Module 5 - On-Premise Data Gateway

Lab 1

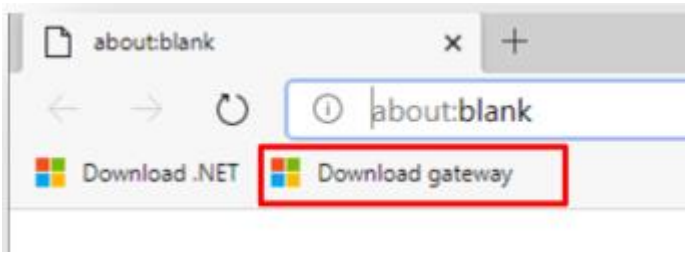
Estimated Time

20 minutes

Exercise 1 - Download and Install Data Gateway

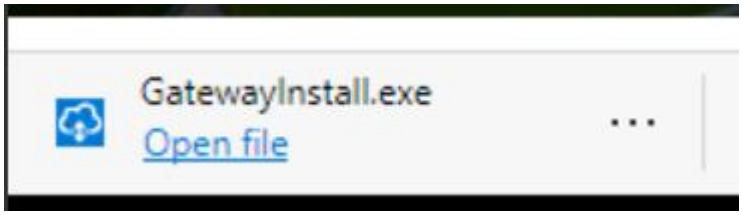
In this exercise, we will install and configure On-Premises Data Gateway

1. On **Virtual Machine** open the **Edge Chromium** browser and navigate to https://www.microsoft.com/en-us/download/details.aspx?id=53127

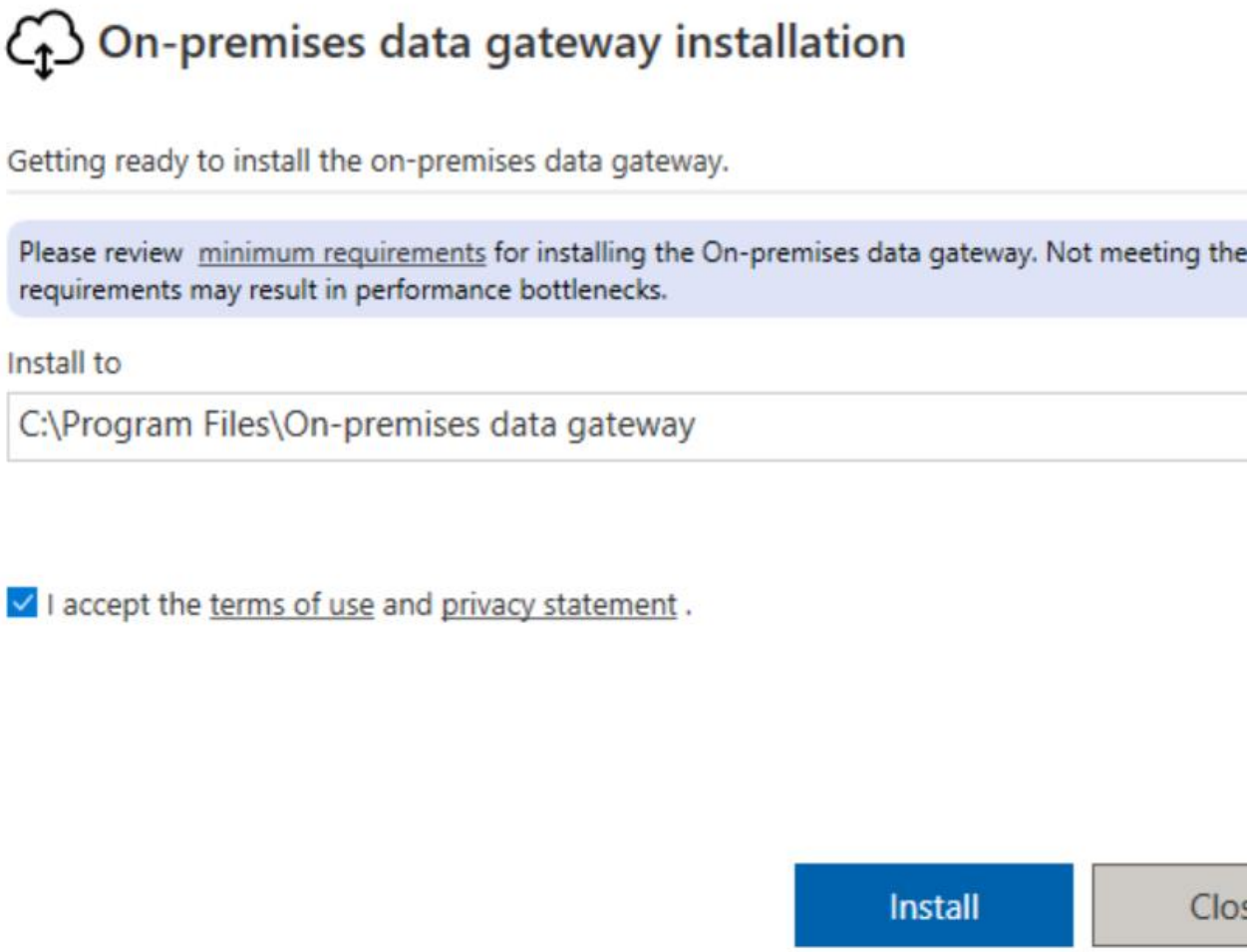


2. Select **Download** to start On-premise data gateway download

3. Run the Installation File.



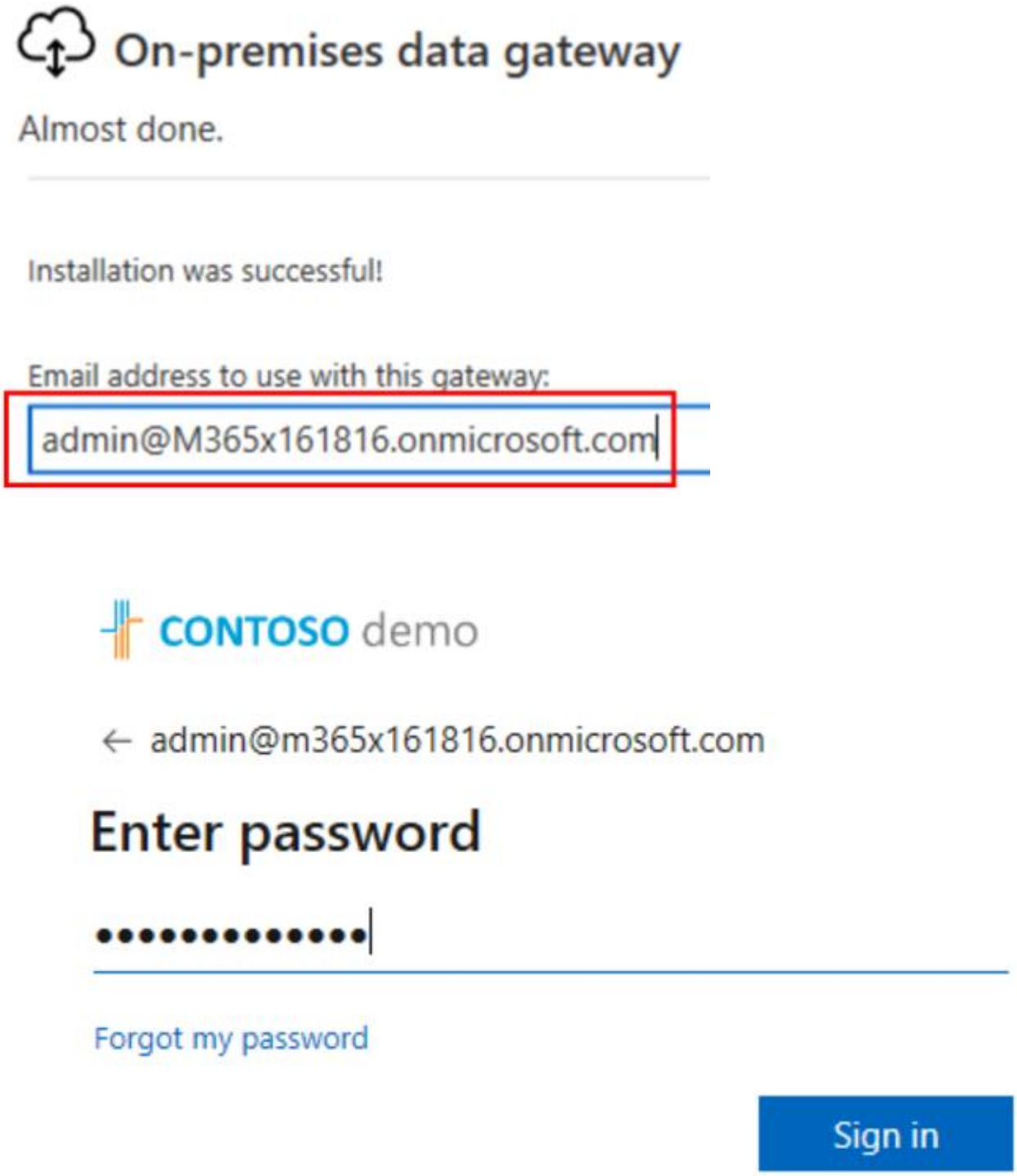
4. Review Terms of Use and then click **Install**.



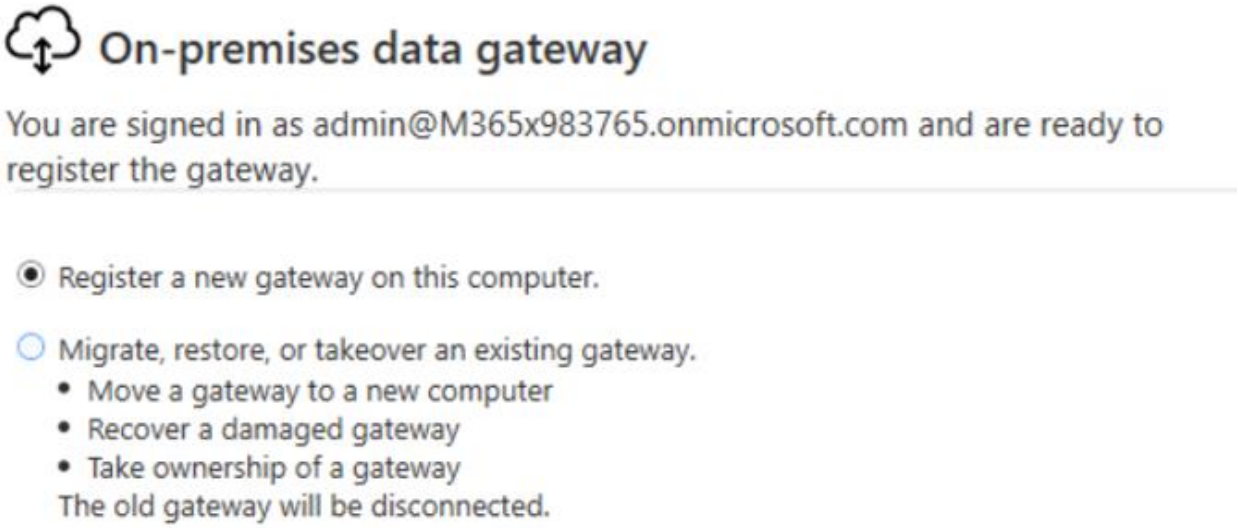
5. Sign-in using **MOD Administrator** account

   MOD Administrator Username

   MOD Administrator Password



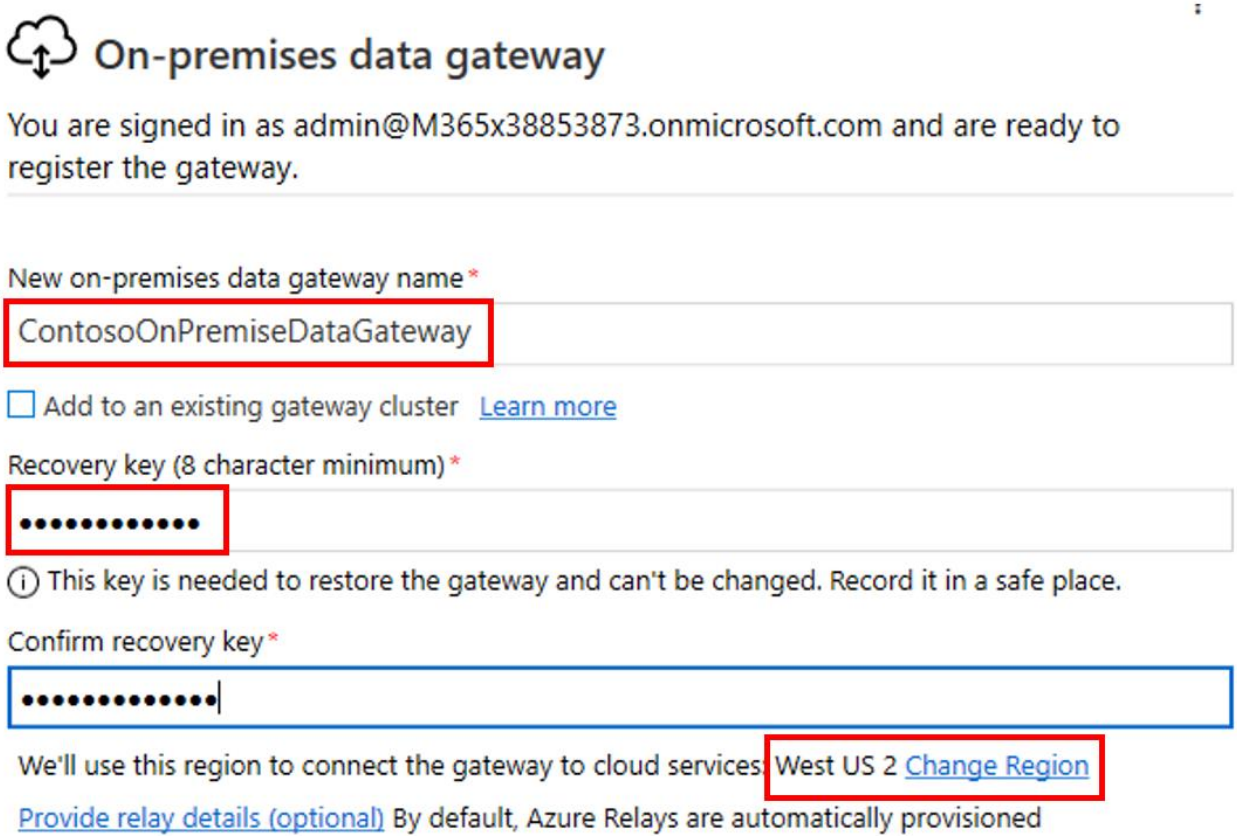
6. Register a new gateway on this computer, then click **Next**.



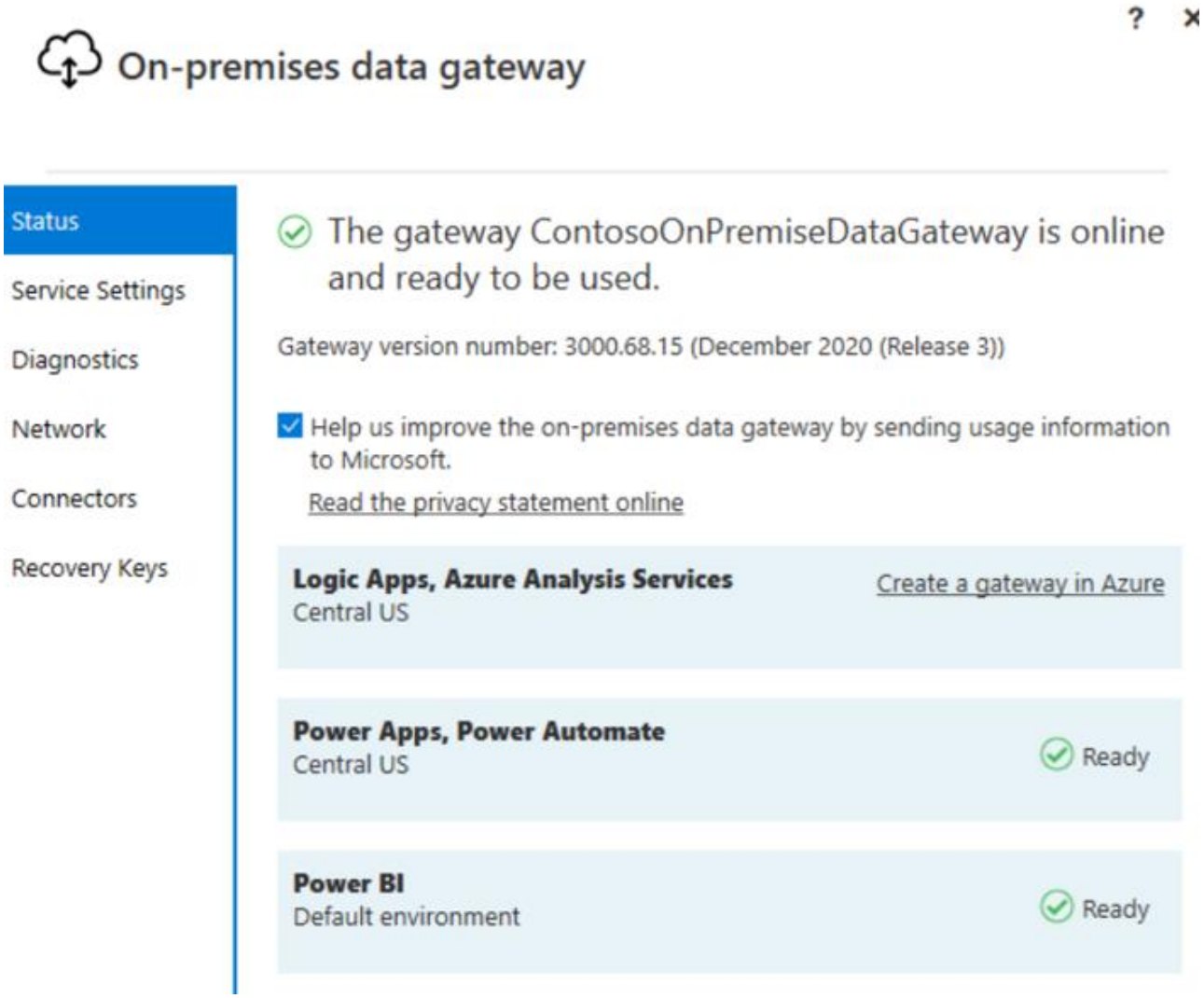
7. Type-in the gateway name and recovery key

    - Name = ContosoOnPremiseDataGateway
    - Recovery key = **MOD Administrator Password**.

Make sure that the selected data gateway region matches the targeted environment's region, by default the selected region will be Azure AD default region. Then click **Configure**.

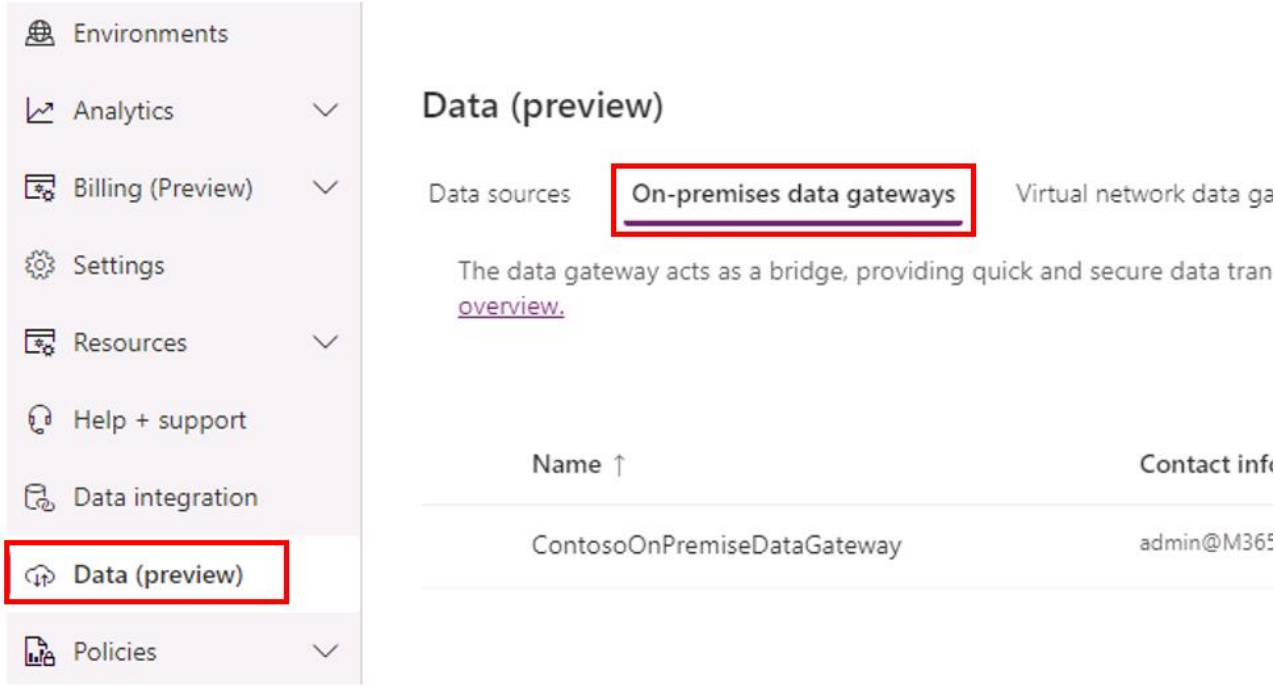


8. Data gateway is configured and ready to be used.



9. Open browser and navigate to Power Platform Admin Center **https://aka.ms/ppac**

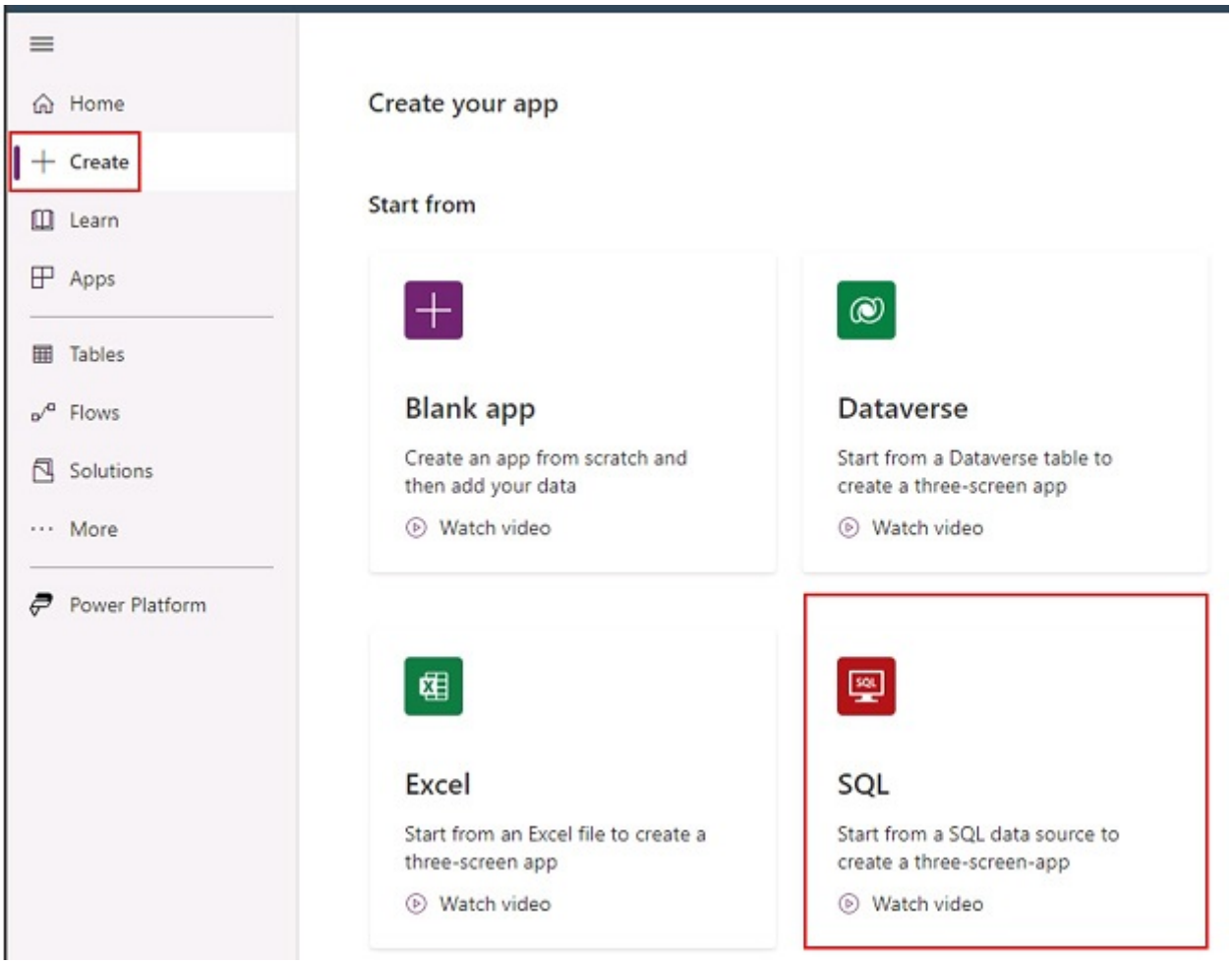
10. From the left menu, select **Data (preview)** > **On-premises data gateways** to see the gateway we installed



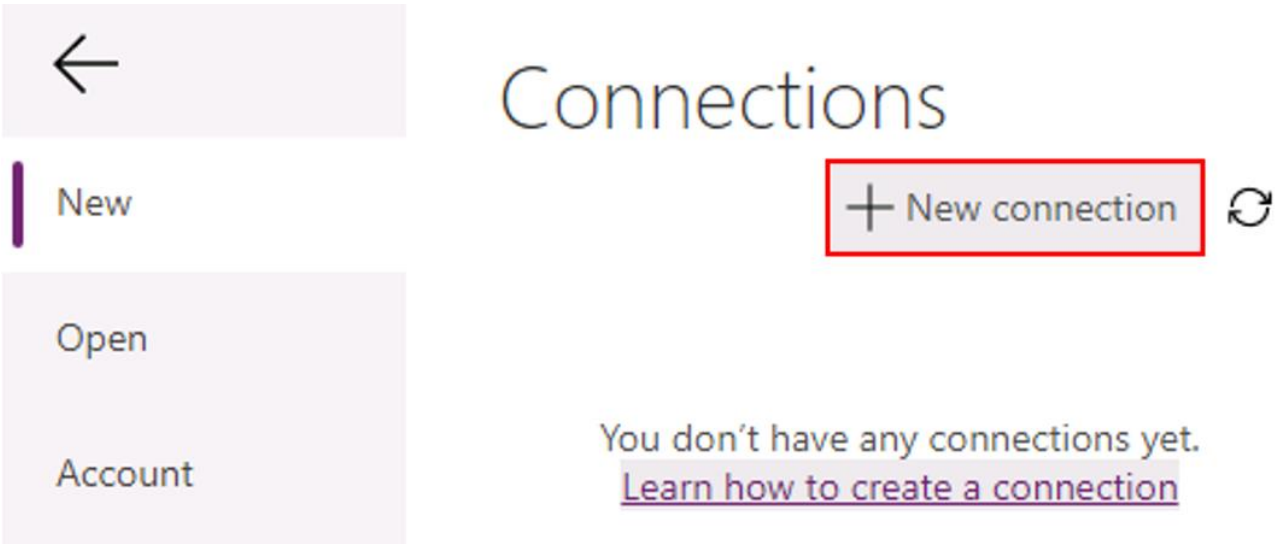
Exercise 2 - Build an app using On-premise data gateway

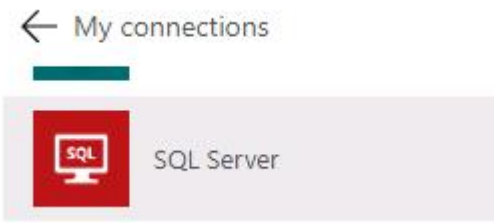
In this exercise, we will build a Power App which connects to SQL Server database and returns data from one database table to the app

1. Navigate to **http://make.powerapps.com** then switch to the **Contoso (default)** environment.
2. Select **+ Create** from the left menu and then from the **Start from** select **SQL**



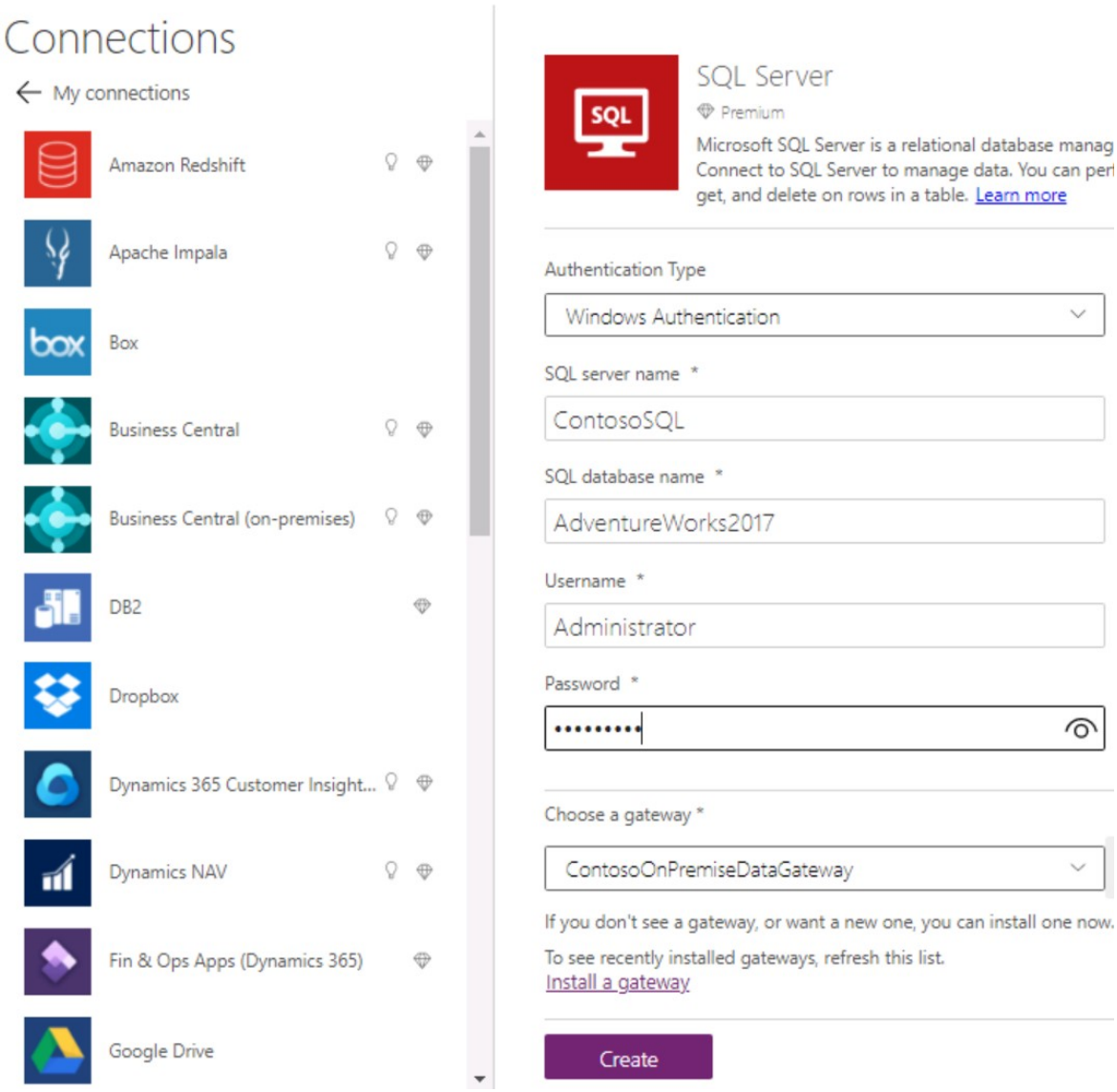
3. First you need to create new SQL Server connection, so click **New connection** and select **SQL Server**



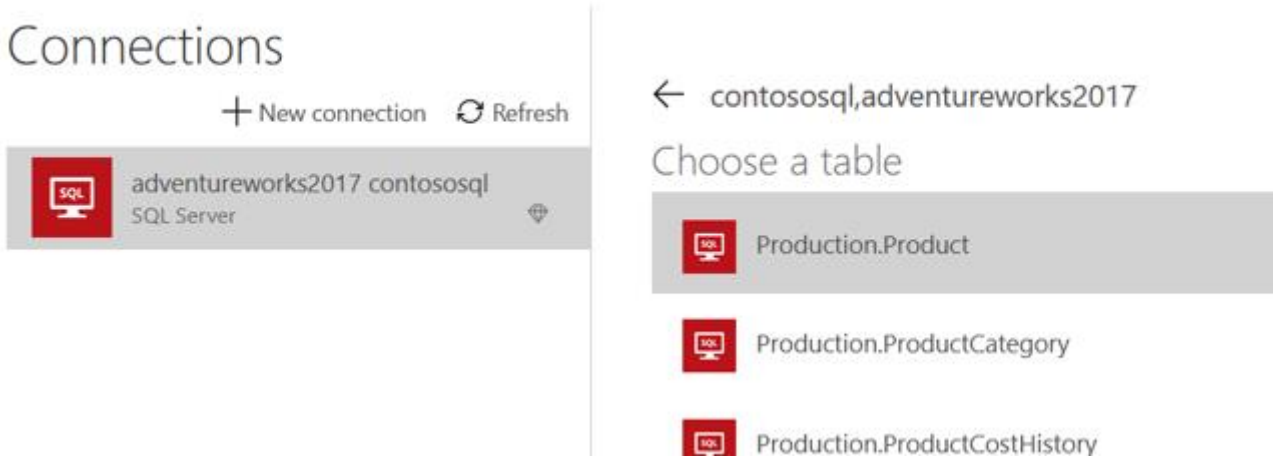


4. Type the required connection information like below

    - Authentication Type: **Windows Authentication**
    - SQL server name: **ContosoSQL**
    - SQL database name: **AdventureWorks2017**
    - Username: **Administrator**
    - Password: **VM Password**
    - Choose a gateway: **ContosoOnPremiseDataGateway**
    - Then select **Create**

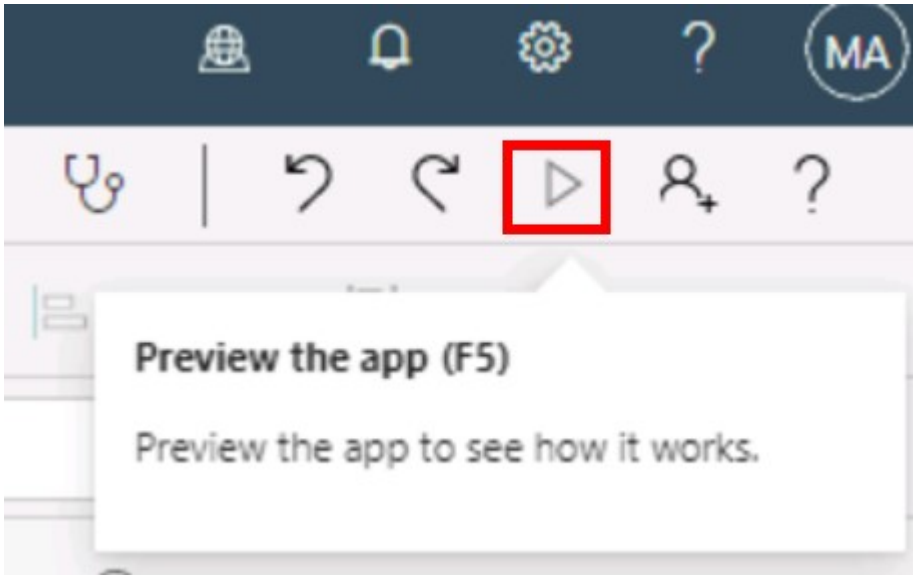


5. At the created connection, search for **Production.Product** table then select the table and click Connect.



5. Power Apps will start creating an app managing all CRUD operations for the selected table using SQL server on-premises.

6. Run the app by clicking the Play button on top-right corner (or press F5). Test, for example, how sort, search and edit item operations works

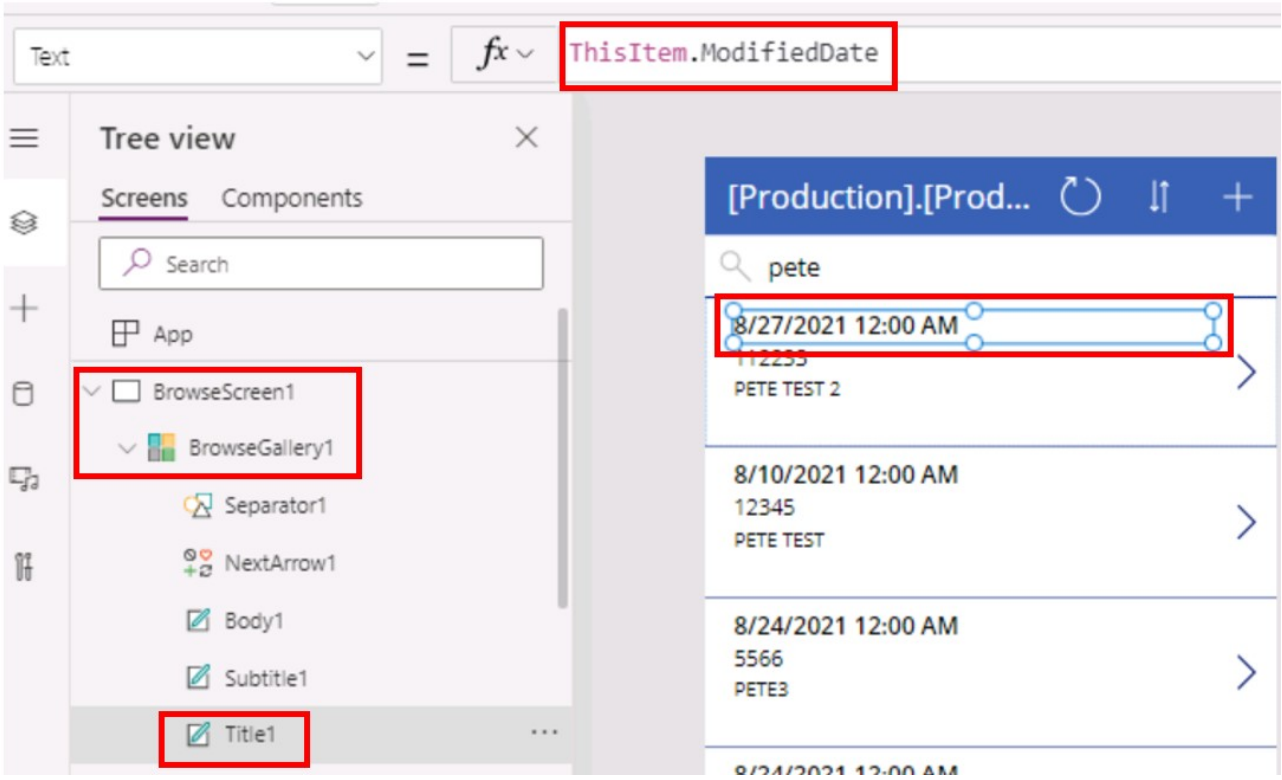


NOTE: Adding new items does not work in generated by default because mandatory ROWGUID column is not created to the form. This type of column is not yet supported by Power Apps. To get insert operation working, you need to change the submit icon on top-right corner OnSelect event formula of the EditScreen. Instructions how to get insert working are in the next optional tasks, so if you have time then continue to Exercise 3

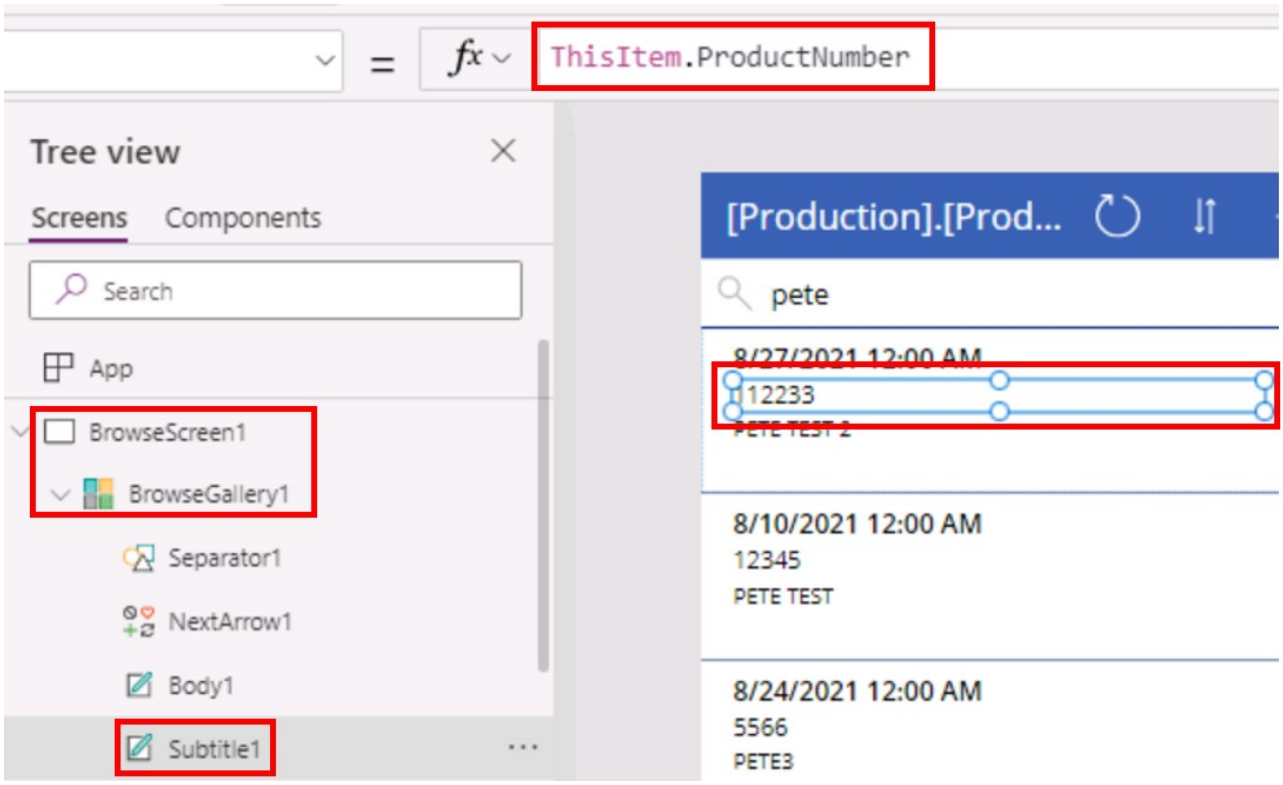
Exercise 3 (OPTIONAL) - Add insert new product functionality to the app

In this exercise, we will change how new product is saved into the database

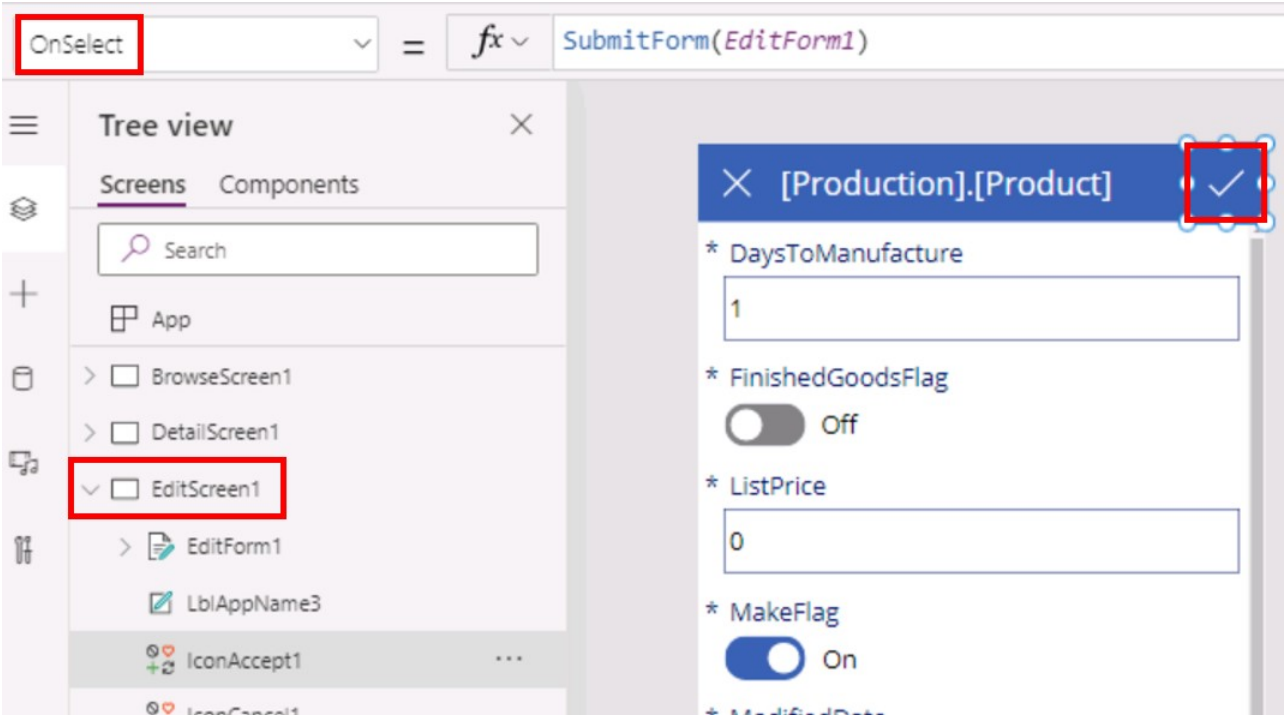
1. On **BrowseScreen1** screen, open the **BrowseGallery1** control, select **Title1** field and then Change its **Text** property to **ThisItem.ModifiedDate**



2. In a similar way, change the **Subtitle1** field **Text** property to **ThisItem.ProductNumber**

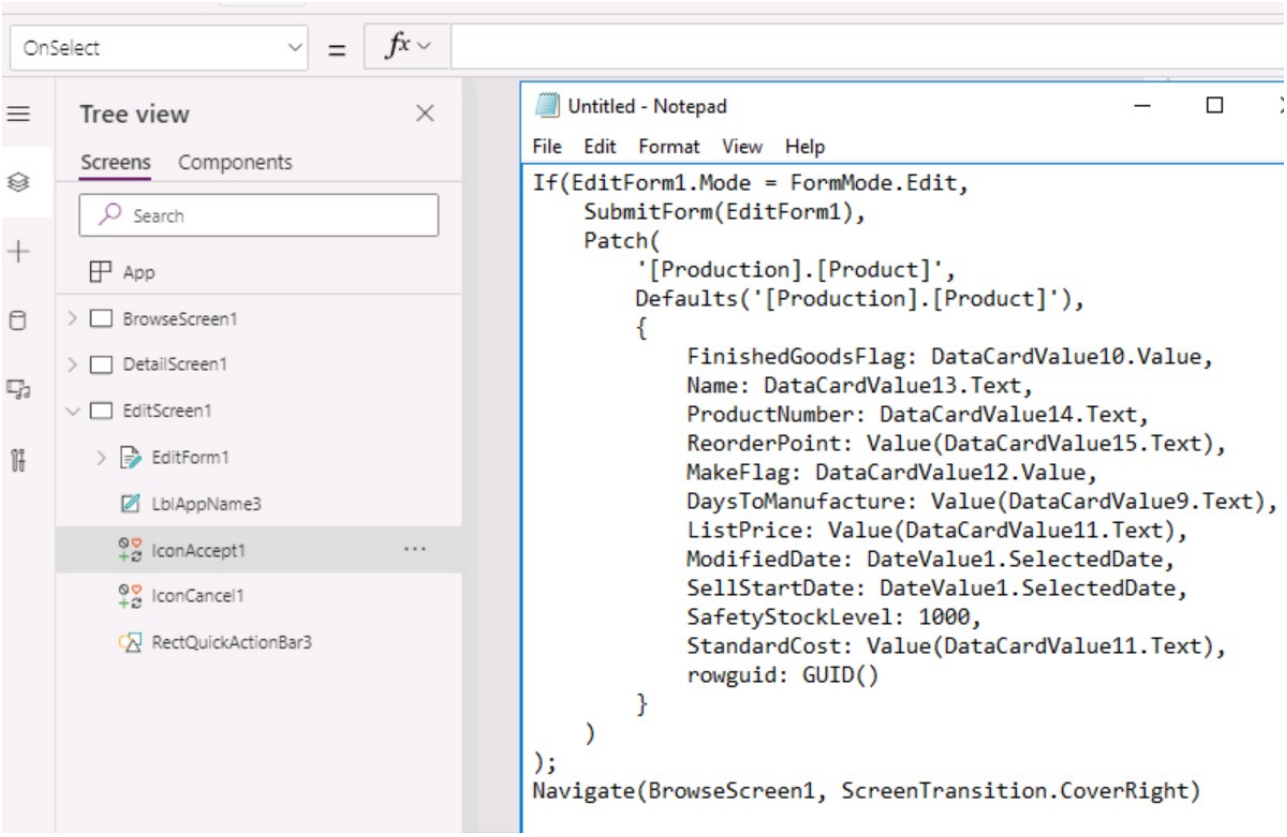


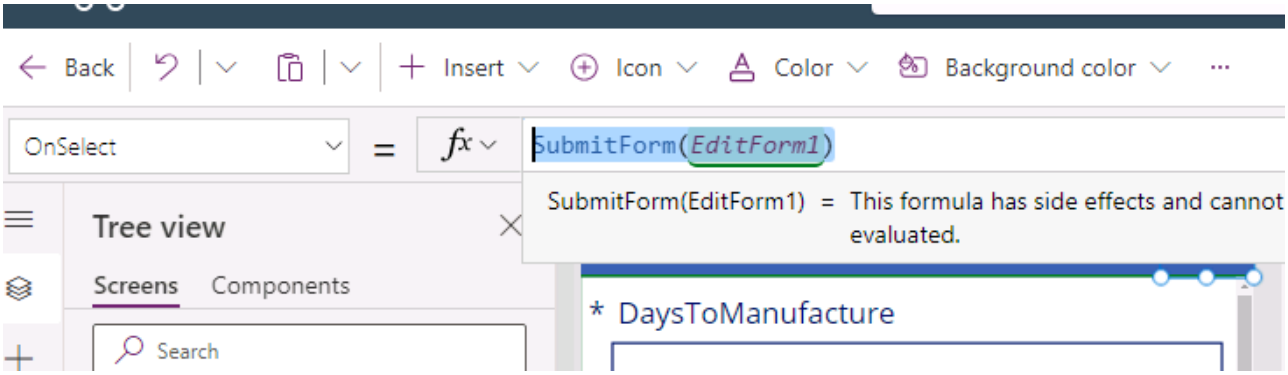
3. Select **EditScreen1** screen and then select the submit icon on top-right corner (accept icon). Verify that **OnSelect** event in formula bar is selected

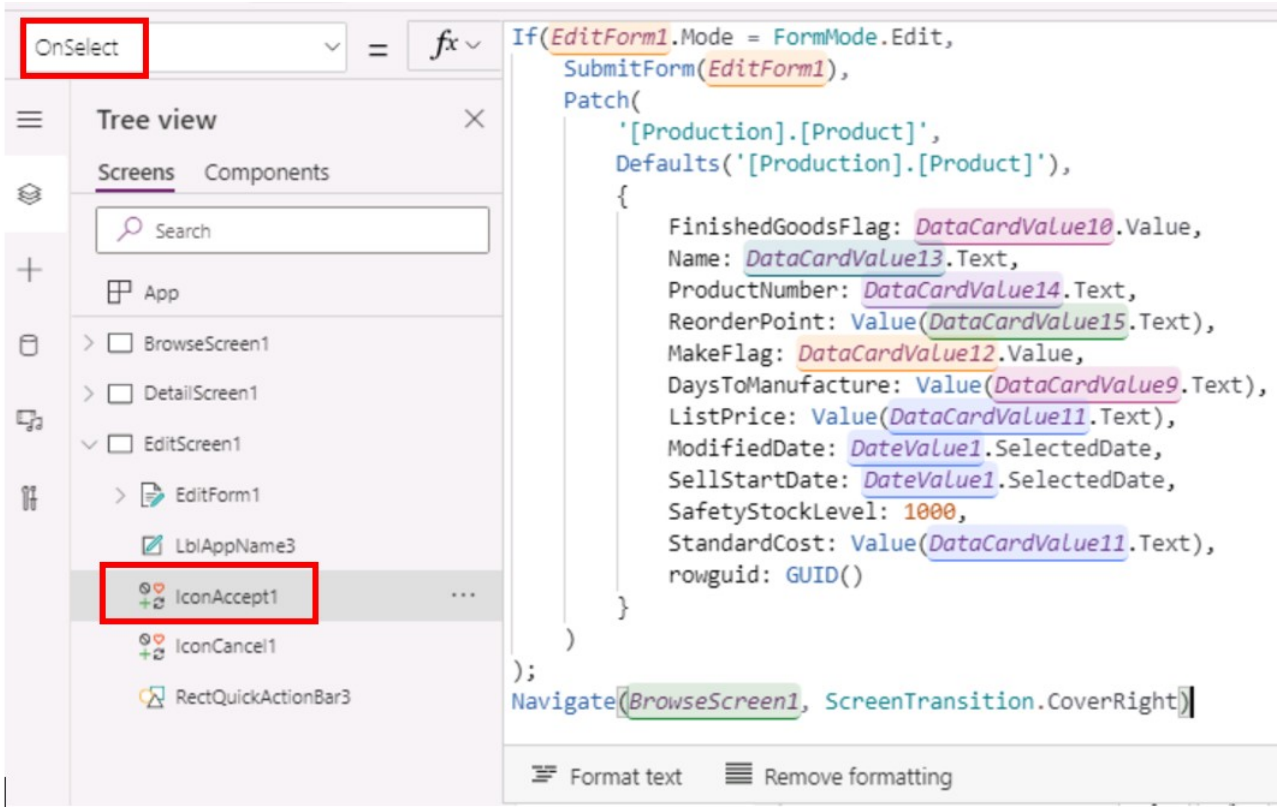


4. Copy-paste the formula below first to **notepad** and from there to **OnSelect** event by **replacing** the existing formula SubmitForm(EditForm1)

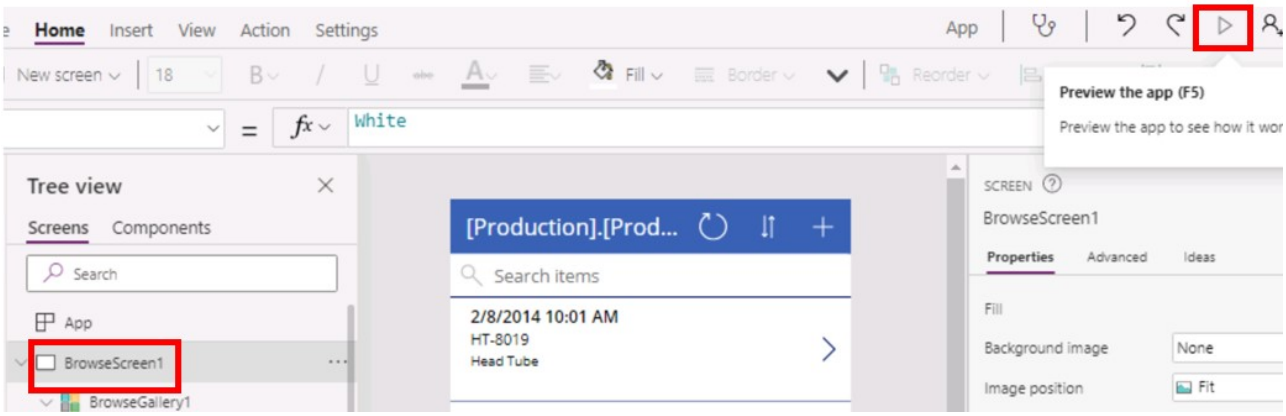
```
If(EditForm1.Mode = FormMode.Edit,
    SubmitForm(EditForm1),
    Patch(
        '[Production].[Product]',
        Defaults('[Production].[Product]'),
        {
            FinishedGoodsFlag: DataCardValue10.Value,
            Name: DataCardValue13.Text,
            ProductNumber: DataCardValue14.Text,
            ReorderPoint: Value(DataCardValue15.Text),
            MakeFlag: DataCardValue12.Value,
            DaysToManufacture: Value(DataCardValue9.Text),
            ListPrice: Value(DataCardValue11.Text),
            ModifiedDate: DateValue1.SelectedDate,
            SellStartDate: DateValue1.SelectedDate,
            SafetyStockLevel: 1000,
            StandardCost: Value(DataCardValue11.Text),
            rowguid: GUID()
        }
    )
);
Navigate(BrowseScreen1, ScreenTransition.CoverRight)
```


OnSelect
=
Tree view
Screens
Components
Search
App
BrowseScreen1
DetailScreen1
EditScreen1
EditForm1
LblAppName3
IconAccept1
IconCancel1
RectQuickActionBar3
Untitled - Notepad
File Edit Format View Help
If(EditForm1.Mode = FormMode.Edit,
SubmitForm(EditForm1),
Patch(
'[Production].[Product]',
Defaults('[Production].[Product]'),
{
FinishedGoodsFlag: DataCardValue10.Value,
Name: DataCardValue13.Text,
ProductNumber: DataCardValue14.Text,
ReorderPoint: Value(DataCardValue15.Text),
MakeFlag: DataCardValue12.Value,
DaysToManufacture: Value(DataCardValue9.Text),
ListPrice: Value(DataCardValue11.Text),
ModifiedDate: DateValue1.SelectedDate,
SellStartDate: DateValue1.SelectedDate,
SafetyStockLevel: 1000,
StandardCost: Value(DataCardValue11.Text),
rowguid: GUID()
}
)
);
Navigate(BrowseScreen1, ScreenTransition.CoverRight)

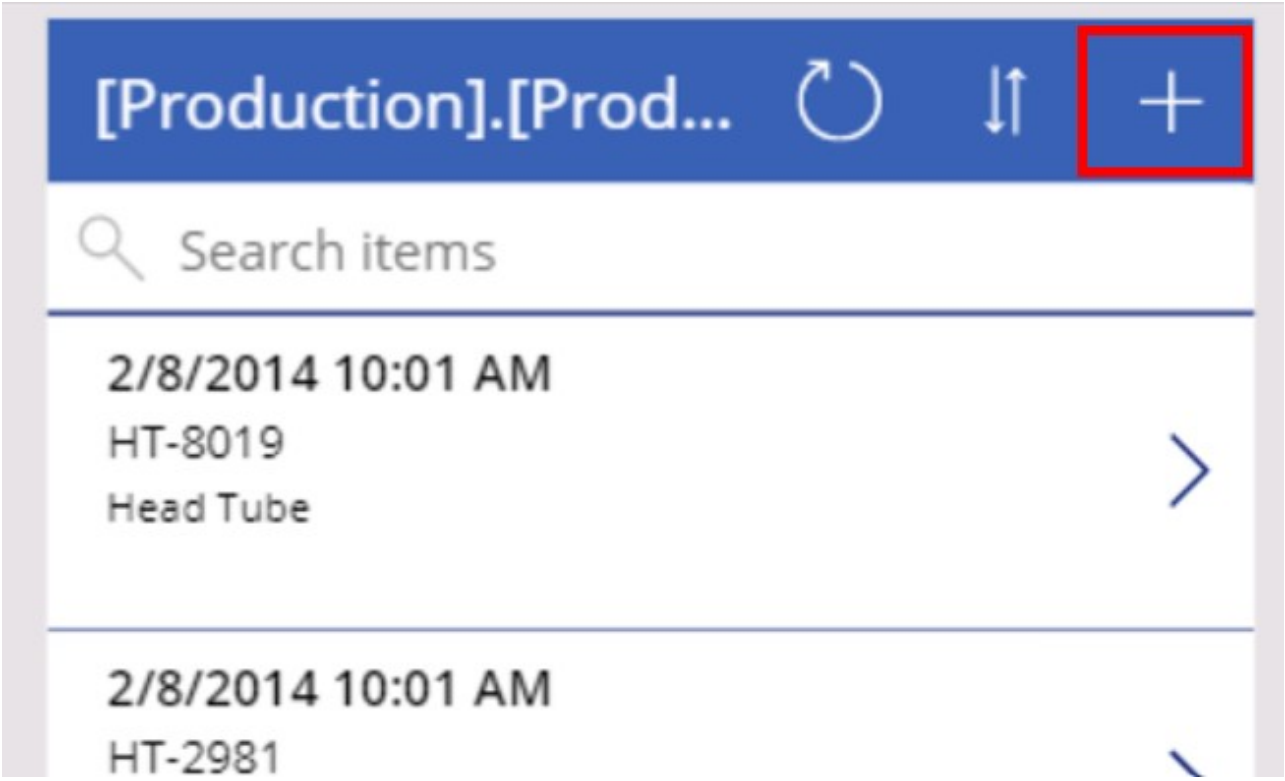

Back
Insert
Icon
Color
Background color
OnSelect
=
SubmitForm(EditForm1)
SubmitForm(EditForm1) = This formula has side effects and cannot be evaluated.
Tree view
Screens
Components
Search
* DaysToManufacture



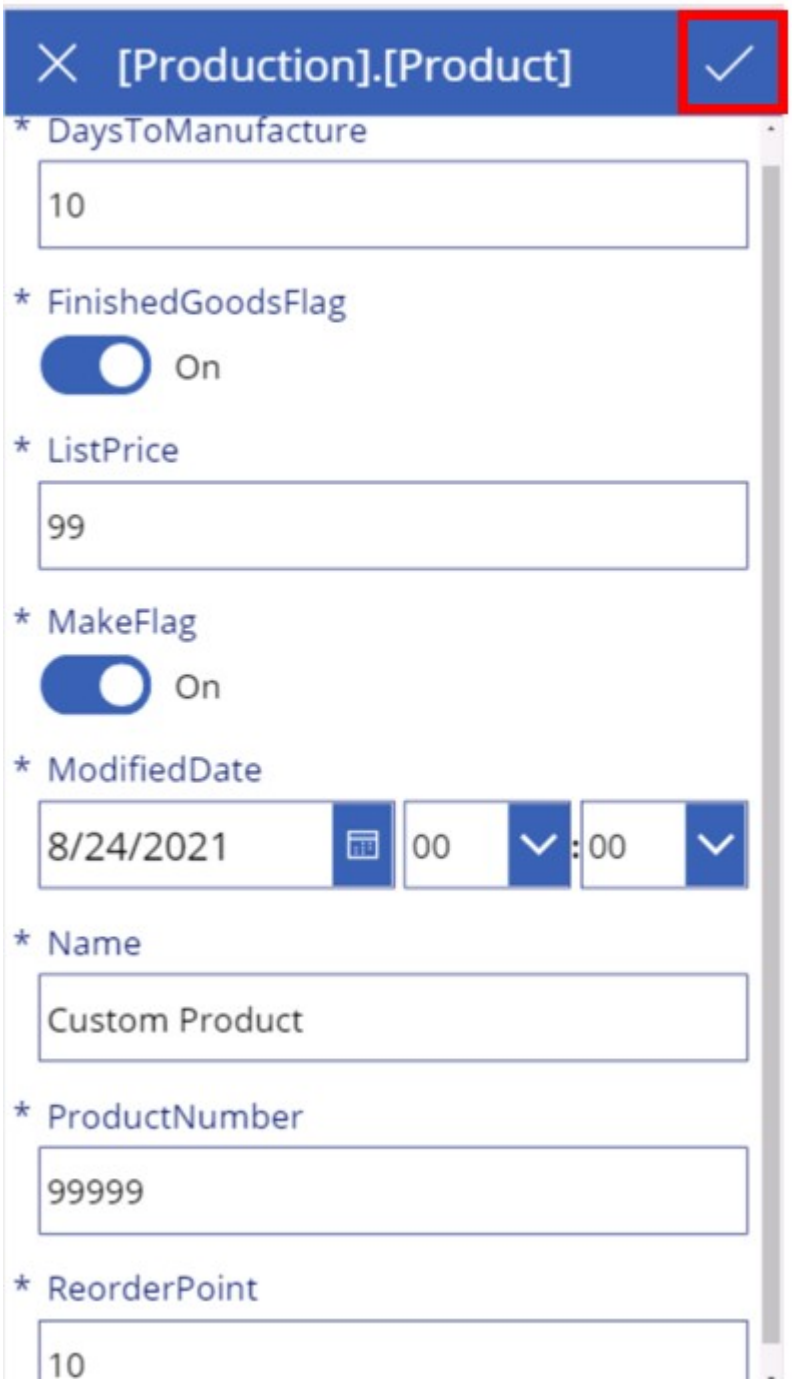
5. Now you can test to add new product, so select **BrowseScreen1** and press play button or F5 to test the app



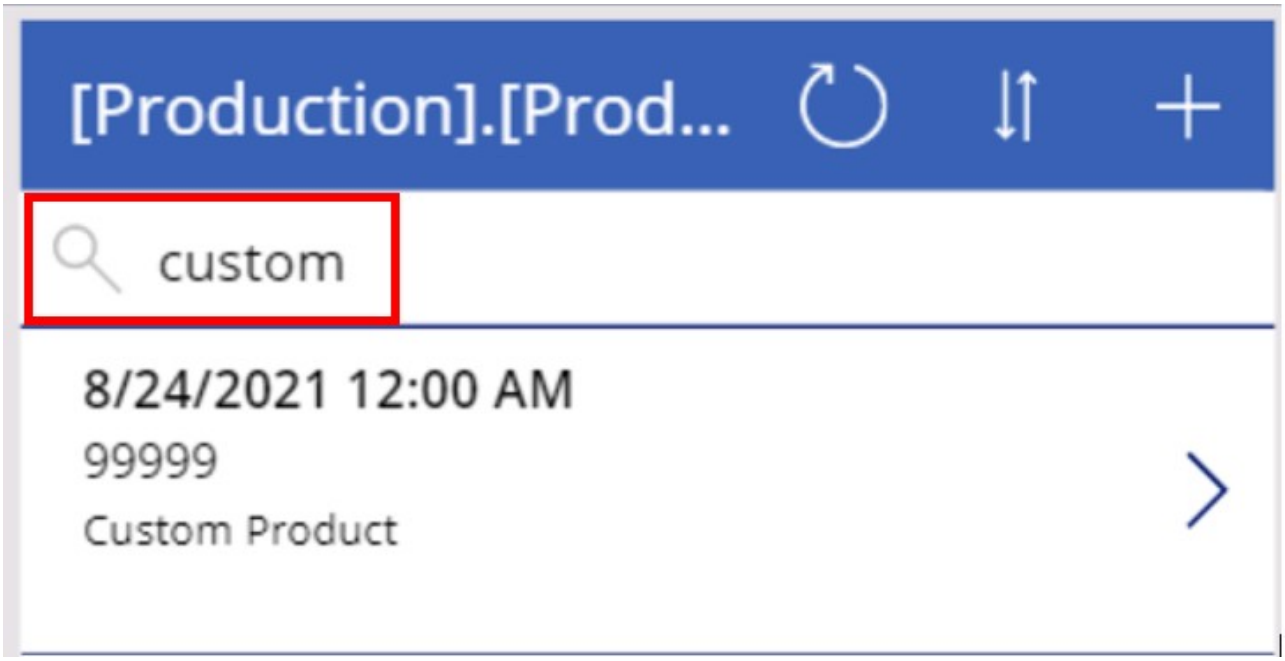
6. Click + icon on top right of the **BrowseScreen**



7. Fill-in all the fields and, submit the form



8. Use search functionality in **BrowseScreen** to search your product you just added



> [!KNOWLEDGE] For more information about On-Premise Data Gateway, review the Microsoft document **here**.

Congratulations

You have now completed the lab. Check if any of your colleagues need help with completing the lab.