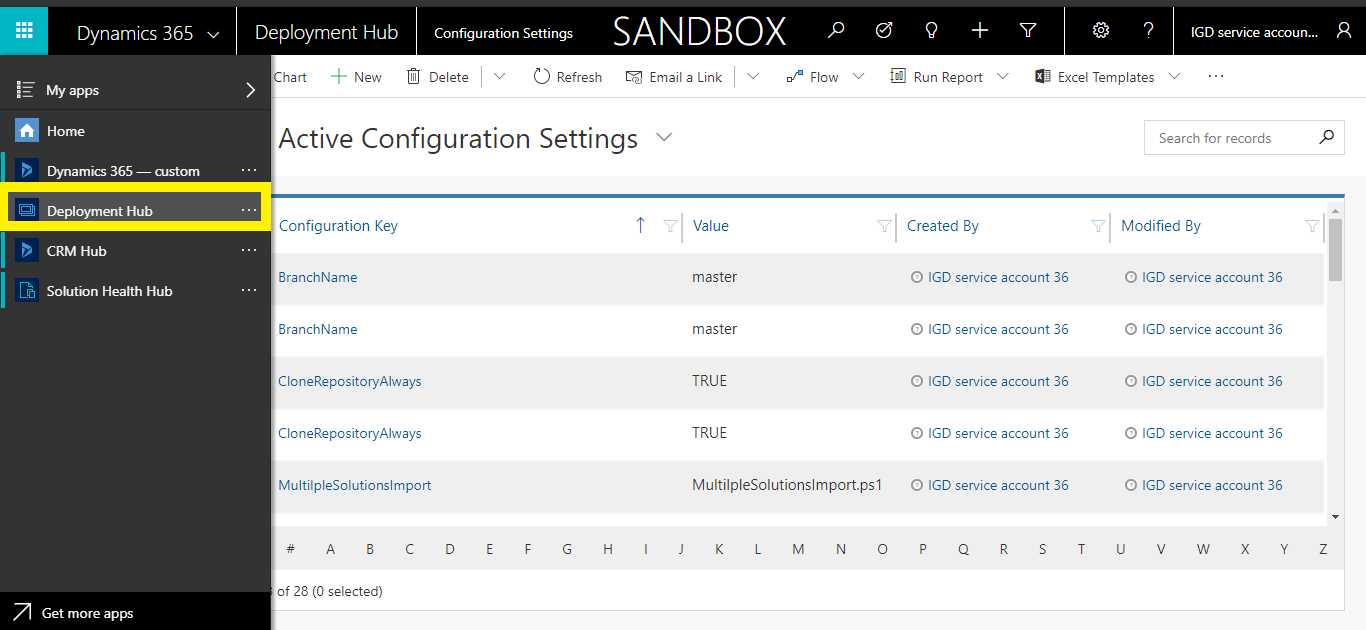


## Configuration Settings.

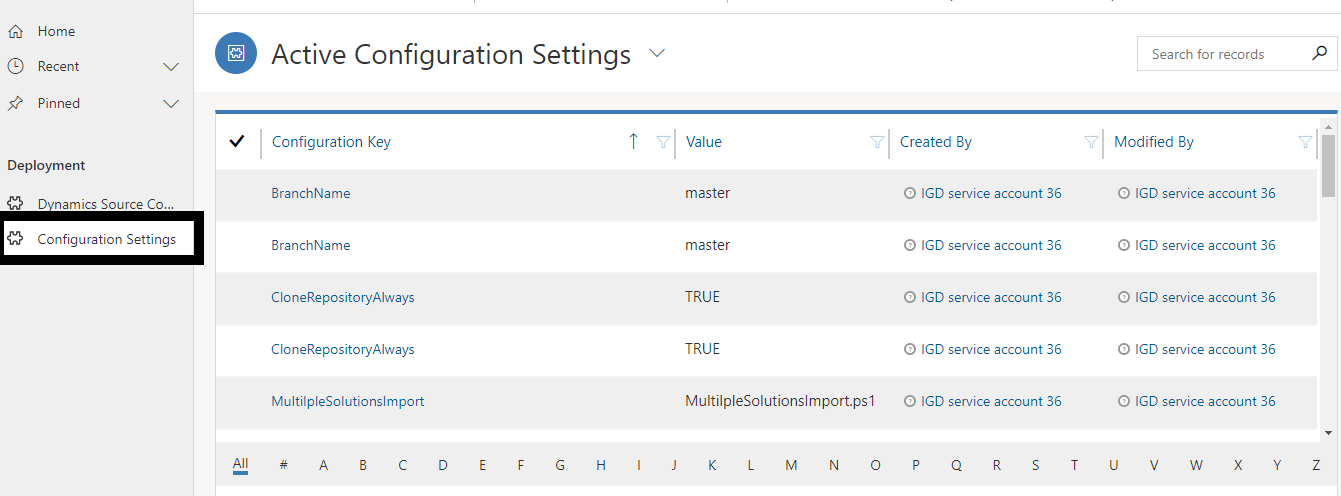
1. Download the attached configuration settings file.



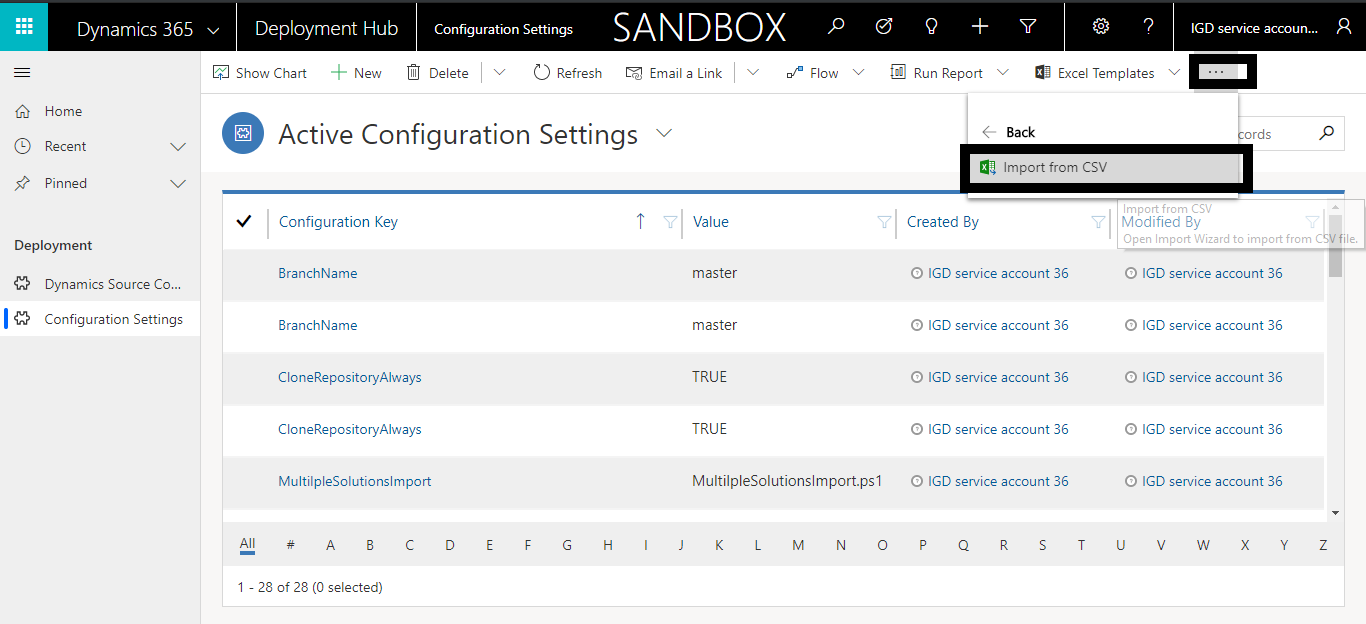
1. Fill the respective values for configuration key and save the file.
2. Once it is saved, open CRM instance in browser and navigate to My apps-> Deployment Hub.



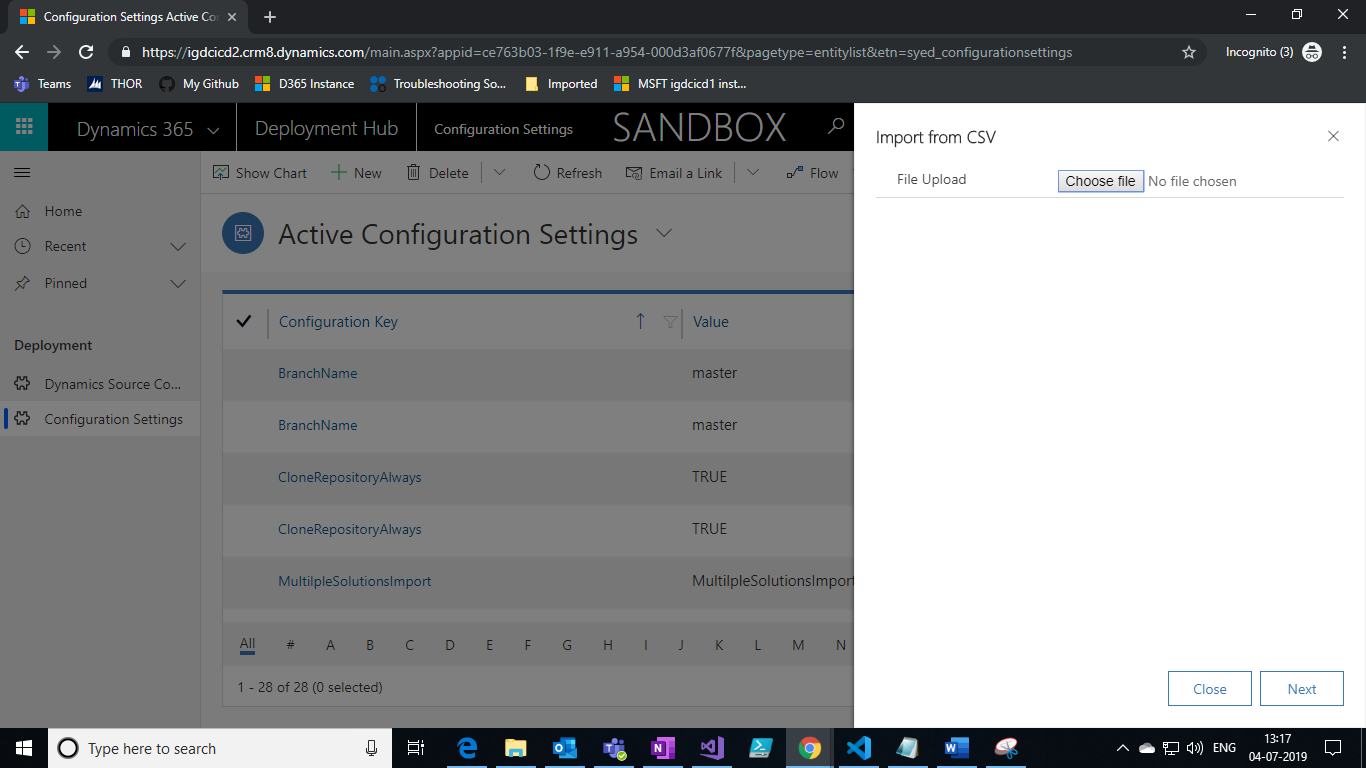
1. Click Configuration Settings.



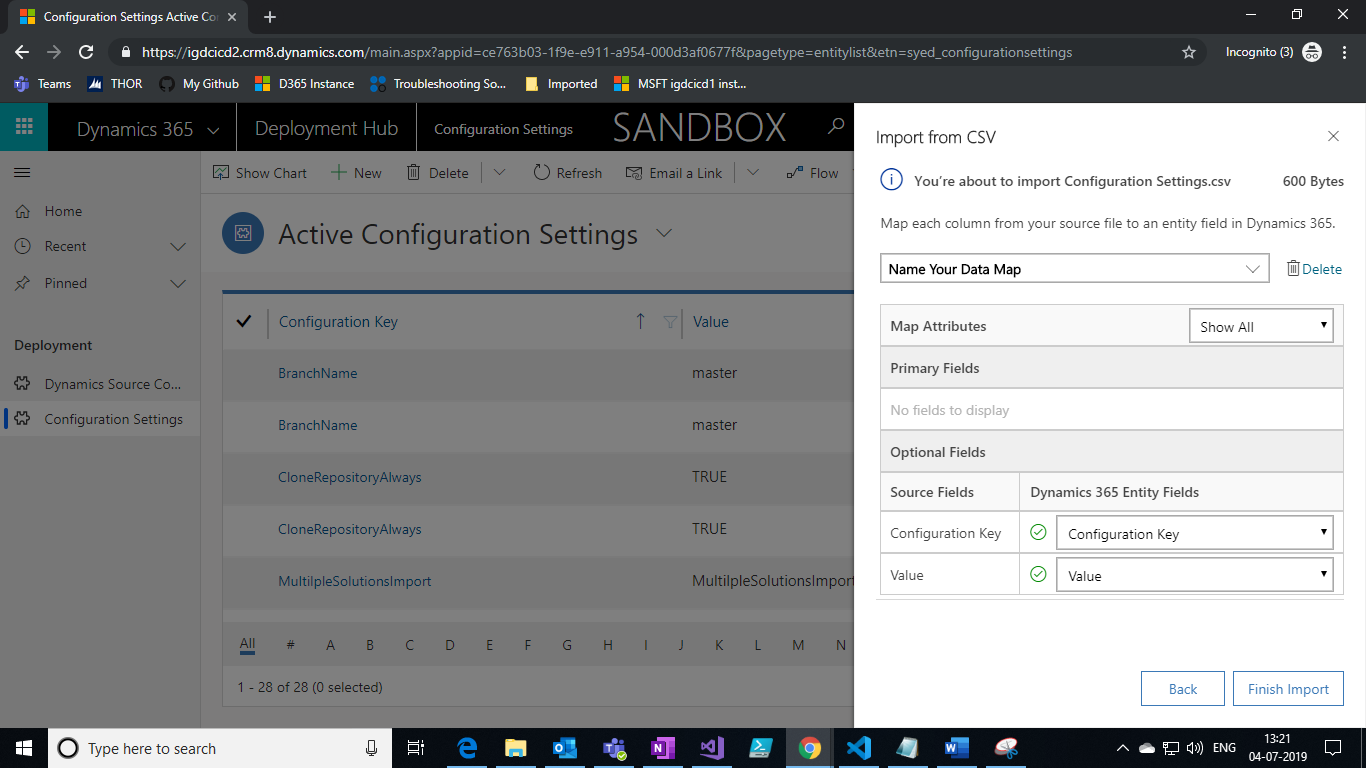
1. Go to top right click on … button and select Import from CSV.



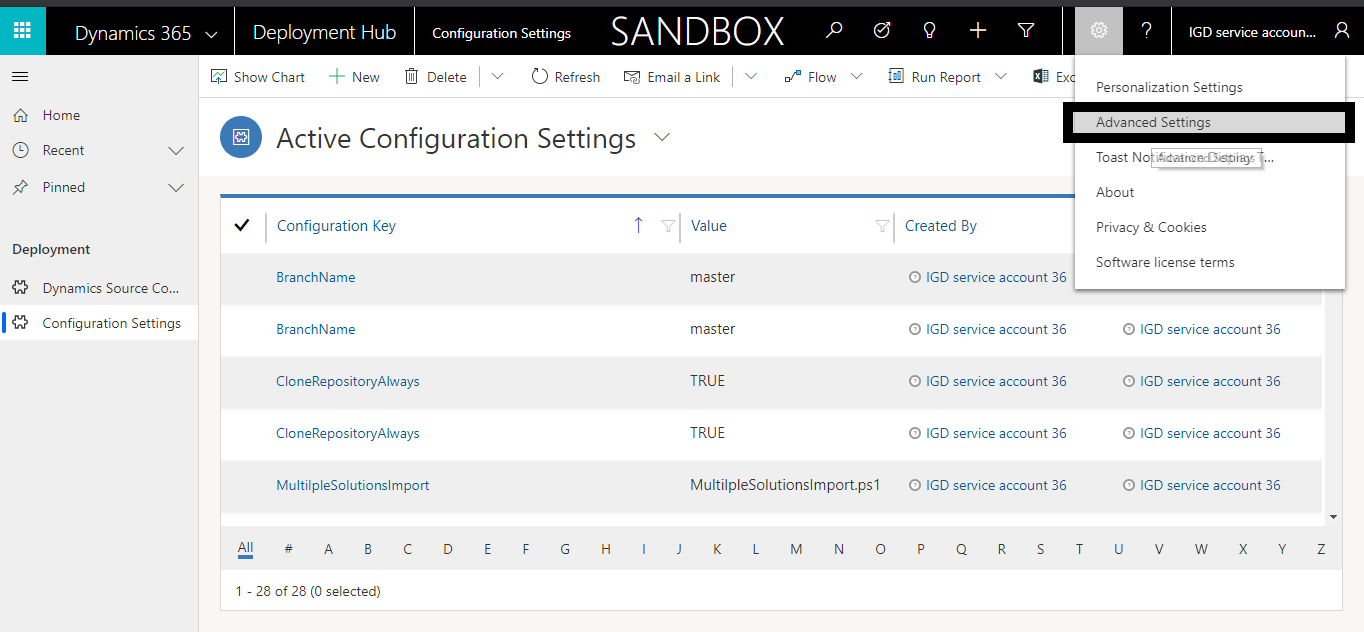
1. After clicking on Import from CSV, wizard gets opened.



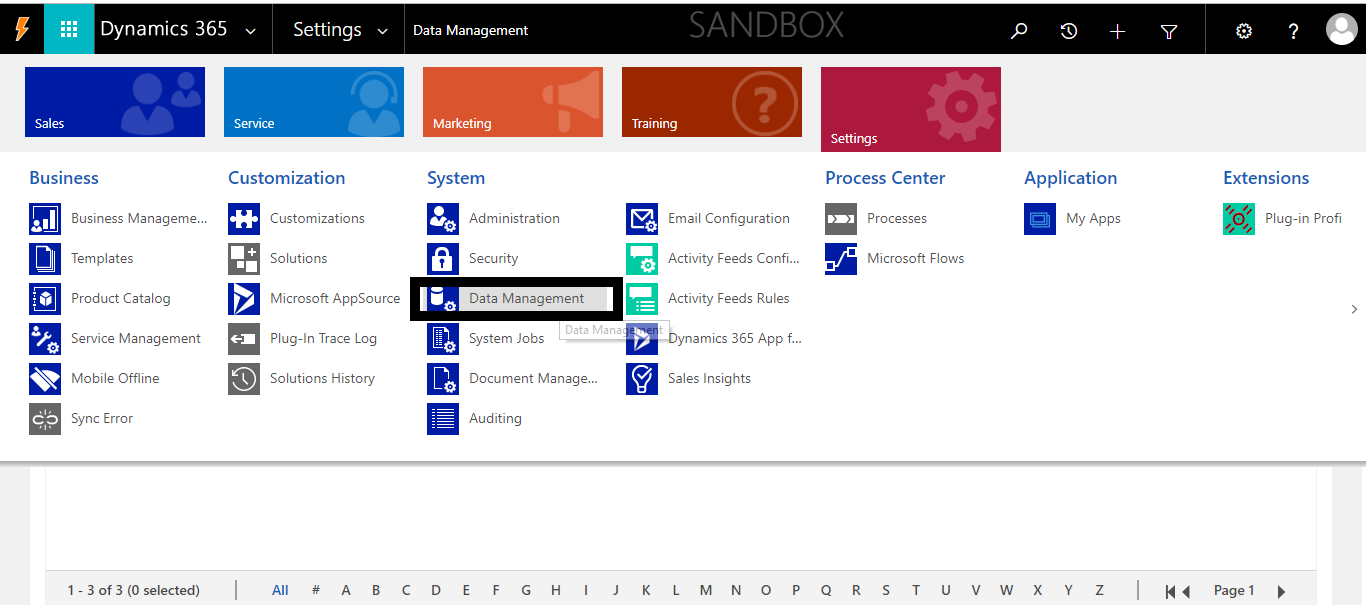
1. Select configuration settings csv file and click on Next.
2. Review mapping and finish the import.



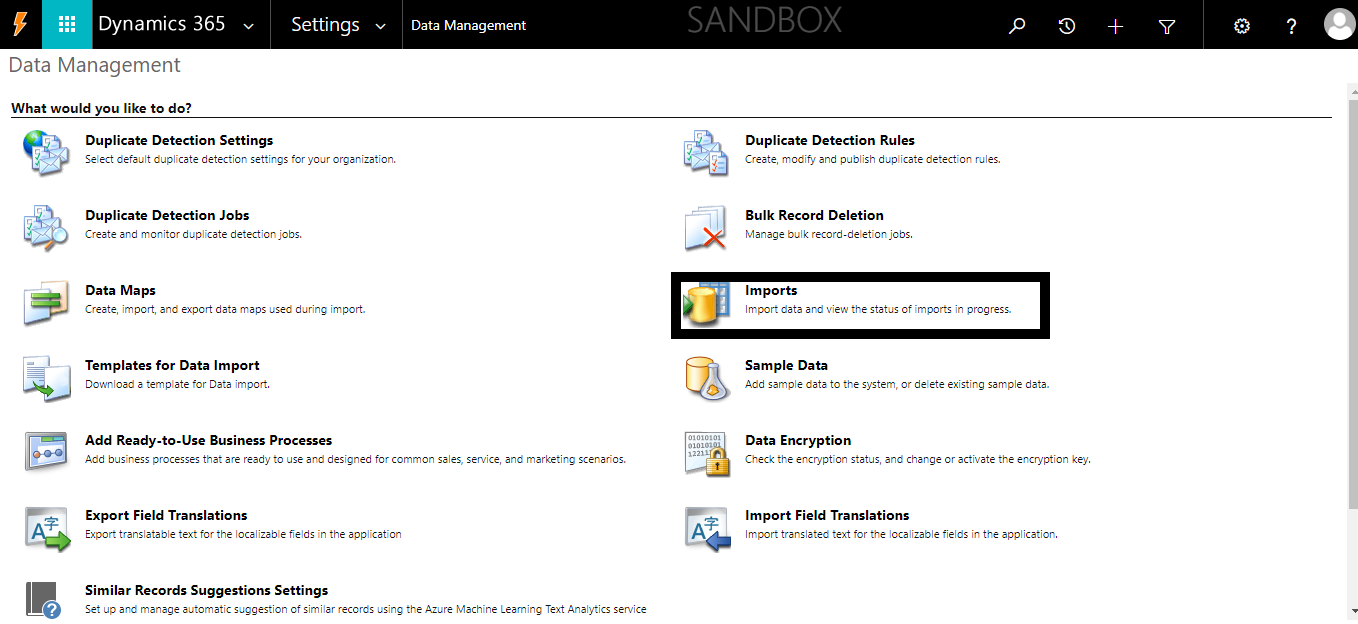
1. To check the status of import, click on setting button and select Advanced Settings it will navigate to settings page.



1. Follow step to check the status of import.
2. To check the status of import, Navigate to Settings->Data Management->Imports

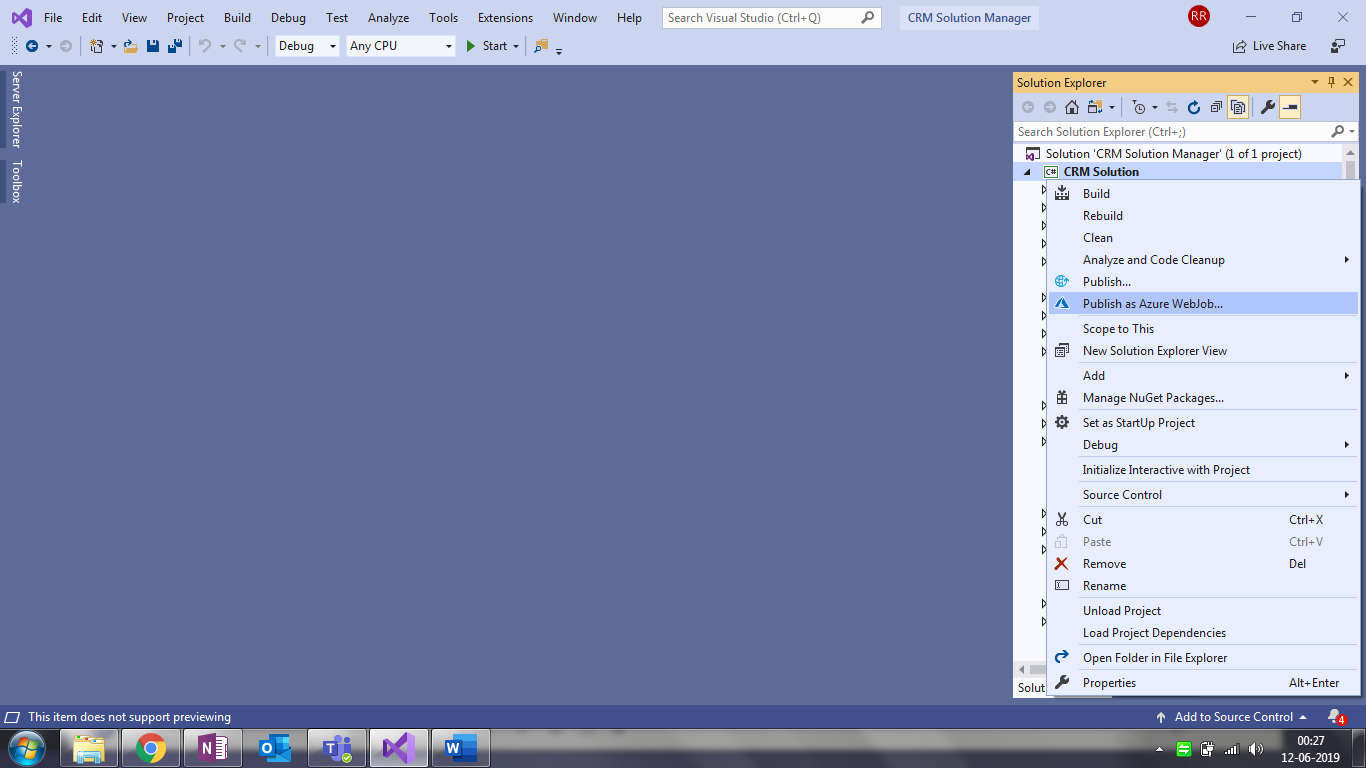


1. Click on Imports and check the status.

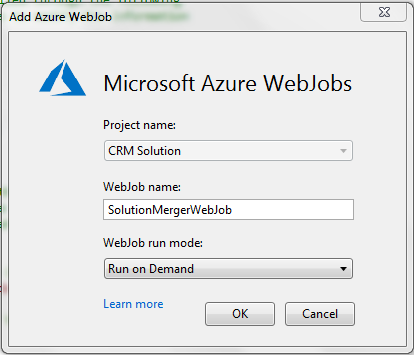


## Publishing Console Application as an Azure WebJob:

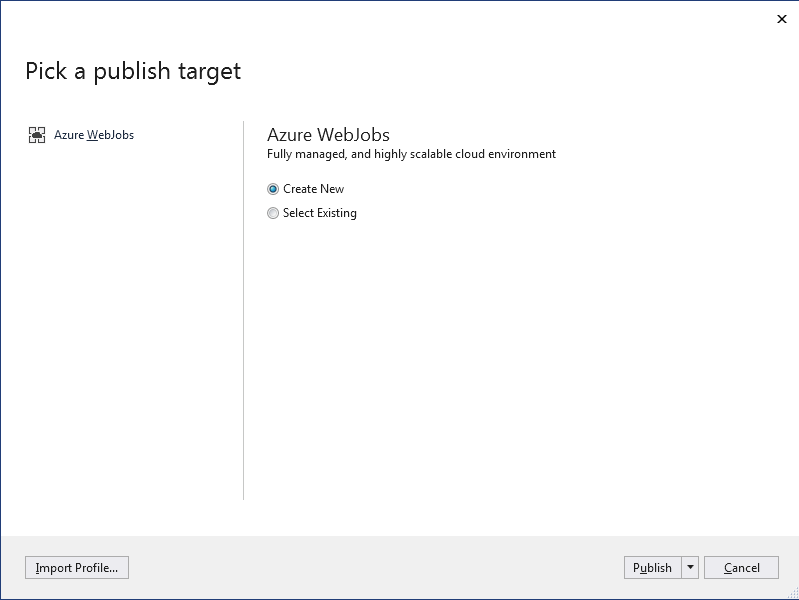
1. Open **‘CRM Solution Manager’** Solution in Visual Studio and provide values where keys are blank in **App.config** file.
2. Build the solution
3. After Successful build, go to Solution Explorer and right click on ‘CRM Solution Manager’ Project and then click on **‘Publish as Azure WebJob’**:



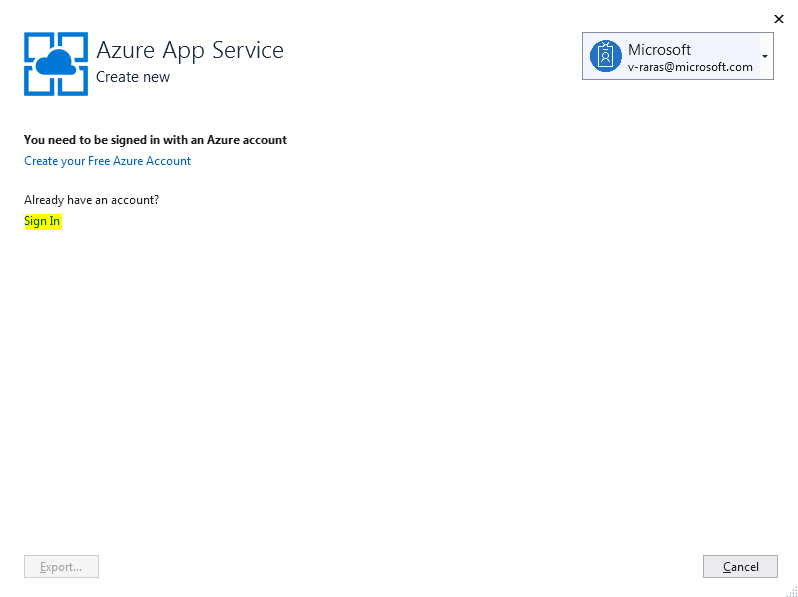
1. On the next screen, under **‘WebJob Name’**, give a name to the WebJob and select **‘WebJob run mode’** as **‘Run on Demand’** and click **OK** button:



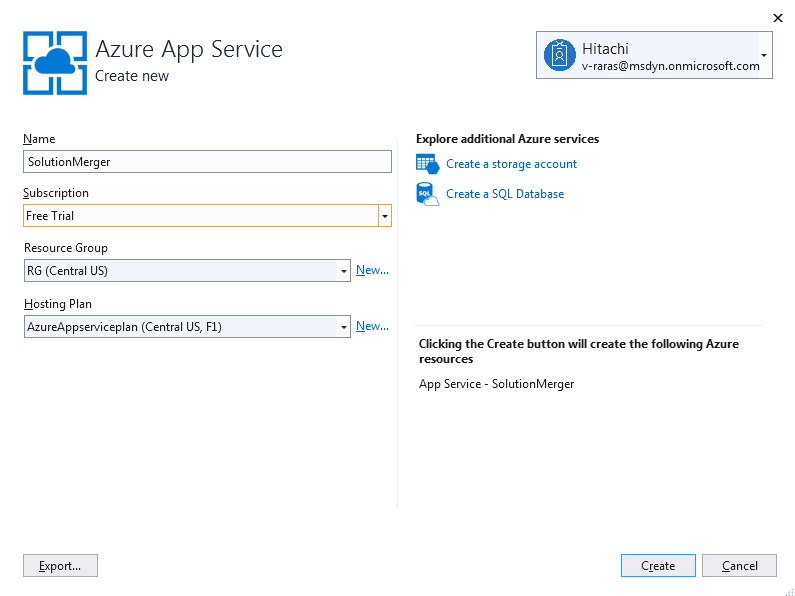
1. On the next screen, select **‘Create New’** and click on **‘Publish’**:



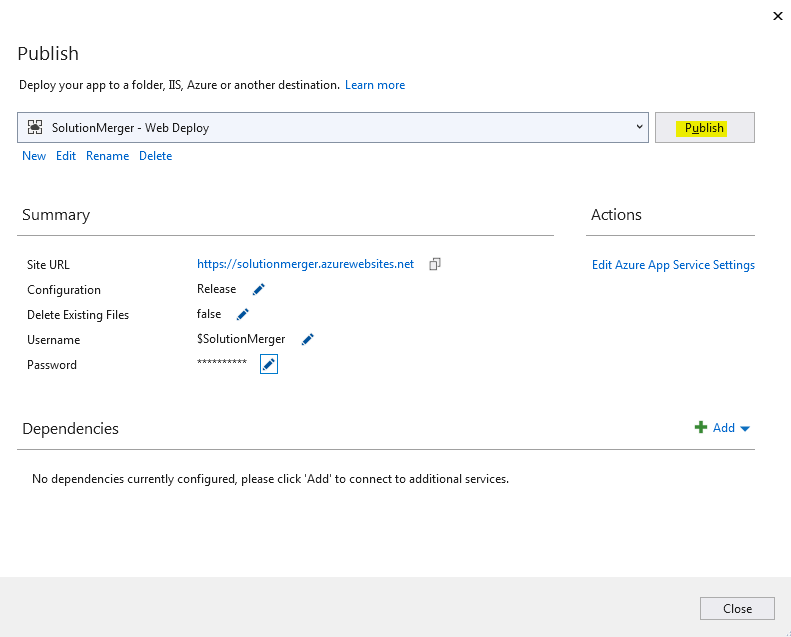
1. On the next screen, sign in to your azure portal by clicking on **Sign In** button:



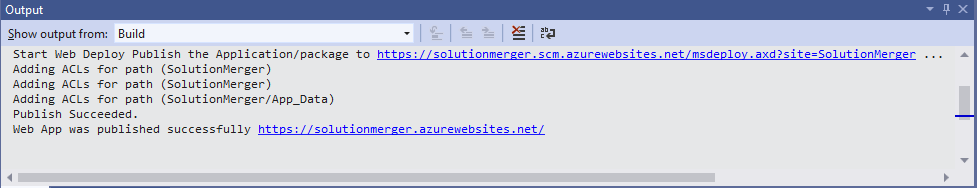
1. On the next screen, under **‘Name’**,give a name to the App Service, select **Subscription** As **‘Free Trial’**, select **Resource Group** for the App Service and click on **Create**:



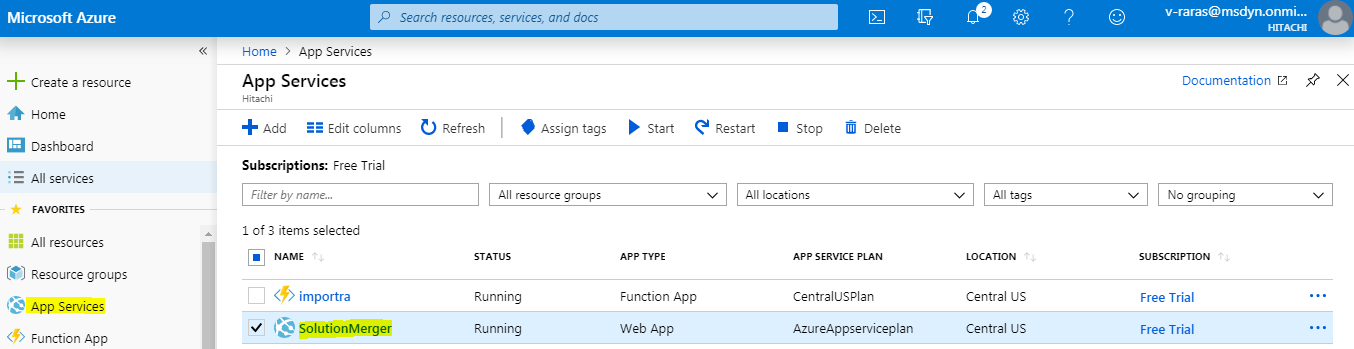
1. On the next screen, click on **Publish**:



This will **Publish the WebJob** **(Web App)** in **Azure**. You can see the output message in **Output Window** in Visual Studio:

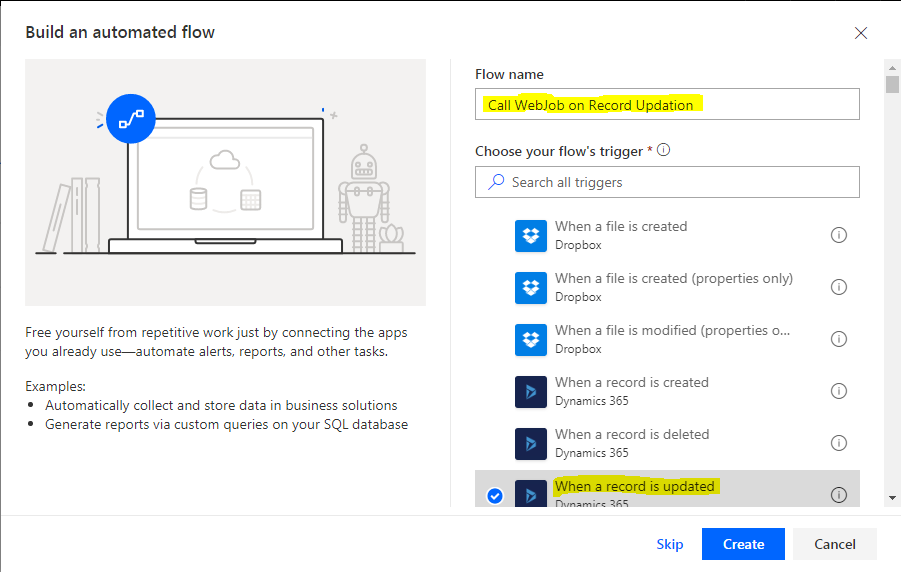


You can login to your Azure Portal and should be able to see your recently created **Webjob** under **App Services** tab:

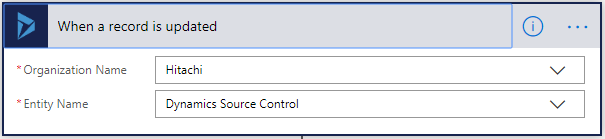


## Creating Microsoft Flow to call Azure WebJob:

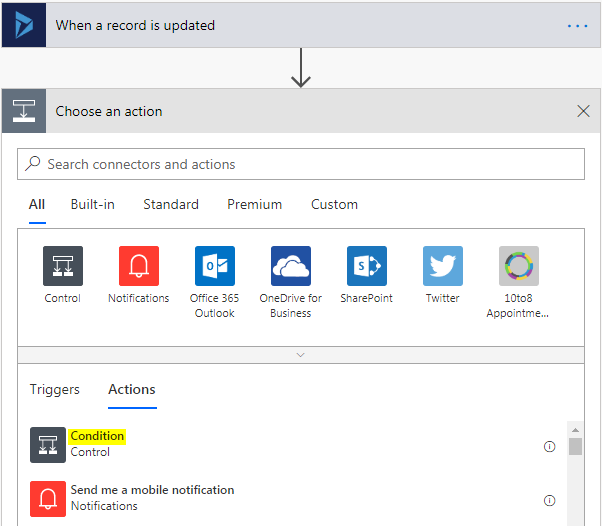
1. Go to [https://india.flow.microsoft.com](https://india.flow.microsoft.com/) and login to flow using CRM credentials where **DevOps** solution is installed.
2. Next, Click on **My Flows** tab under left pane, click on **New** and select **Automated-from blank**.
3. On the next screen, under **Flow Name**, give a name to your flow, under **Choose your flow’s trigger**, select **When a record is updated** and then click **Create**:



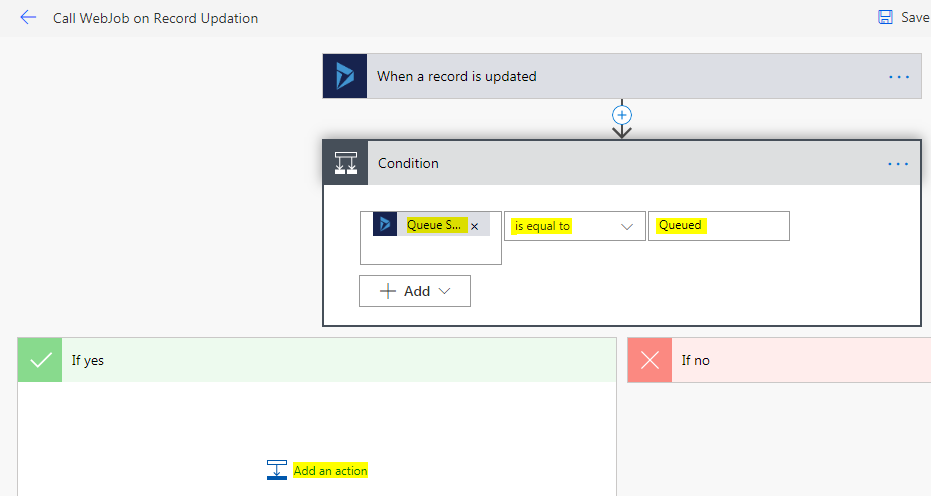
1. On the next screen, under **Organization Name**, select your organization from the drop down, under **Entity Name**, select **Dynamics Source Control** from the drop down and click **New Step**:



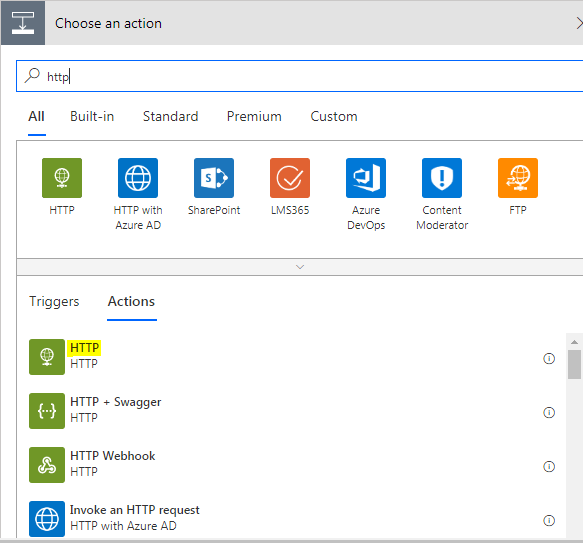
1. On the next screen, under **Actions,** select **Condition:**



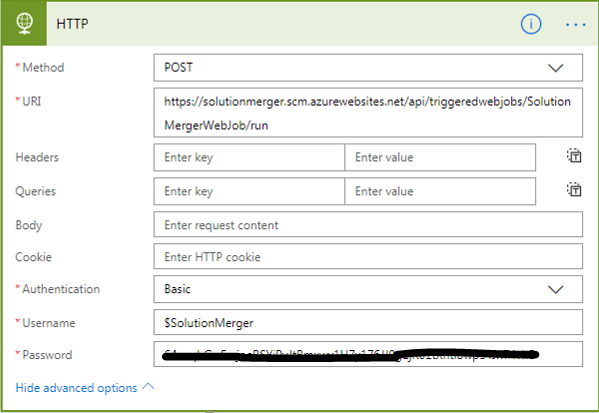
1. On the next screen, specify condition **Queue Status is equal to Queued** as given below and in **If Yes** step, click on **Add an action**:



1. On the next, screen, under **Actions,** search http and select **HTTP** Action:



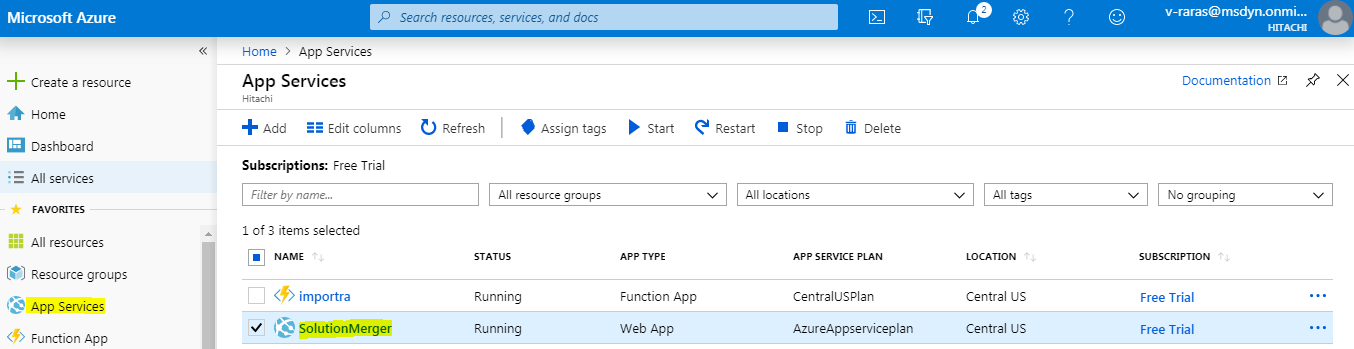
1. Choose **Method** as **POST**, under **Uri**, **Azure WebJob WEB HOOK** needs to be provided. You don’t need to set **Headers, Queries** and **Body**. Under **Advanced Options**, Choose **Authentication** as **Basic** and the **Username** and **Password** also need to be provided from **Azure WebJob** only. (Steps to get Azure WebJob WEB HOOK\*, Username\* and Password\* are provided at the last)



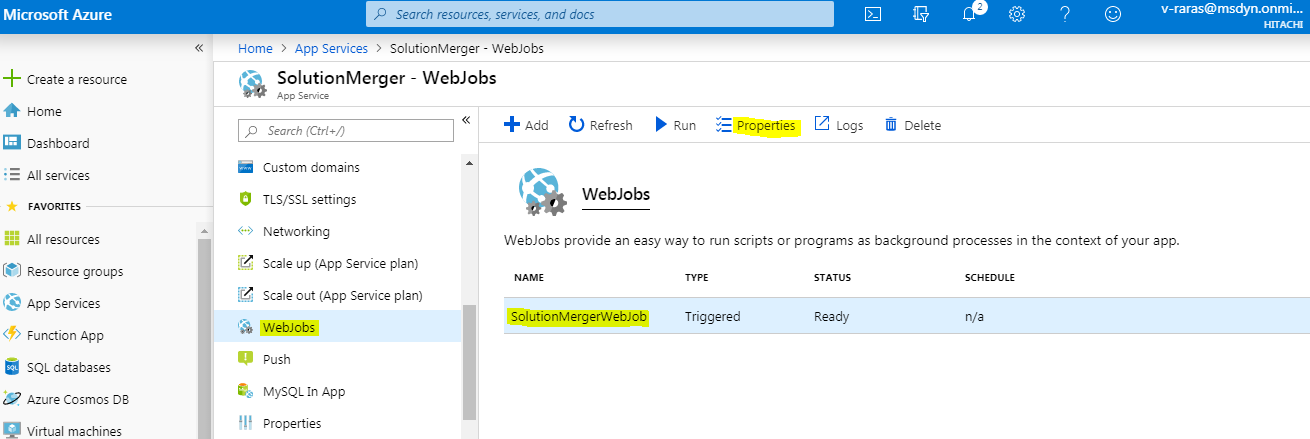
1. Next, click on **Save.** This will create the flow which will call our **Azure Webjob** when condition is satisfied.

**Steps To get Azure WebJob Uri, Username and Password:**

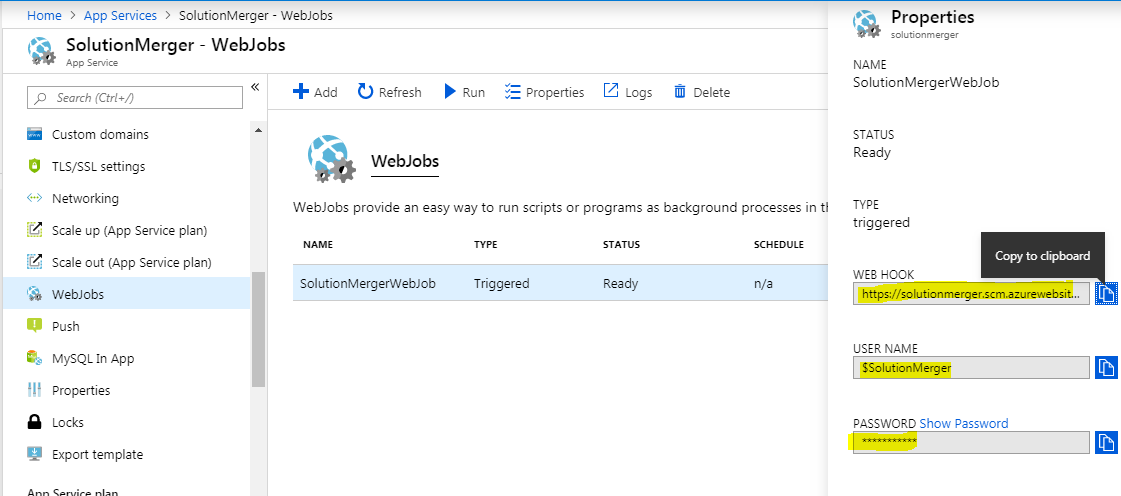
1. Login to Azure portal: [https://portal.azure.com](https://portal.azure.com/)
2. Click on your **Webjob** under **App Services** tab:



1. On the next screen, scroll down the middle pane, click on **Webjobs.** Now select the WebJob and click on **Properties:**

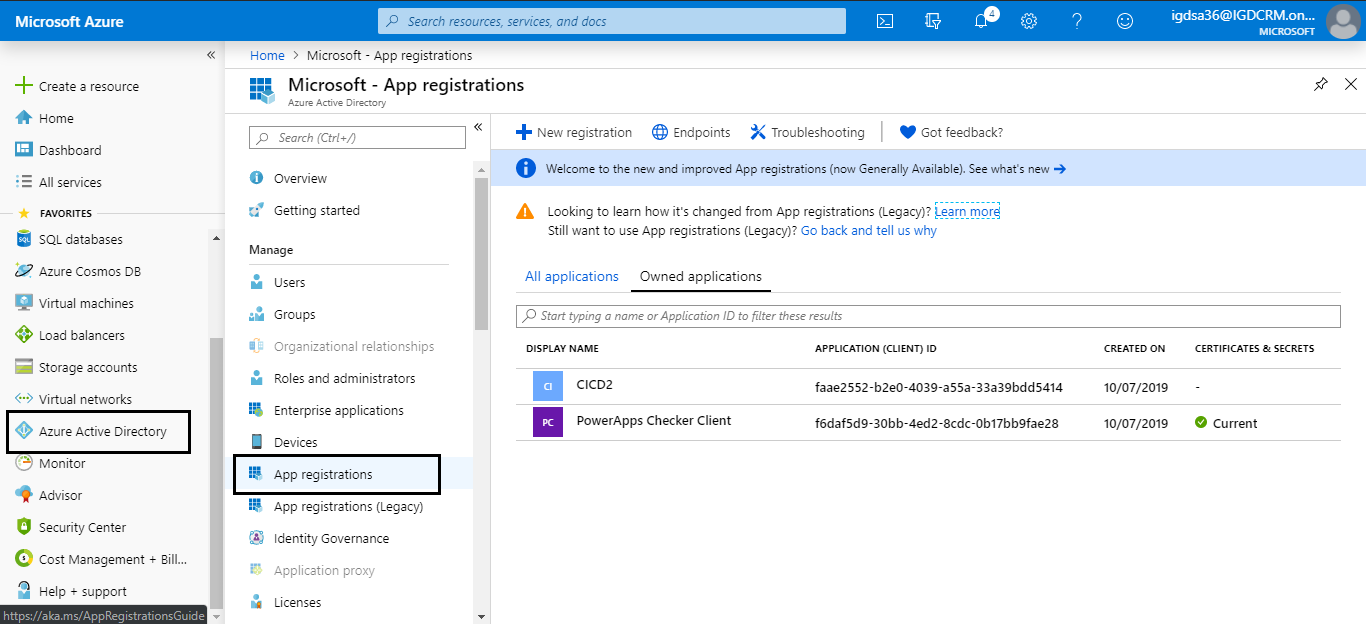


1. A new window will open at right, from where you can copy **WEB HOOK, Username and Password:**

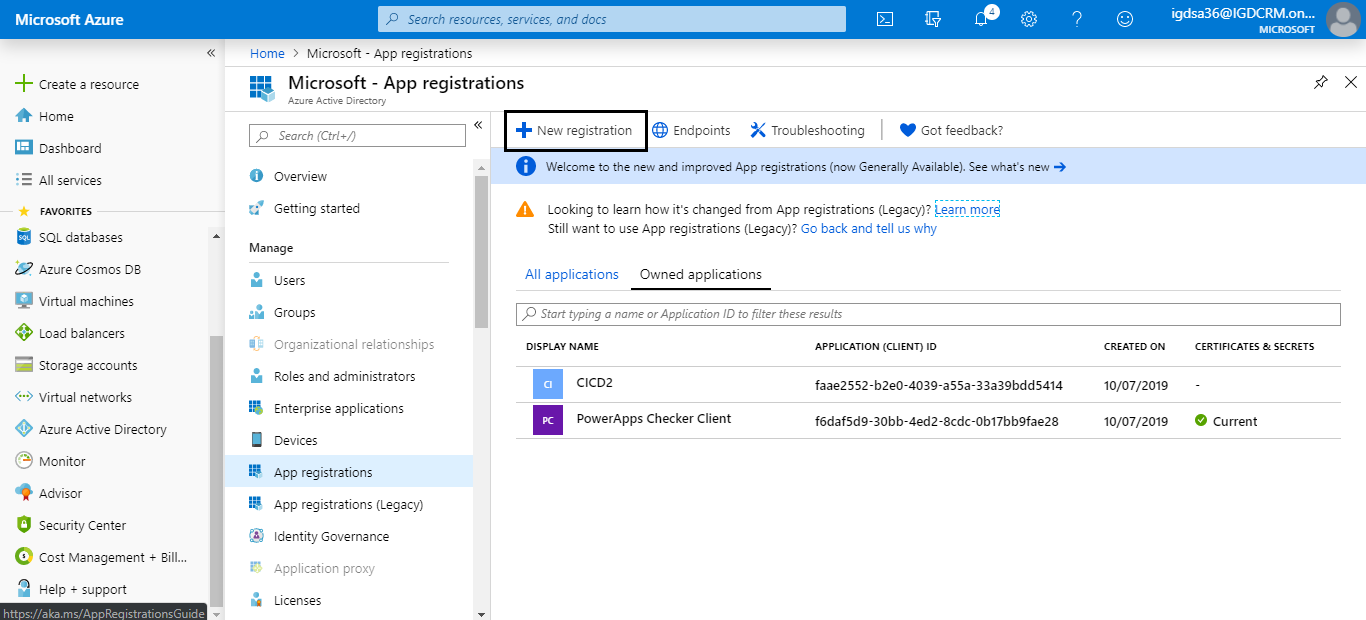


## Validate CRM solutions using PowerApps checker PowerShell module

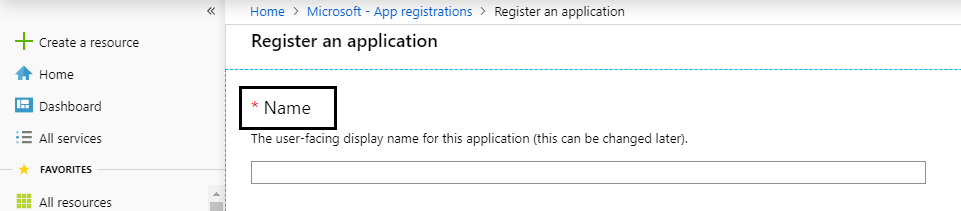
1. Login to azure portal.
2. After logged in successfully, navigate to Azure Active Directory and select App registrations.

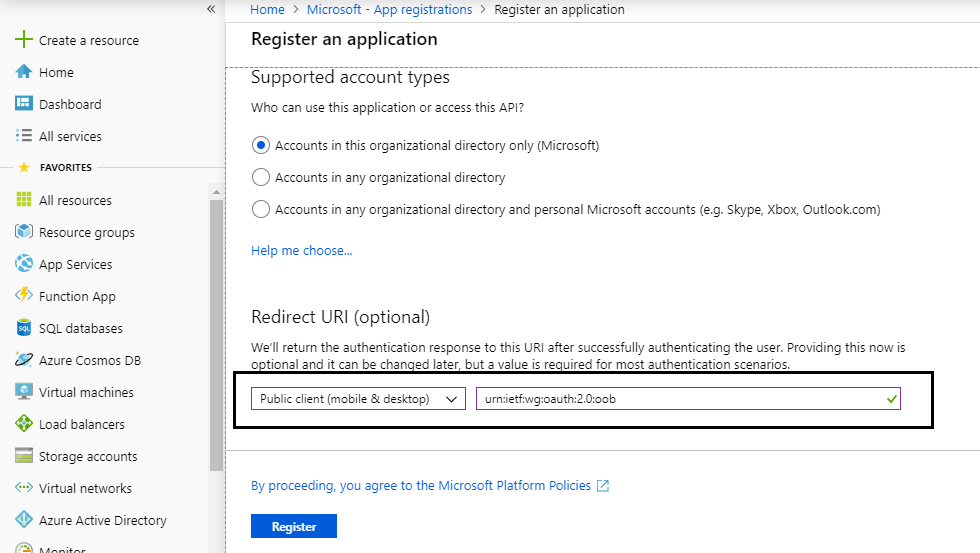


1. Click New registration.

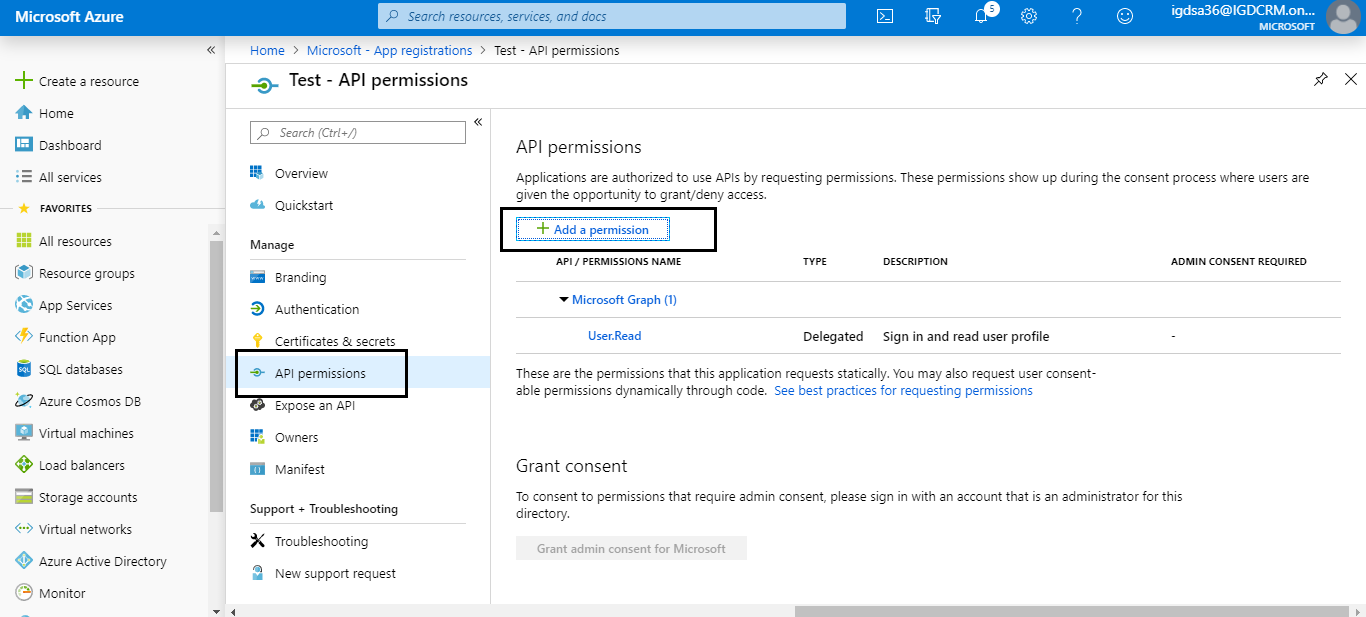


1. Fill Name and in Redirect URI select **Public client (mobile & desktop)** and fill **urn:ietf:wg:oauth:2.0:oob** click register.

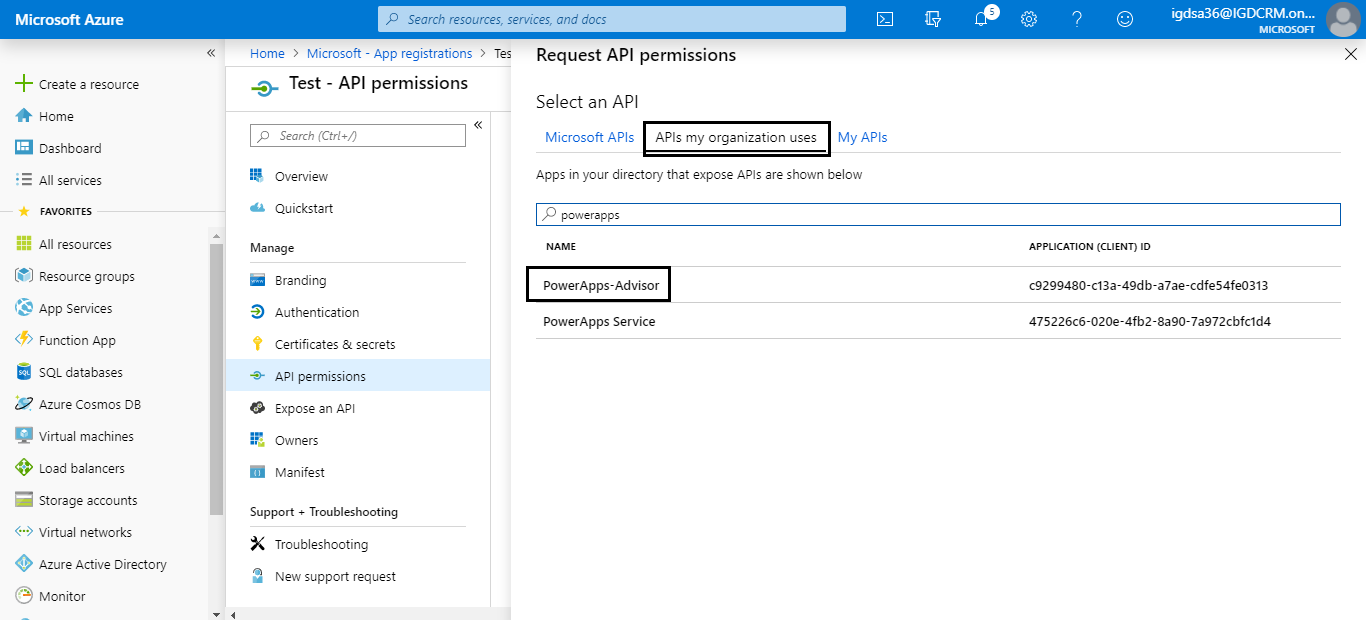




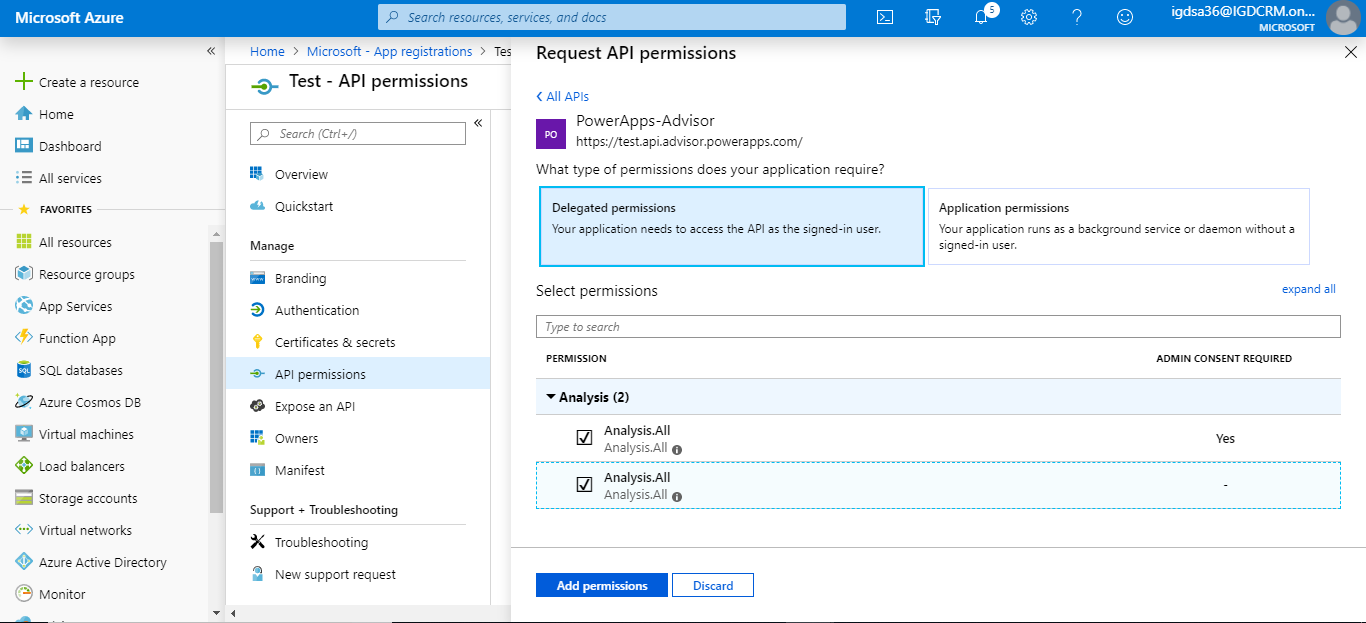
1. After successful register, click API permissions and add a permission for PowerApps advisor.



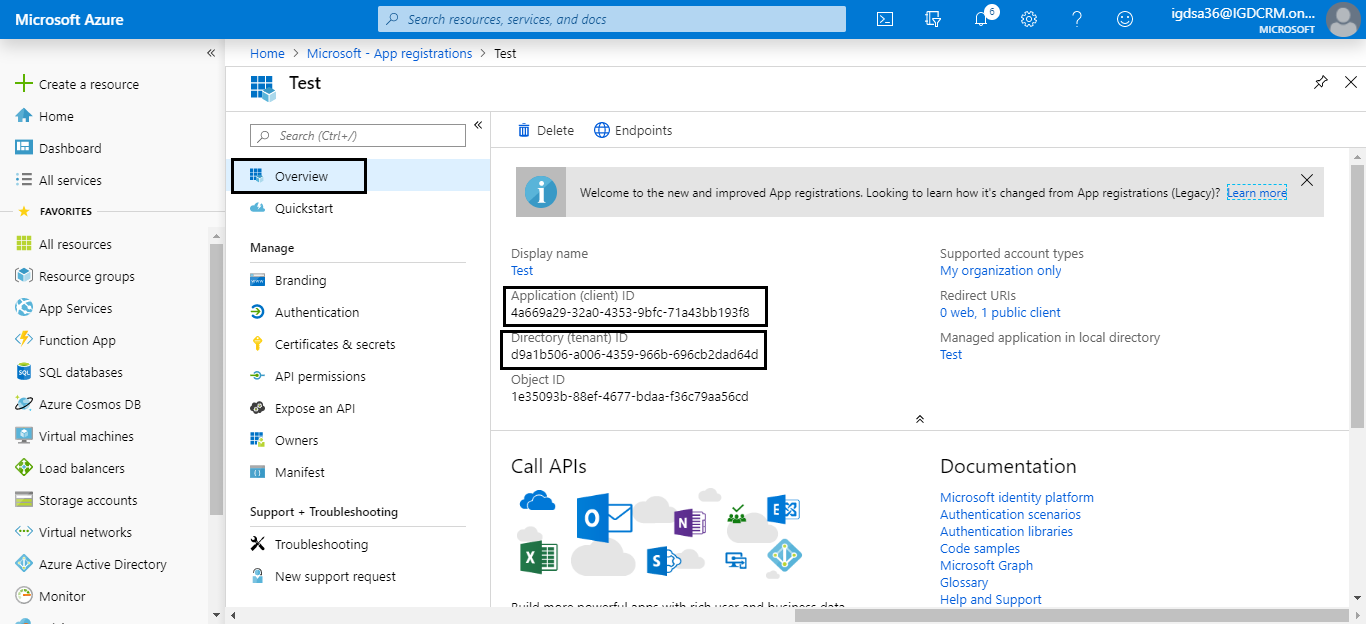
1. In Request API permissions go to APIs my organization uses search for “**powerapps**” and select PowerApps-Advisor.



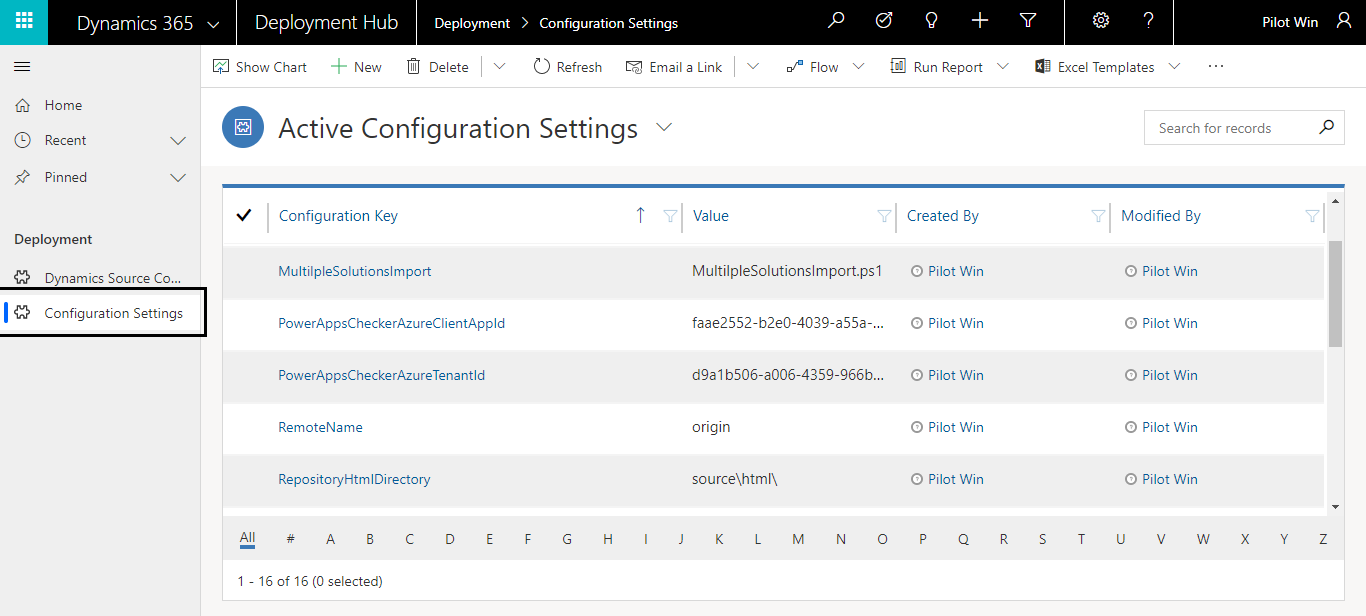
1. In type of permission select **Delegated permissions** and in permission select all and add permission.



1. Finally, go to Overview and copy client id, tenant id.



1. Login to CRM and navigate to configuration settings entity.



1. Copied client and tenant id should be filled in PowerAppsCheckerAzureClientAppId, PowerAppsCheckerAzureTenantId fields respectively.

