Module 1 – Data Modeling with Dataverse

Goal

To design and implement the data model for Inspections while adhering to best practices.

You will learn how to:

- Work with Dataverse tables and configure table settings (metadata)
- Configure columns with various data types
- Create relationships
- Configure relationship behavior
- Add alternate keys

Table Of Contents

Tables 1

Columns 18

Relationships 23

Column Mappings 26

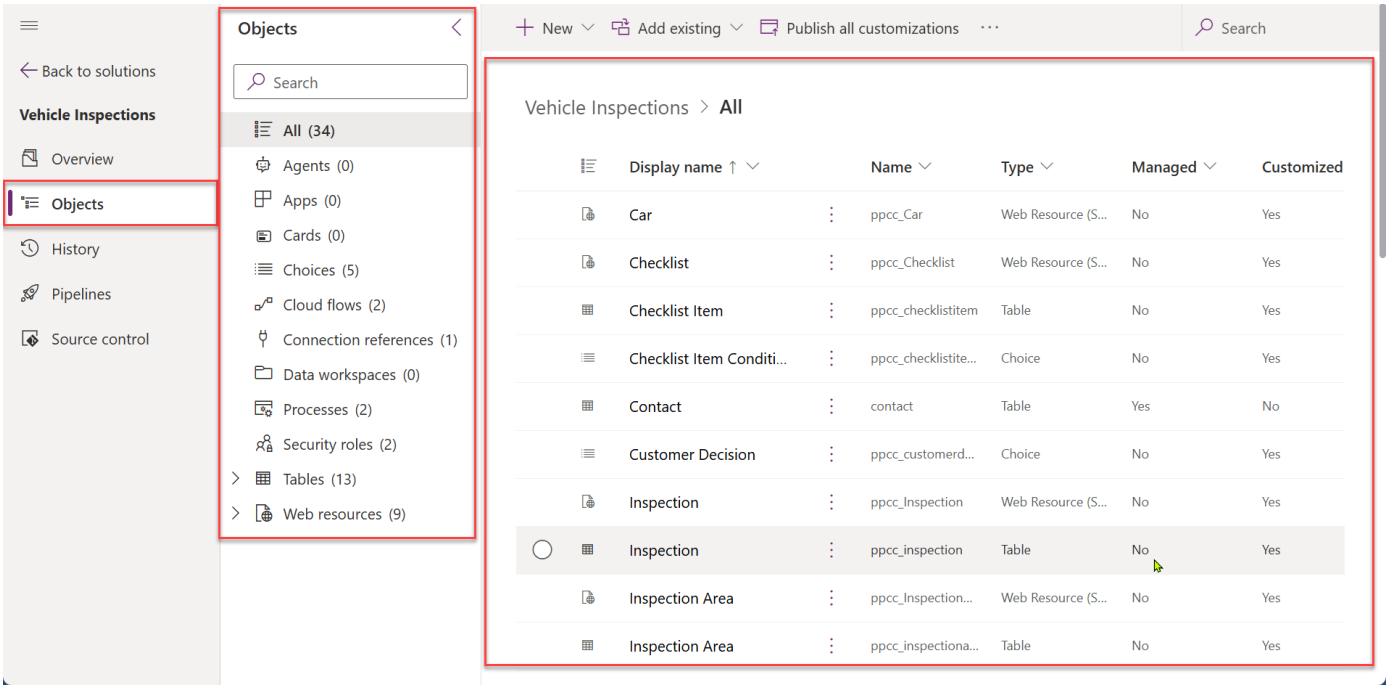
(Alternate) Keys 31

Tables

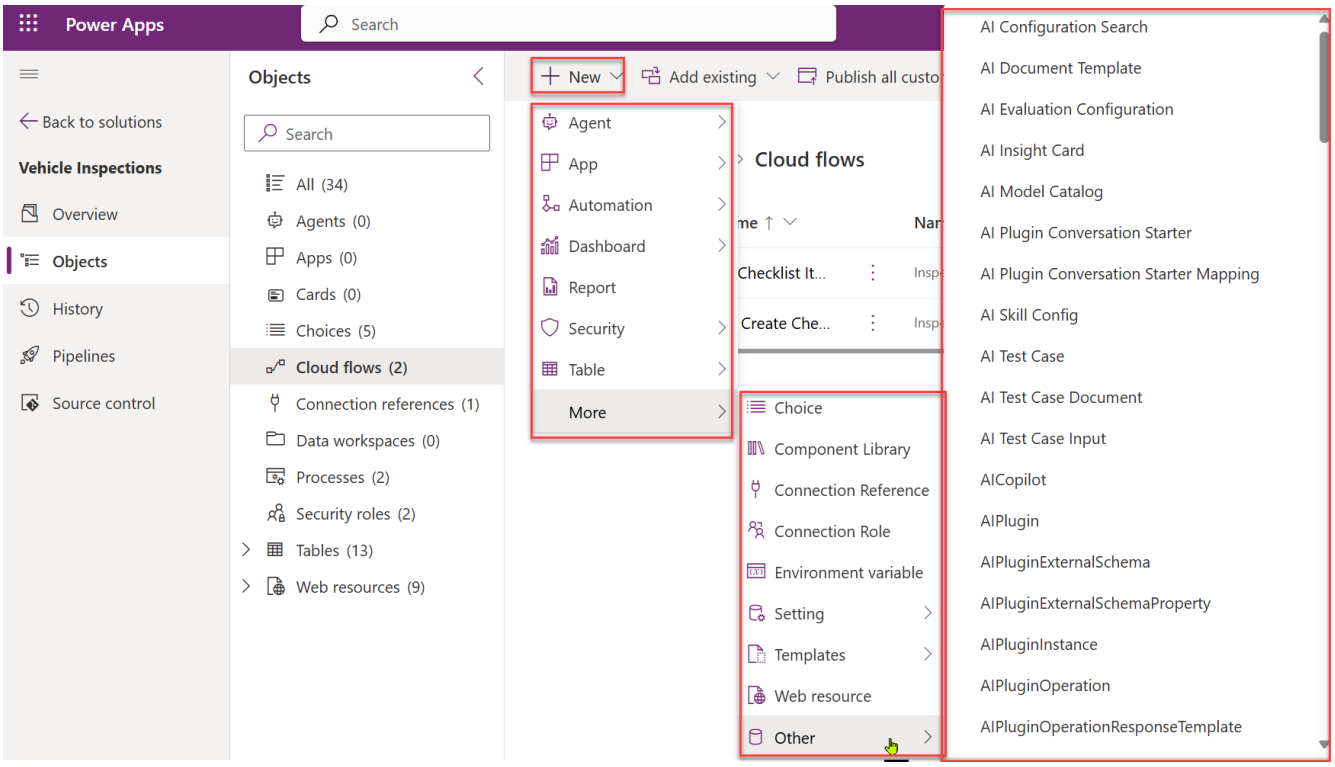
We will begin by exploring tables and learn how to navigate across the various components.

1. Browse to https://make.powerapps.com
2. Select the environment assigned to you
3. Navigate to **Solutions** and open the **Vehicle Inspections** solution

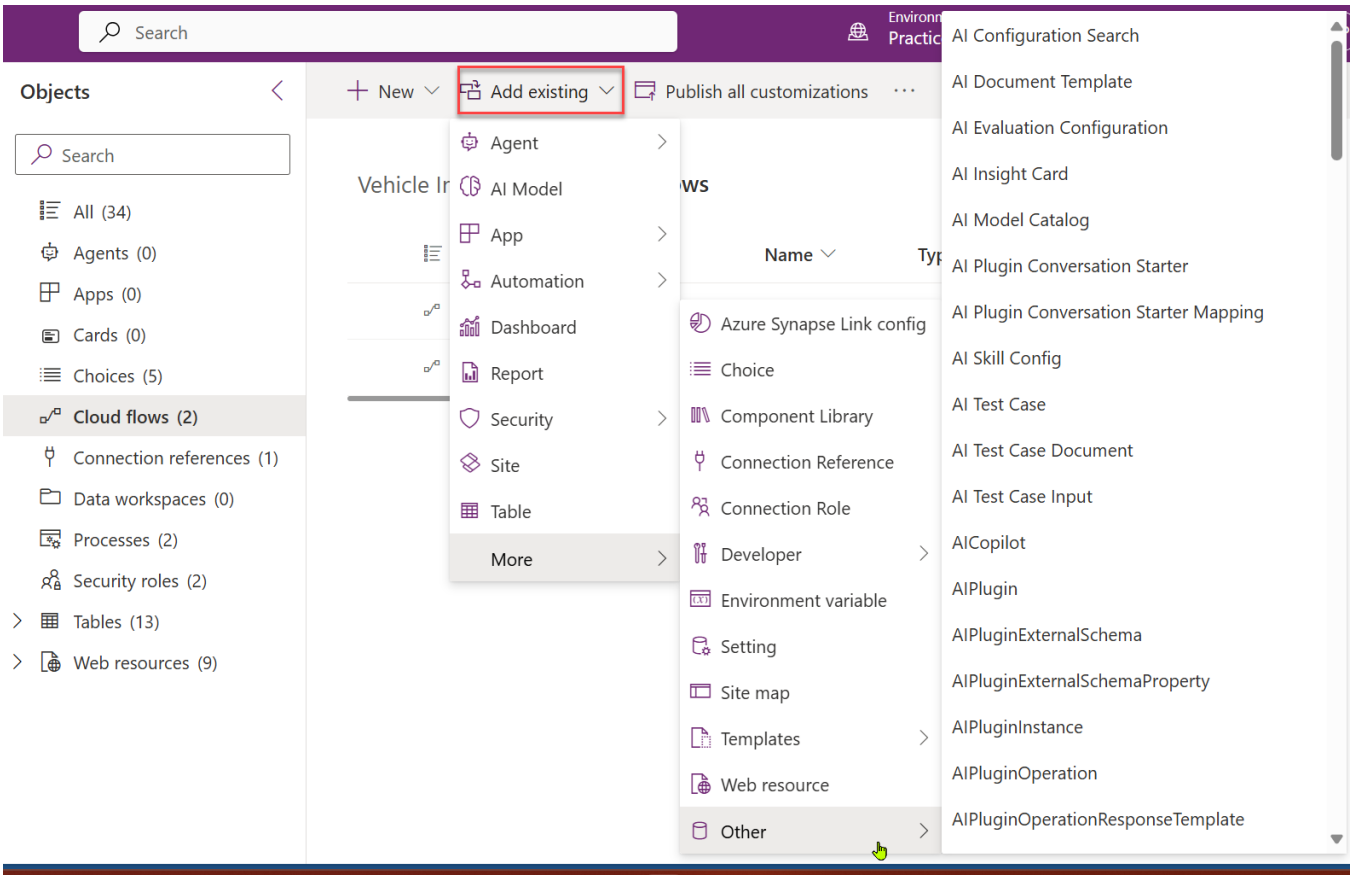
4. By default you'll land on the **Objects** screen that lists all components added to the solution



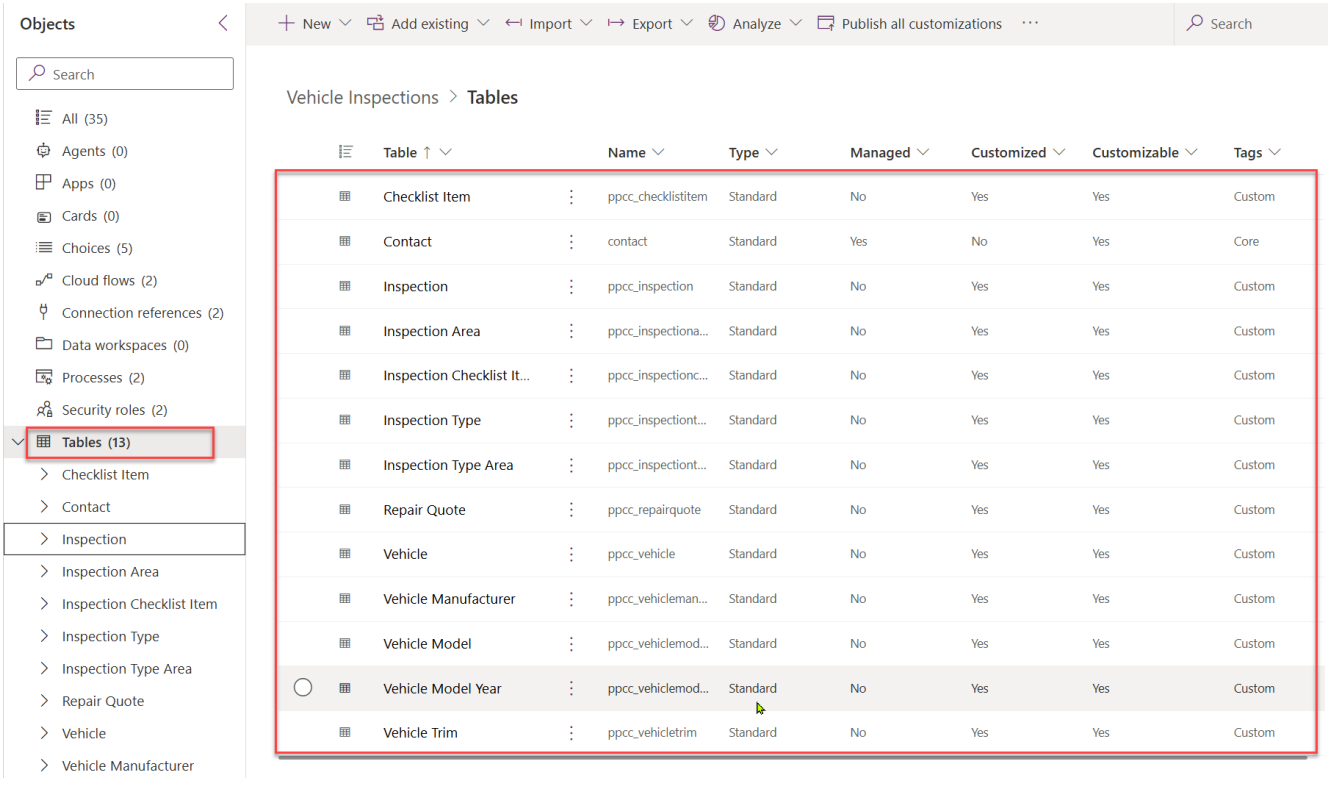
5. You can add a new component from anywhere in a solution by selecting the **+New** button in the top command bar. As you can see there are dozens of solution component types. The most commonly used ones are listed in the first level dropdown, then second level, and the remaining scrollable list under *Other*.



6. You can also add existing components that are not already part of the solution by selecting **Add existing**. You'll see the same list of component types.

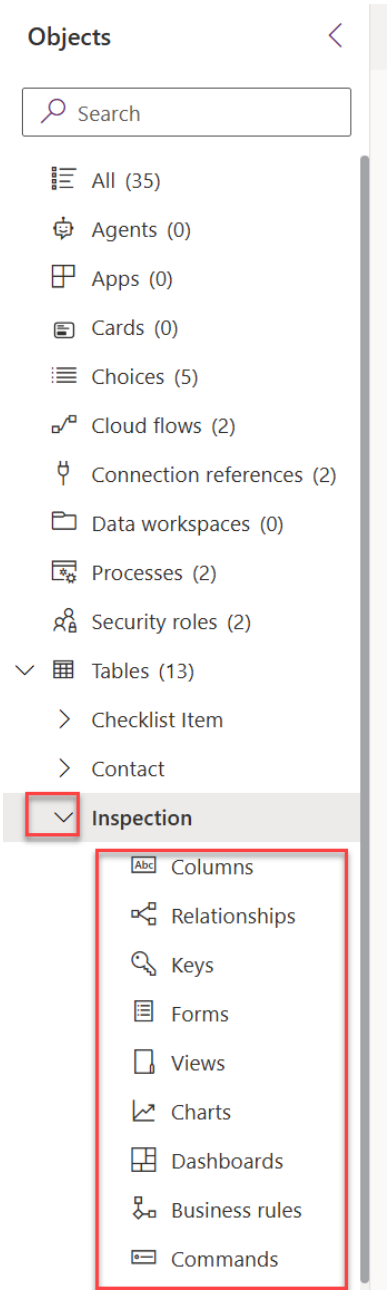


7. Under the **Objects** tree, select **Tables**.
8. Here you will see all tables currently added to the solution.

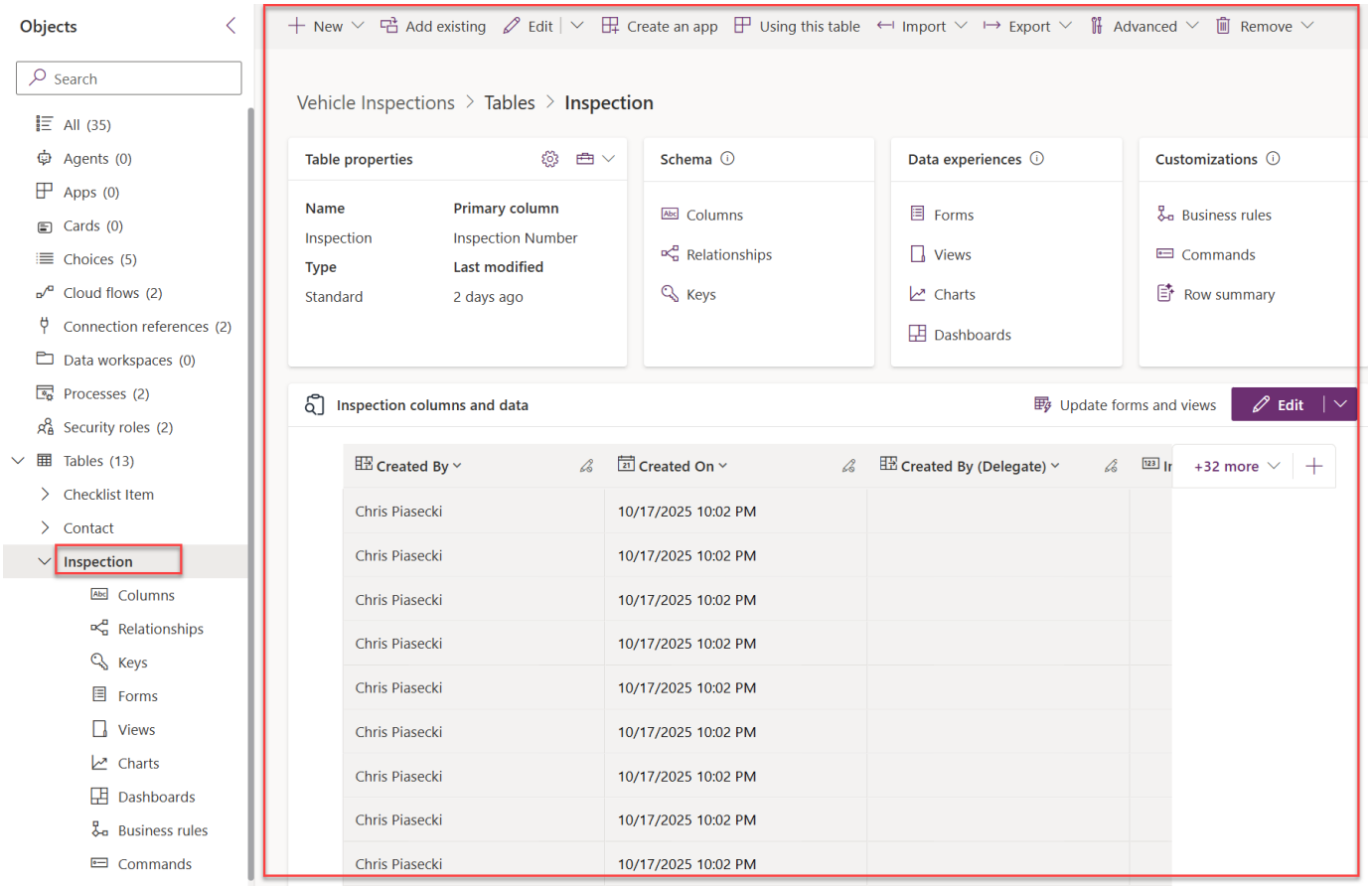


9. Expand **Tables** under the solution object tree.

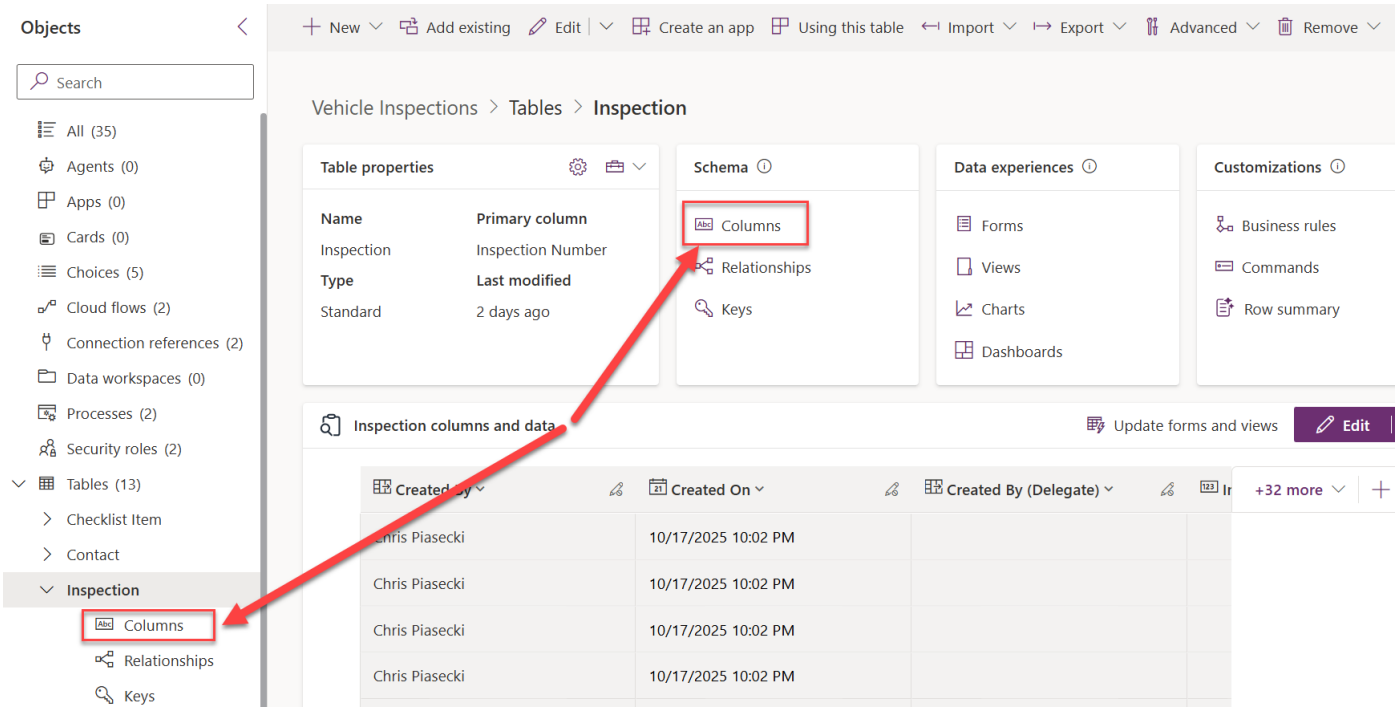
10. Expand the **Inspection** table. You will see all the sub-component types for tables.

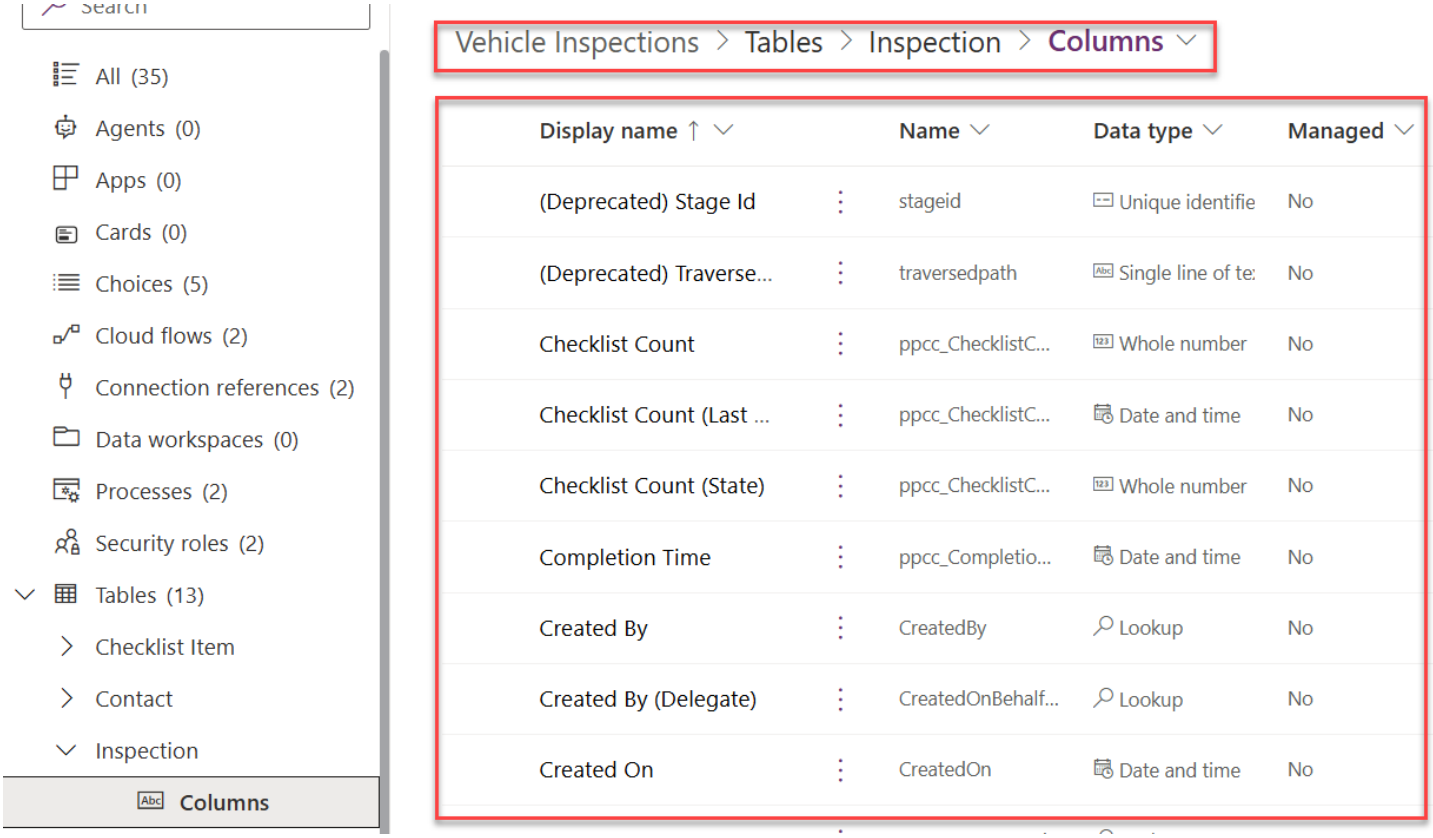


11. Select the Inspection table to see a detailed overview of the table



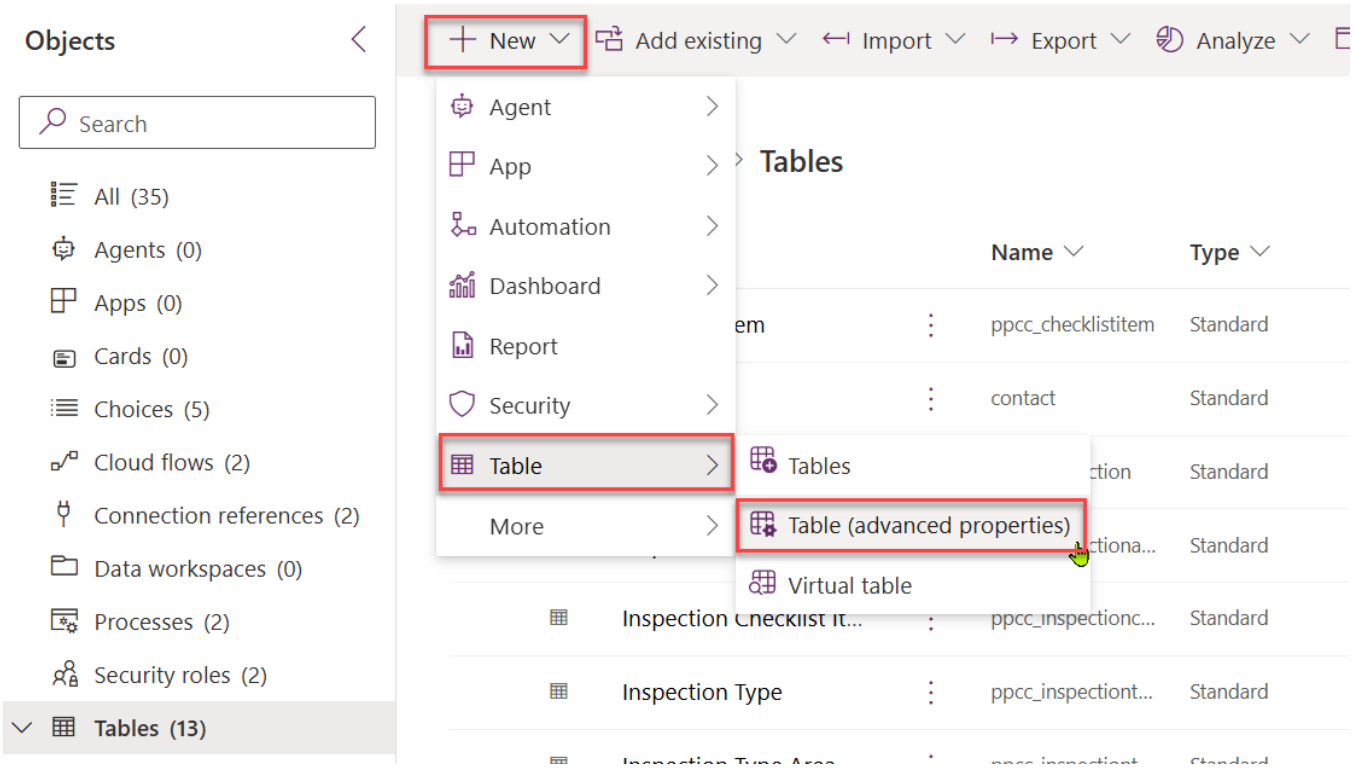
12. You can select a table sub-component type from either the object tree or the table screen to view its list.



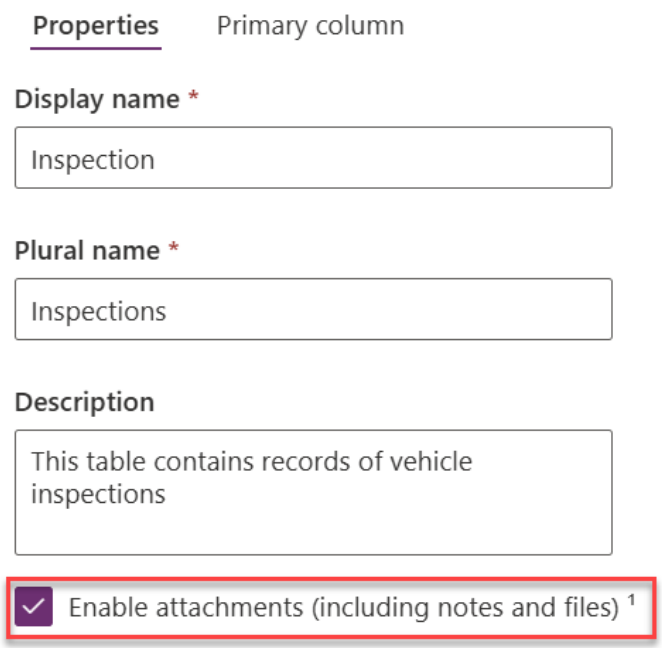


We will now explore the new Table creation process.

13. Select the **New button** > **Table** > **Table (advanced properties)** to create the Inspection table

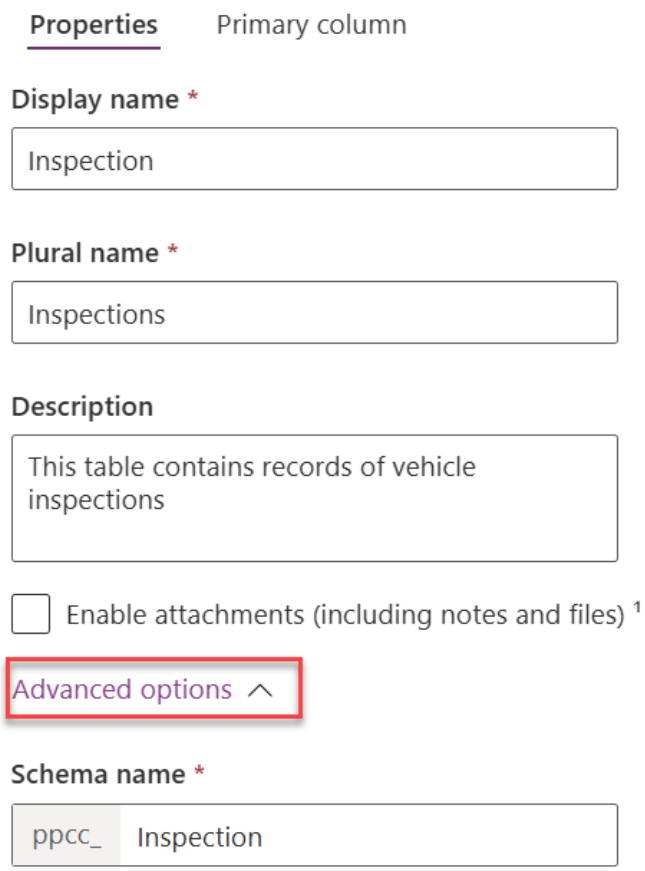


14. Enter a **Display Name** of *Inspection*. Display names should always be <u>singular</u>
15. The plural name is automatically populated but can be updated. Leave it as **Inspections**
16. Copy/paste the following into the description **description**:
    This table contains records of vehicle inspections
17. Check **Enable attachments (including notes and files)**.

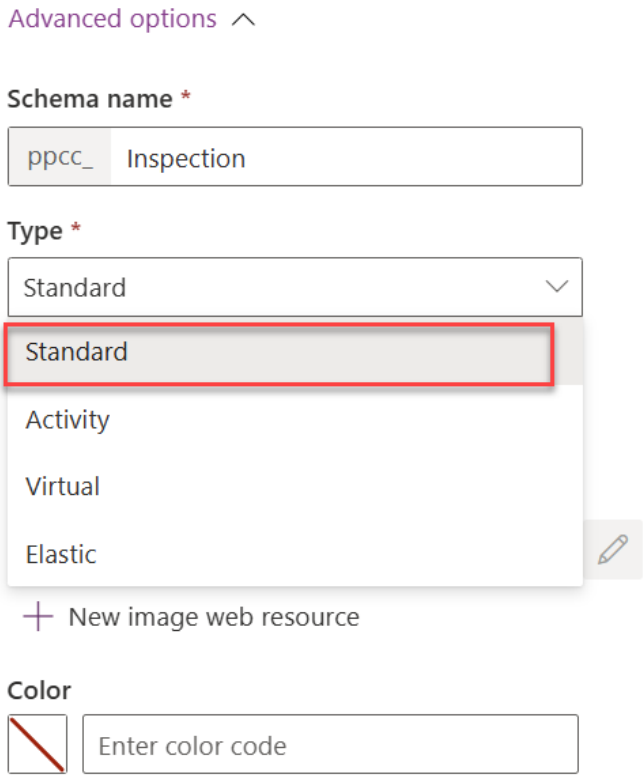


18. Select the **Primary Column** tab and set the display name to *Inspection Number* and optionally enter a description Change the maximum character count to *20*.

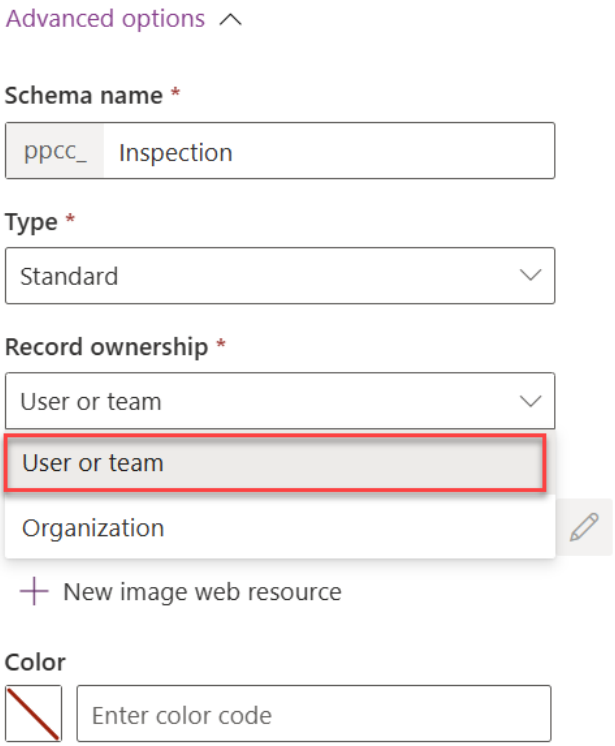
19. Switch back to the **Properties** tab and expand **Advanced options**.



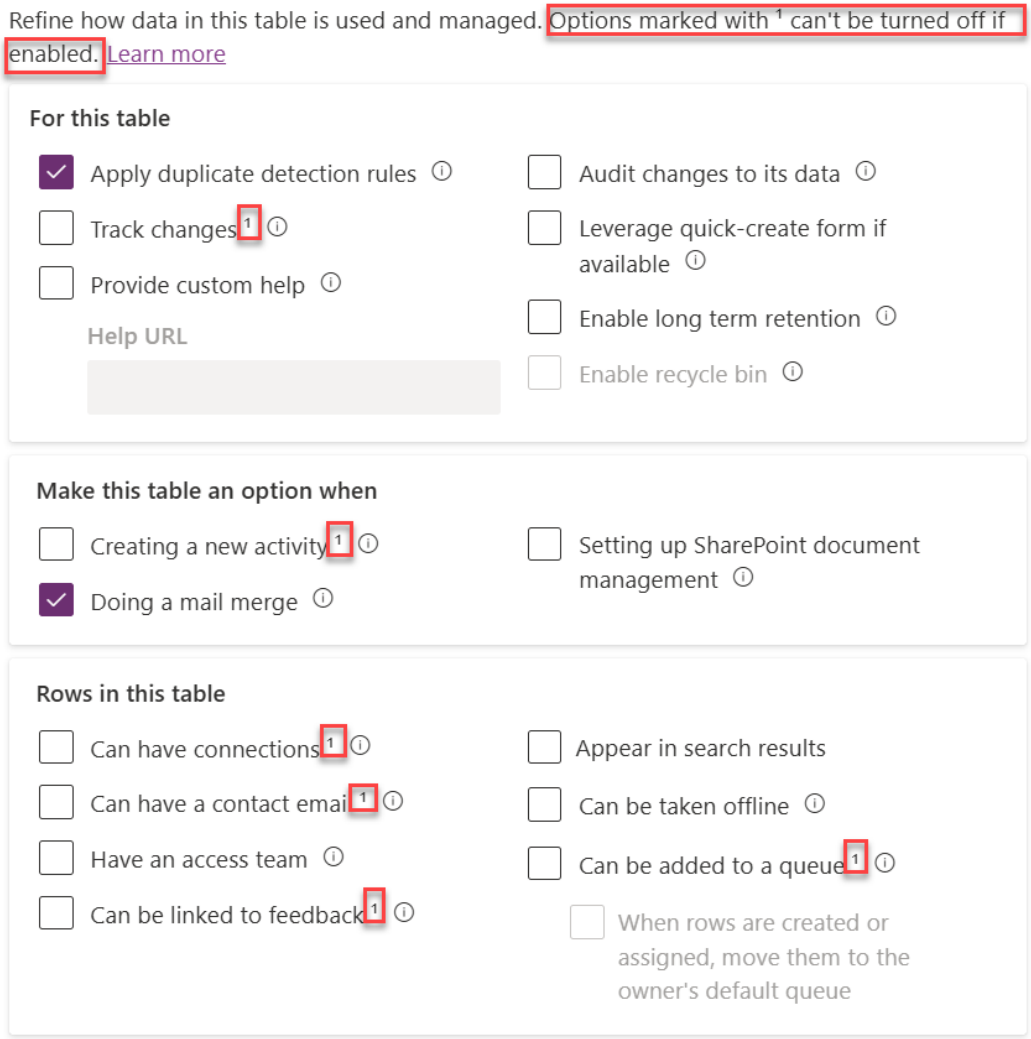
20. Leave **Type** as the default **Standard** for the Inspection table.



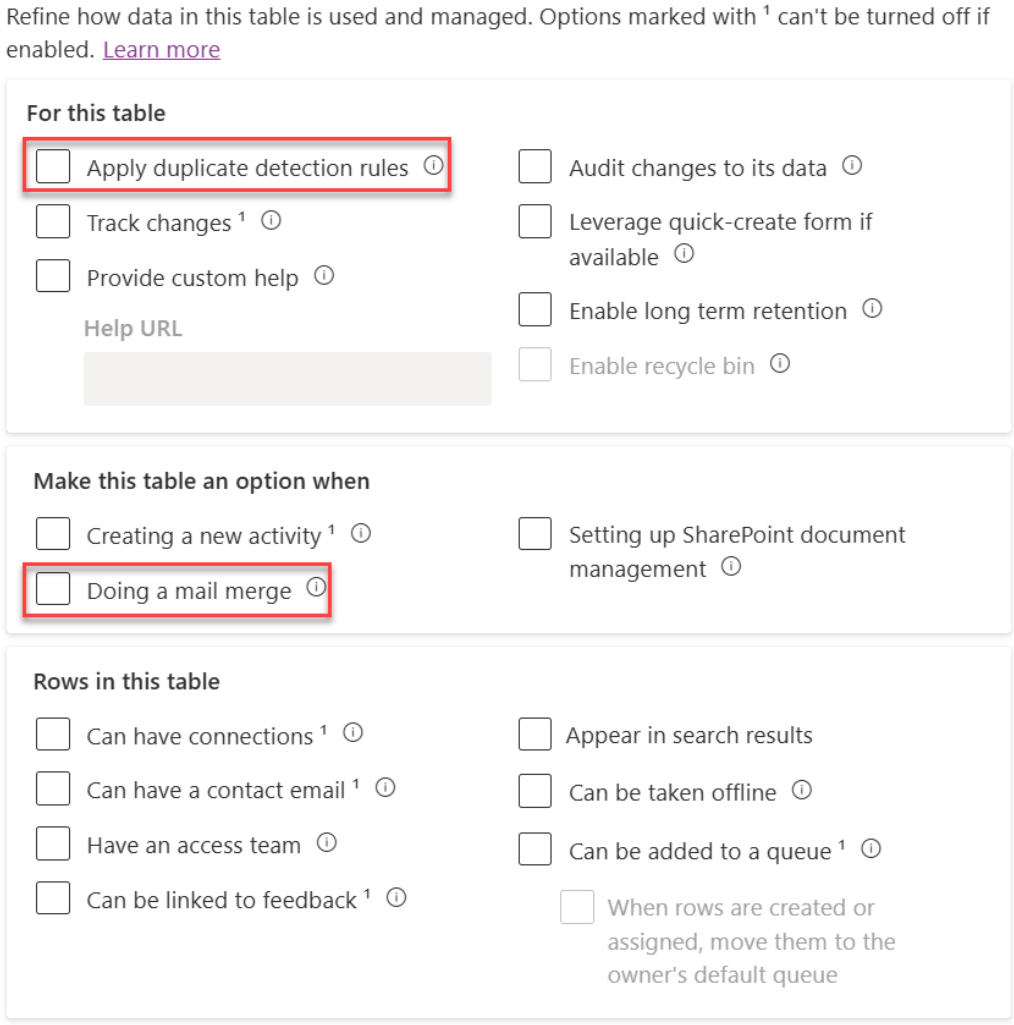
21. The **Record Ownership** cannot be changed after creation. Always default to using **(User or Team)**.



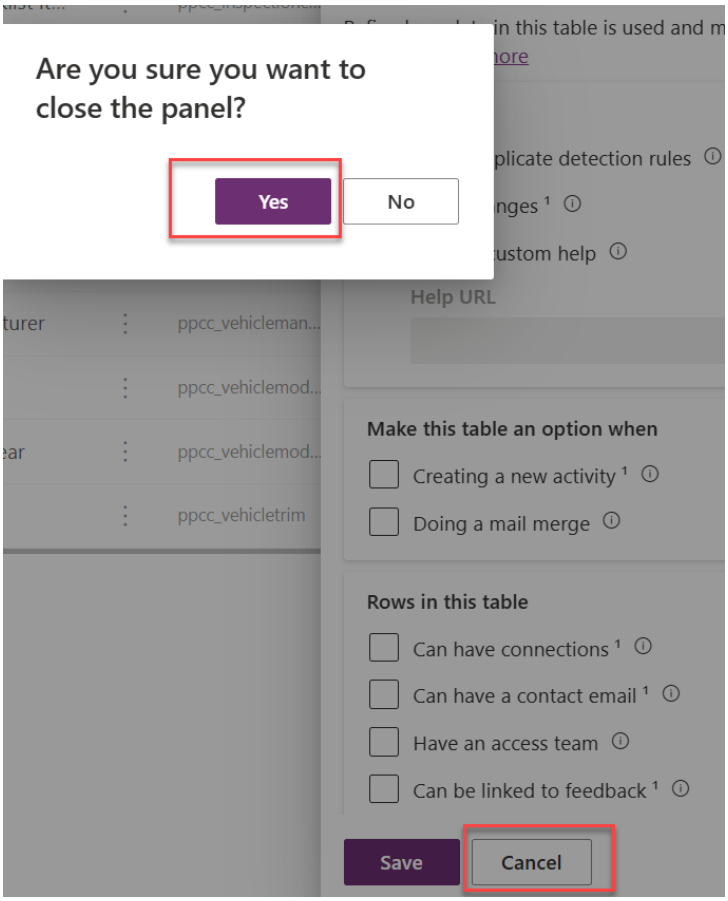
22. There are various **options** that can be enabled for a table. Some options can't be turned off once enabled, including the **Enable attachments (including notes and files)** option previously selected.



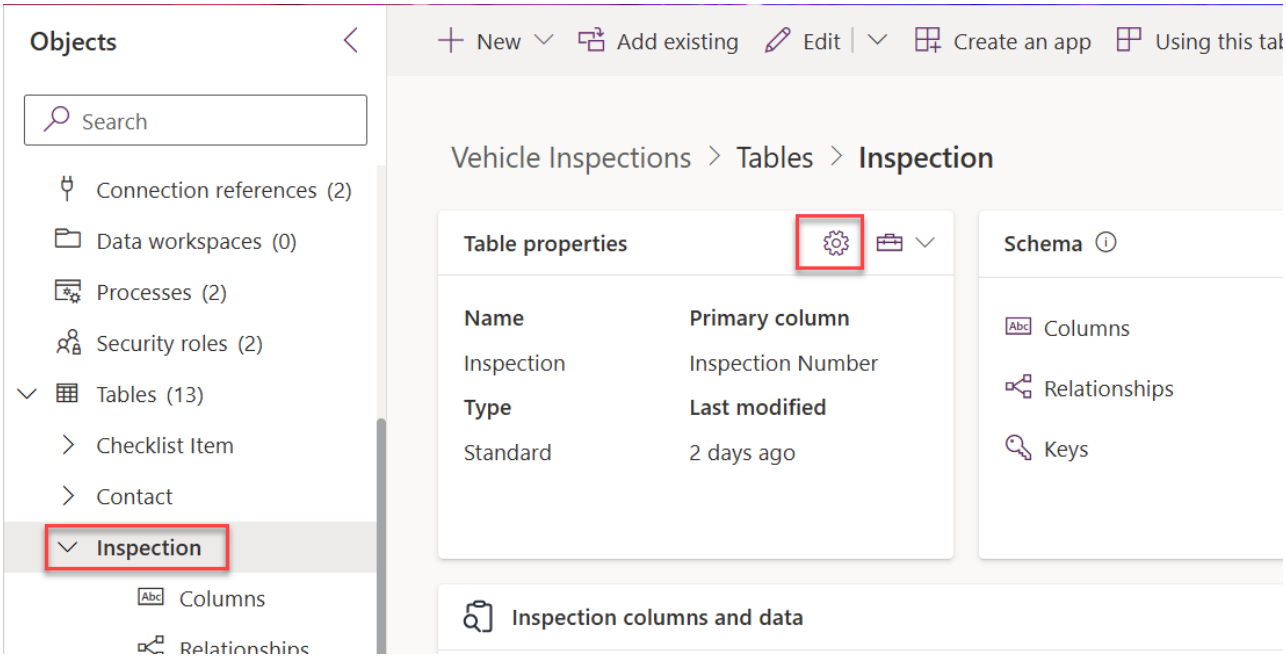
23. Uncheck default options such as **Apply duplicate detection rules** and **Doing a mail merge** unless you know from the start you need it (we will not need it for Inspection).



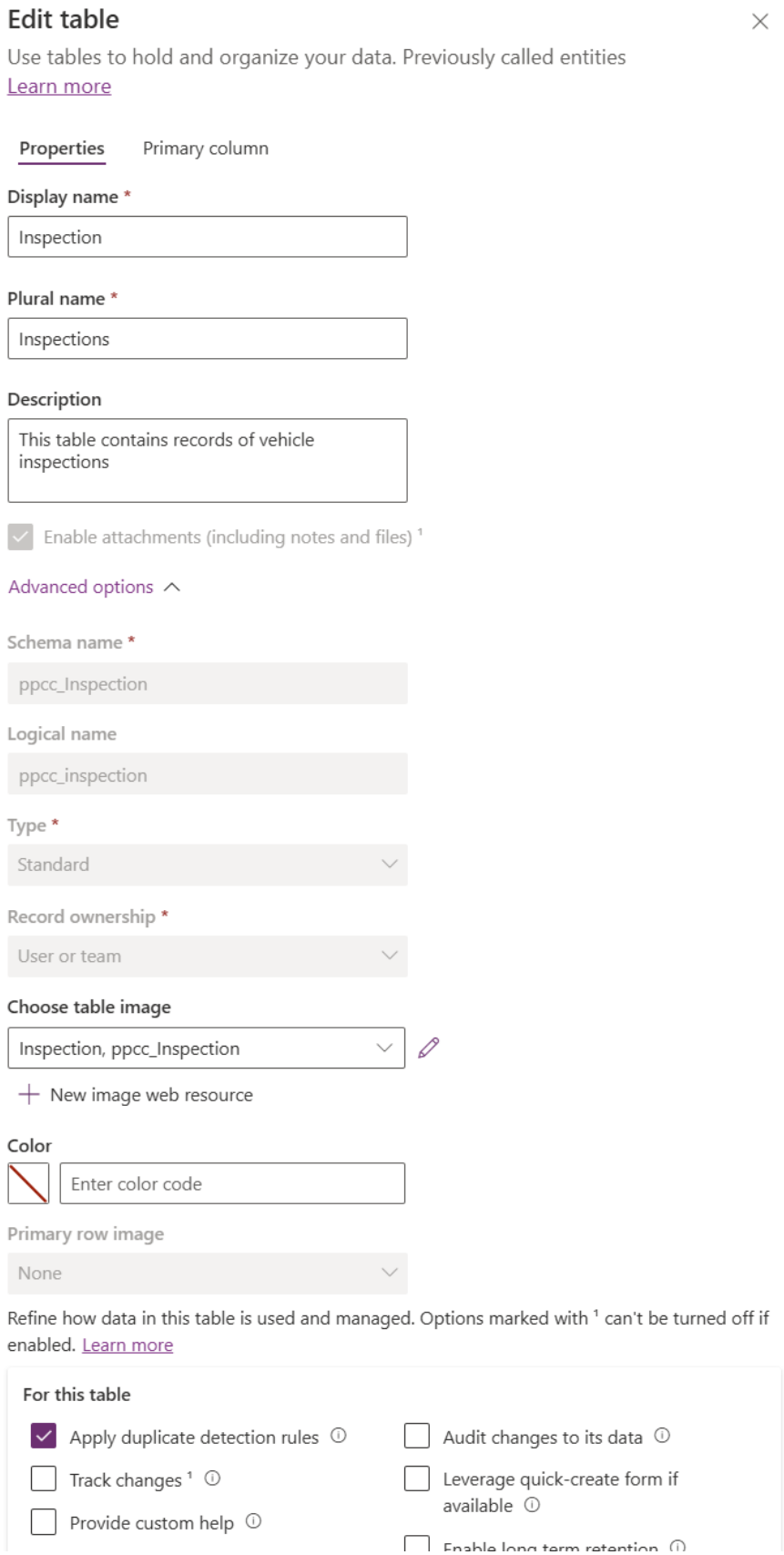
24. Since Inspection is already created, we will stop here. Click **Cancel** , then Yes when prompted with Are you sure you want to close the panel?.



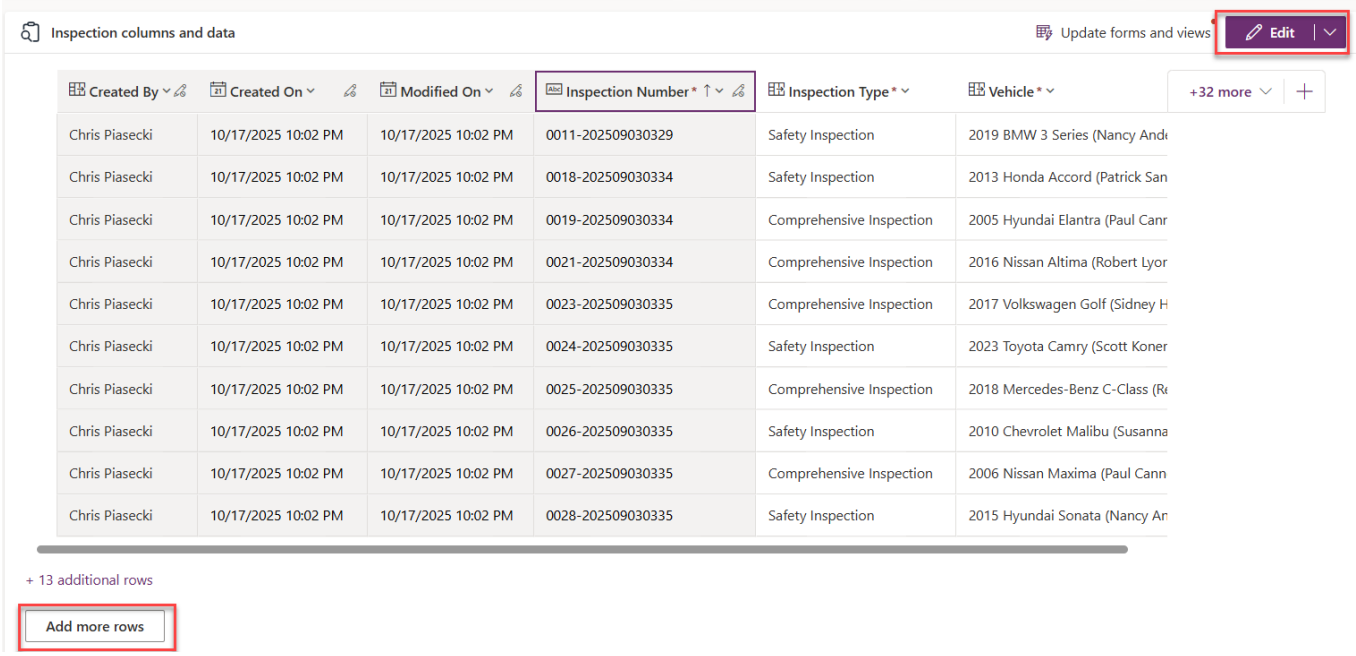
25. You can view/edit table settings for an existing table.

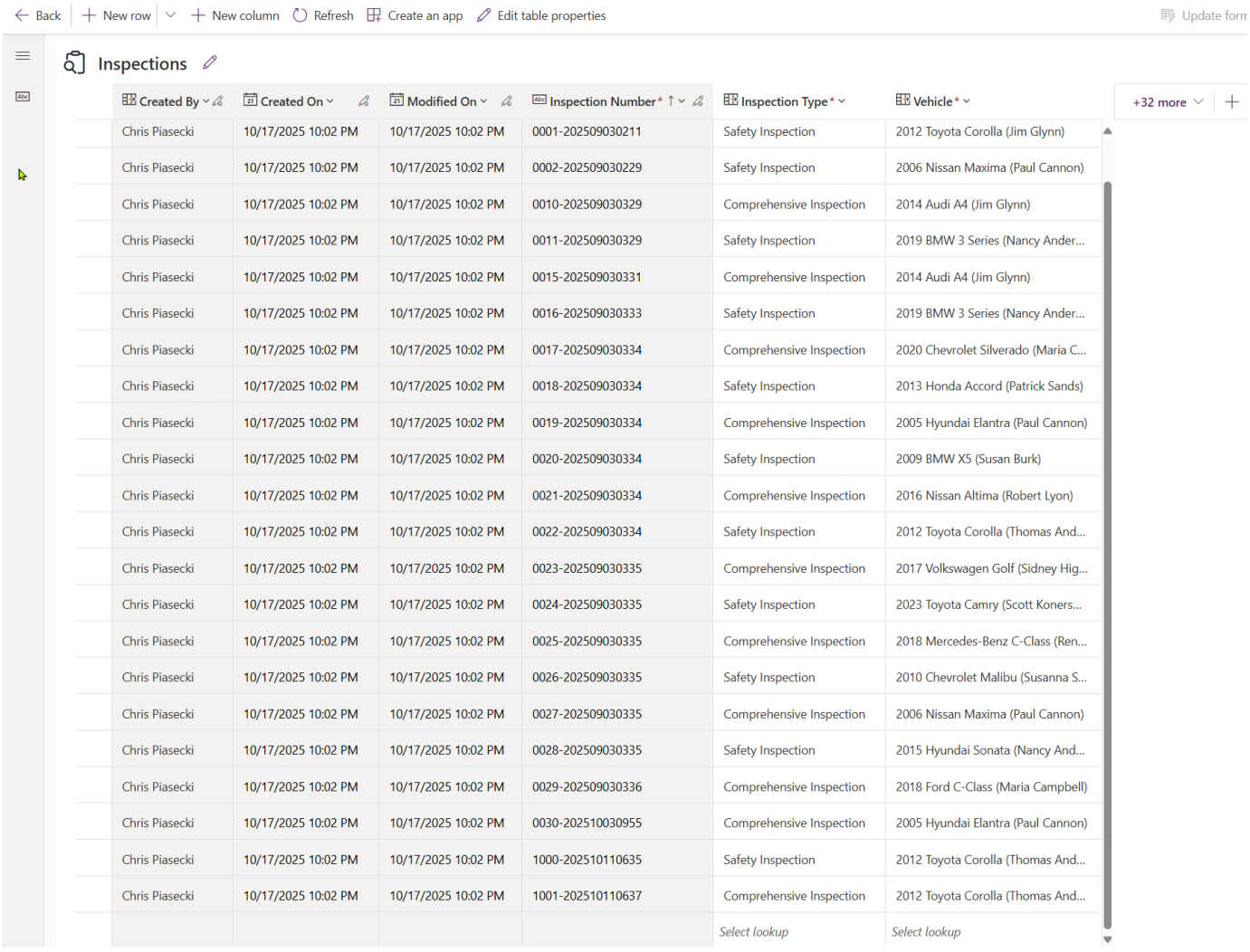


26. Settings such as the schema name, logical name, type, and ownership are not editable after creation.

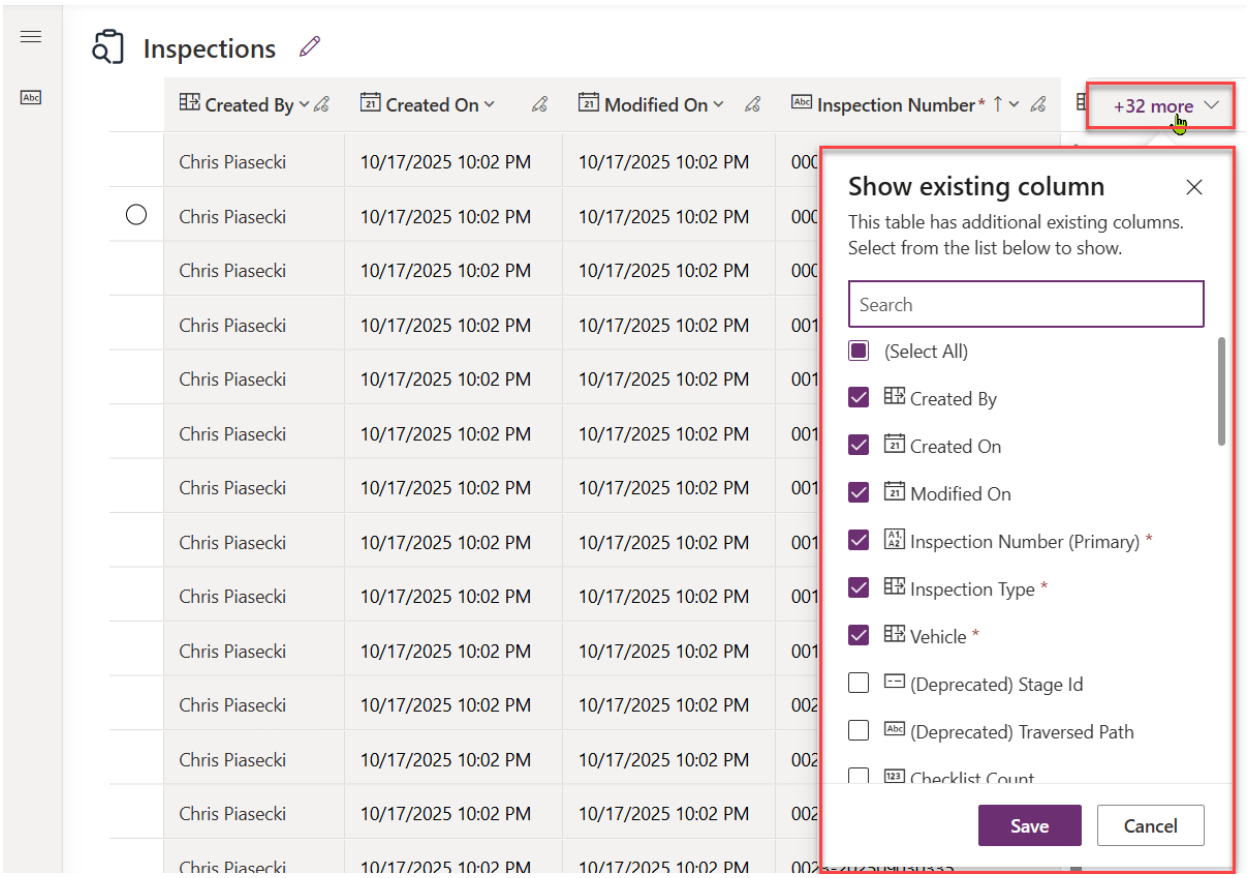


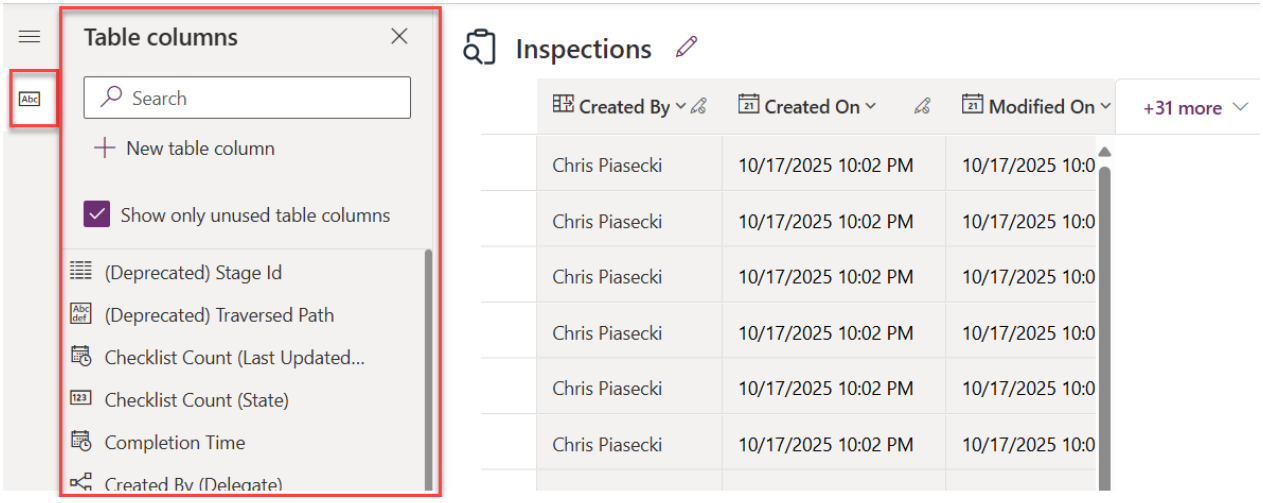
27. You can view/edit up to 10 existing rows from the table screen.
28. Clicking **Add more rows** or **Edit** will open a full screen where you can work with an infinite scrolling list of records.



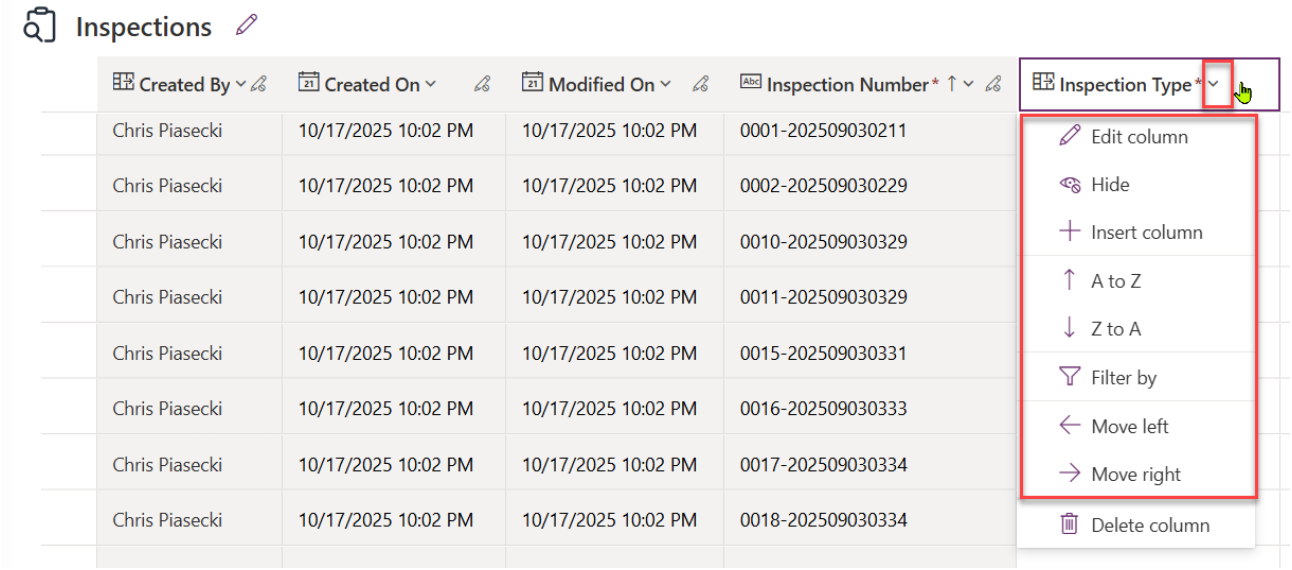


29. Columns can be shown or hidden in a couple different ways.

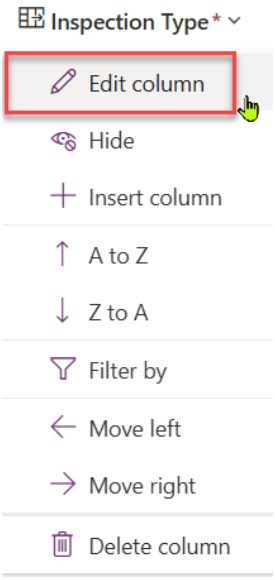


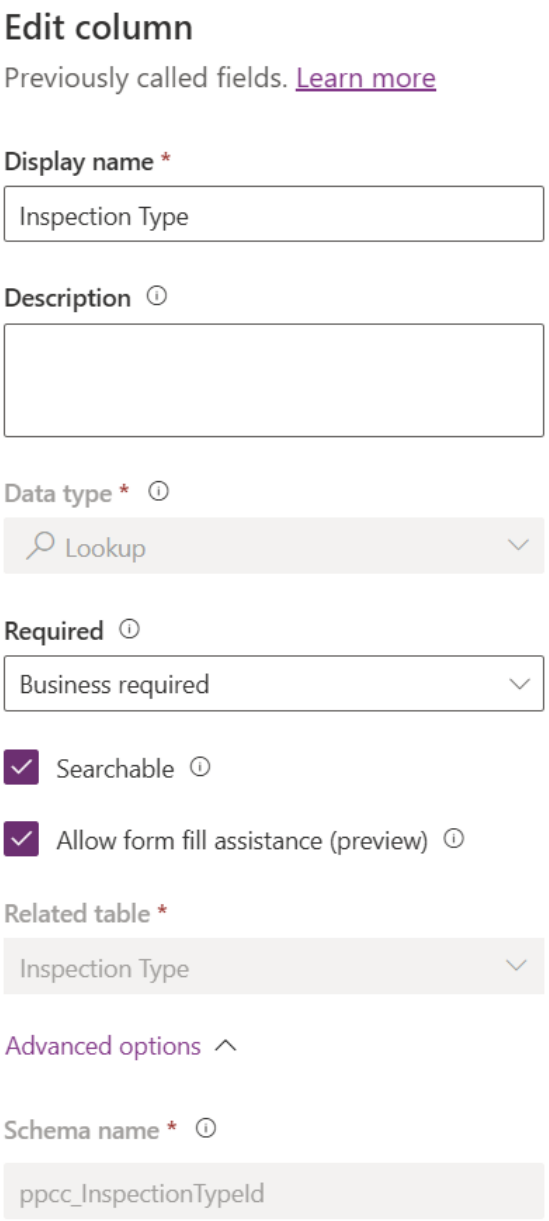


30. You can filter, sort, drag columns to change ordering, or hide the column.

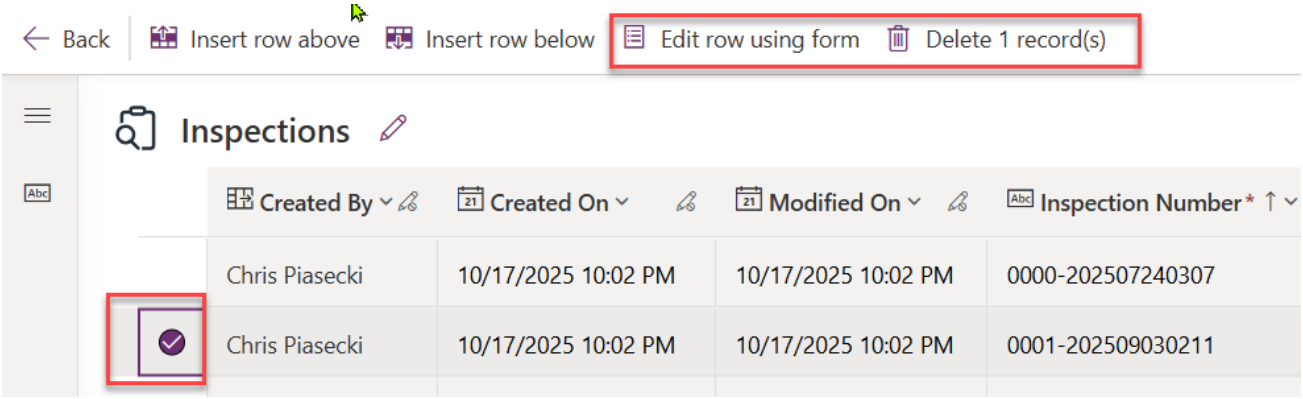


31. The column metadata/options can also be viewed/edited.





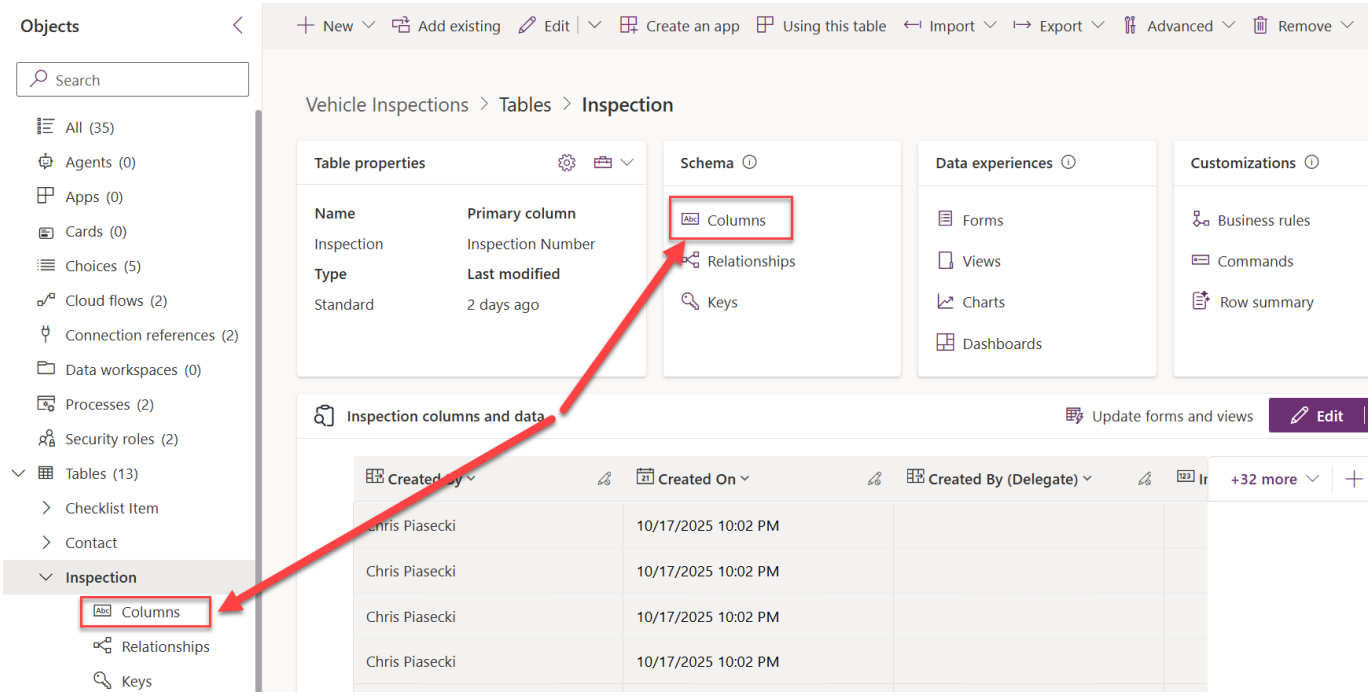
32. Selecting an existing row shows options to edit or delete the row.
    Note: Be careful as to not accidentally Delete as this view does not prompt you to confirm deletions.

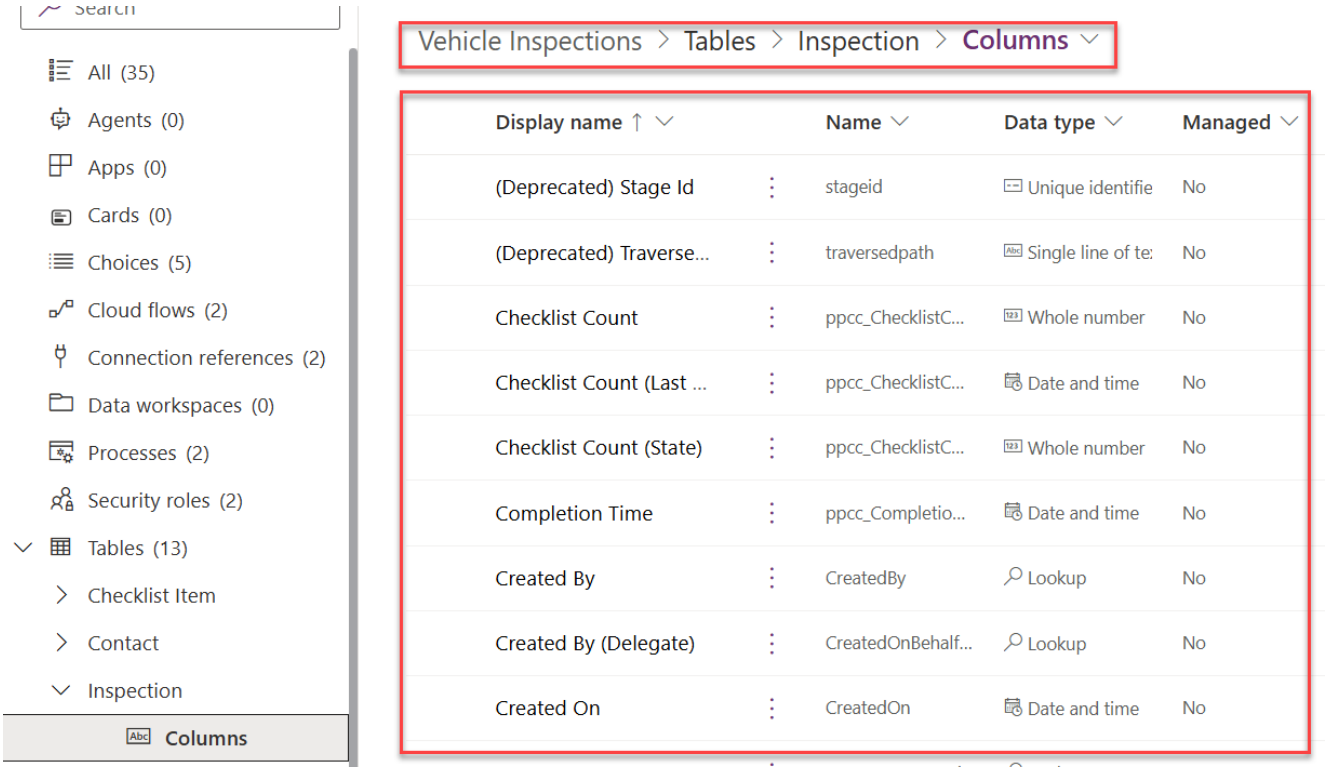


Columns

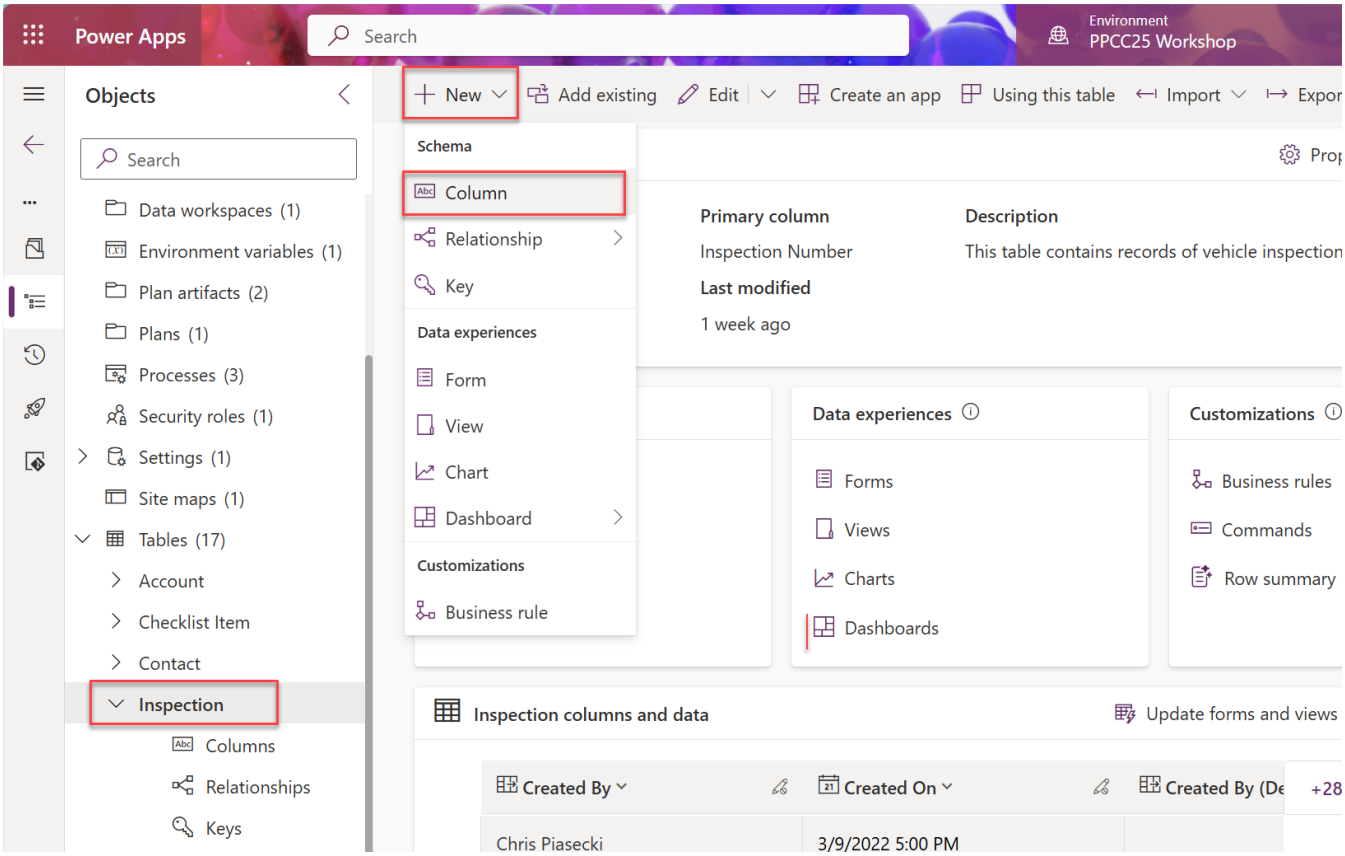
Next, we will explore how columns are created and a few of the most frequently used types.

33. Navigate to the *Inspection* table
34. Click **Columns** to view the existing list.

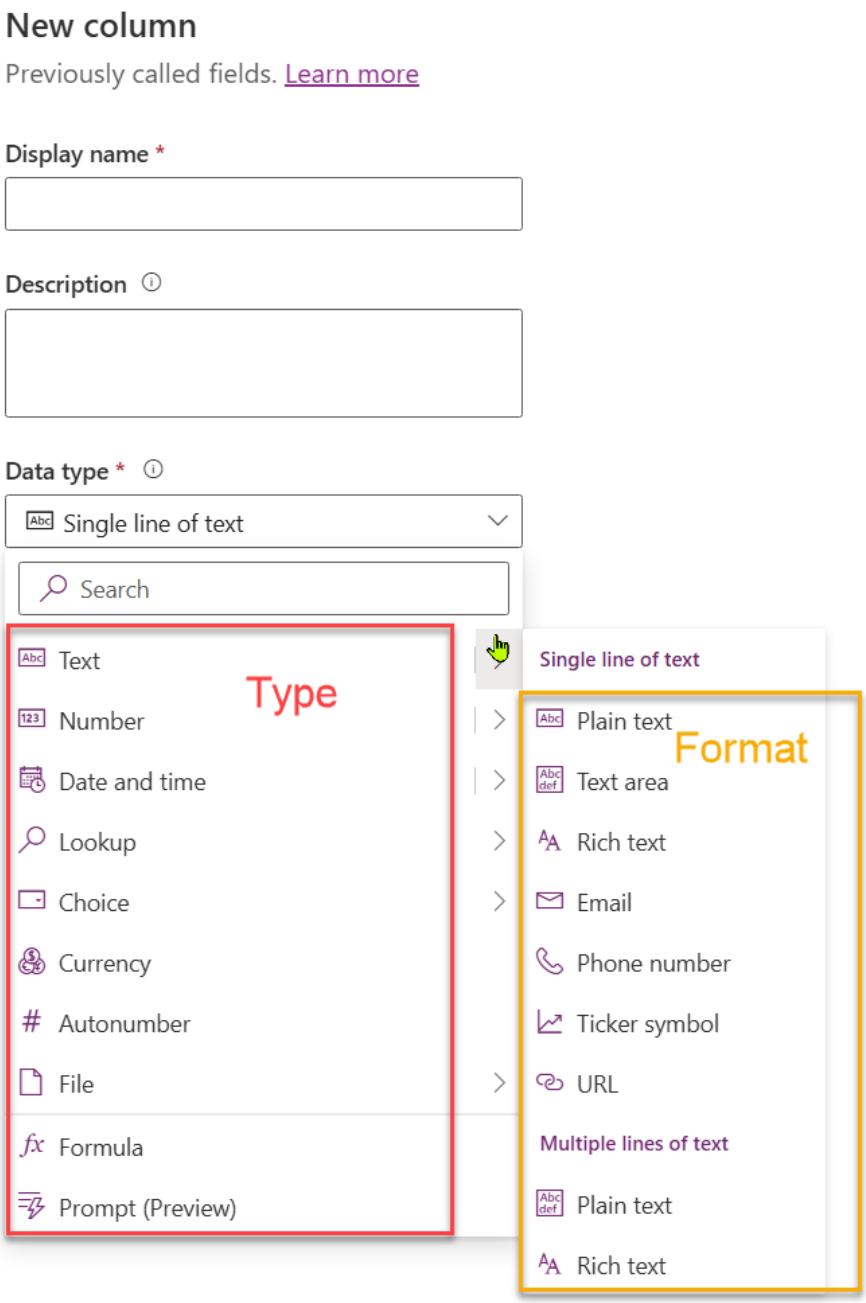




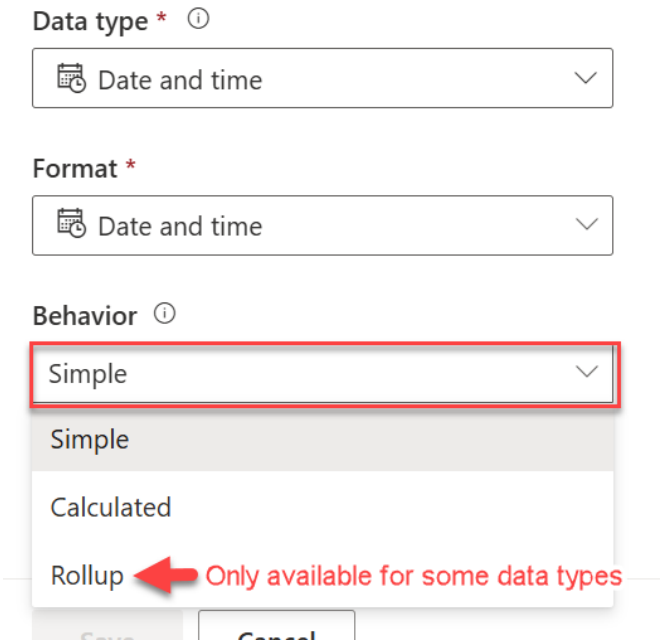
35. To create a new column, select **+New** at the top of the screen,, then **Column**.

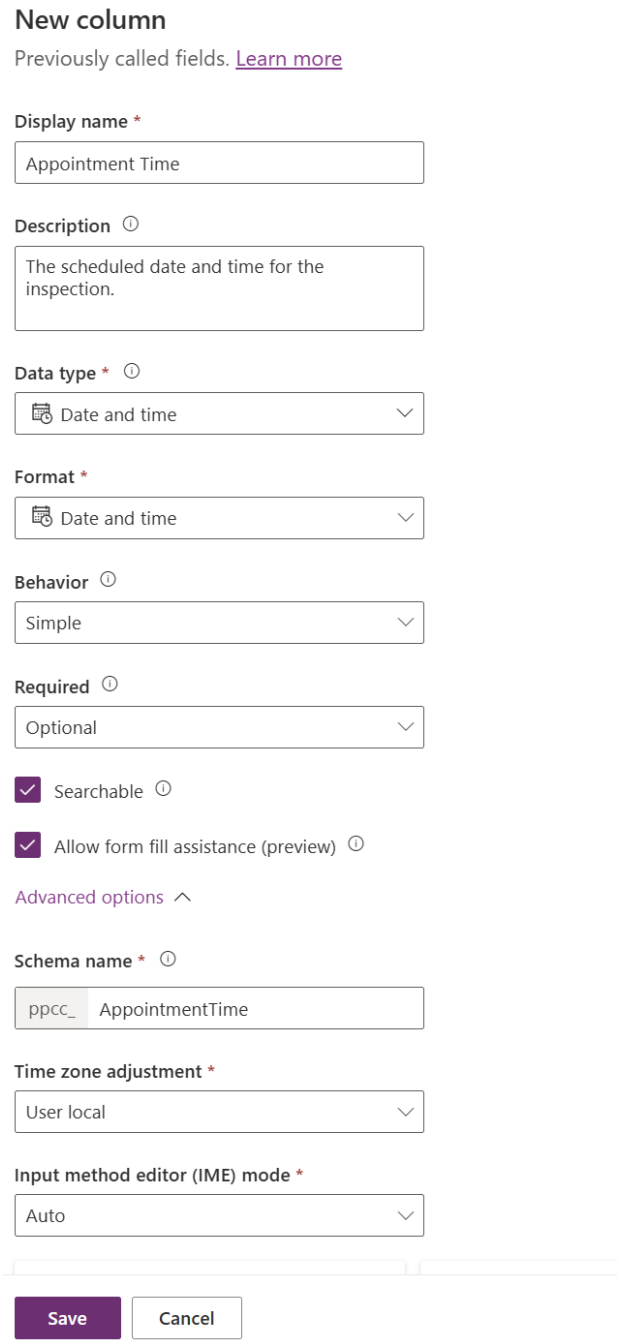


- Like Tables, you enter a **display name** which will appear in user interfaces.
- A **description** is always recommended to further clarify the purpose or provide instruction for data entry to the user. The description is displayed as a tooltip when you hover over the column in a model-driven app form.
- Selecting a **Data type** and **format** is the most important part of column creation. This cannot be changed after creation; you must create a new column if you want a different type.


New column
Previously called fields. Learn more
Display name *
Description
Data type *
Single line of text
Search
Text
Number
Date and time
Lookup
Choice
Currency
Autonumber
File
Formula
Prompt (Preview)
Type
Single line of text
Plain text
Text area
Rich text
Email
Phone number
Ticker symbol
URL
Multiple lines of text
Plain text
Rich text
Format

- The **Behavior** option is where you can select a non-default option such as *Calculated Column* or *Rollup* (if the data type supports it).





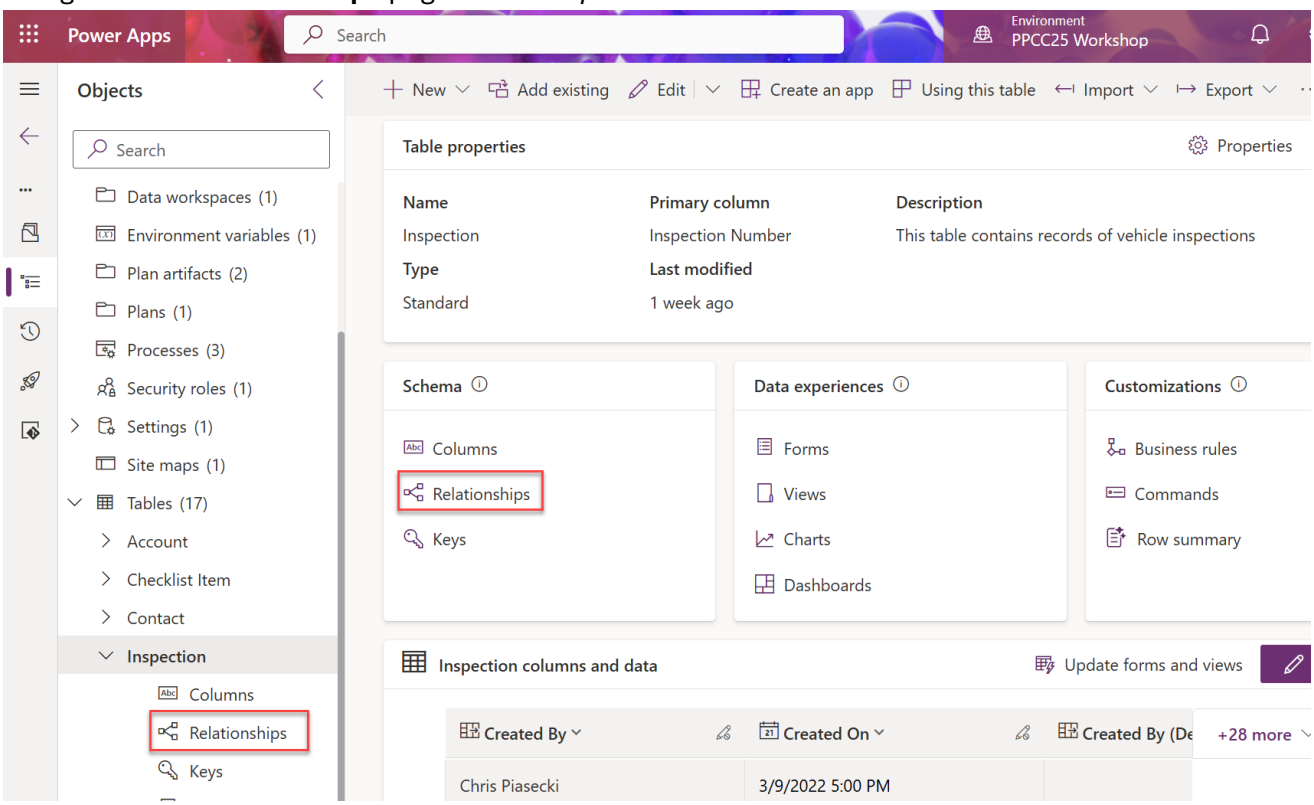
36. Click **Cancel** and confirm **OK** to discard changes.

Relationships

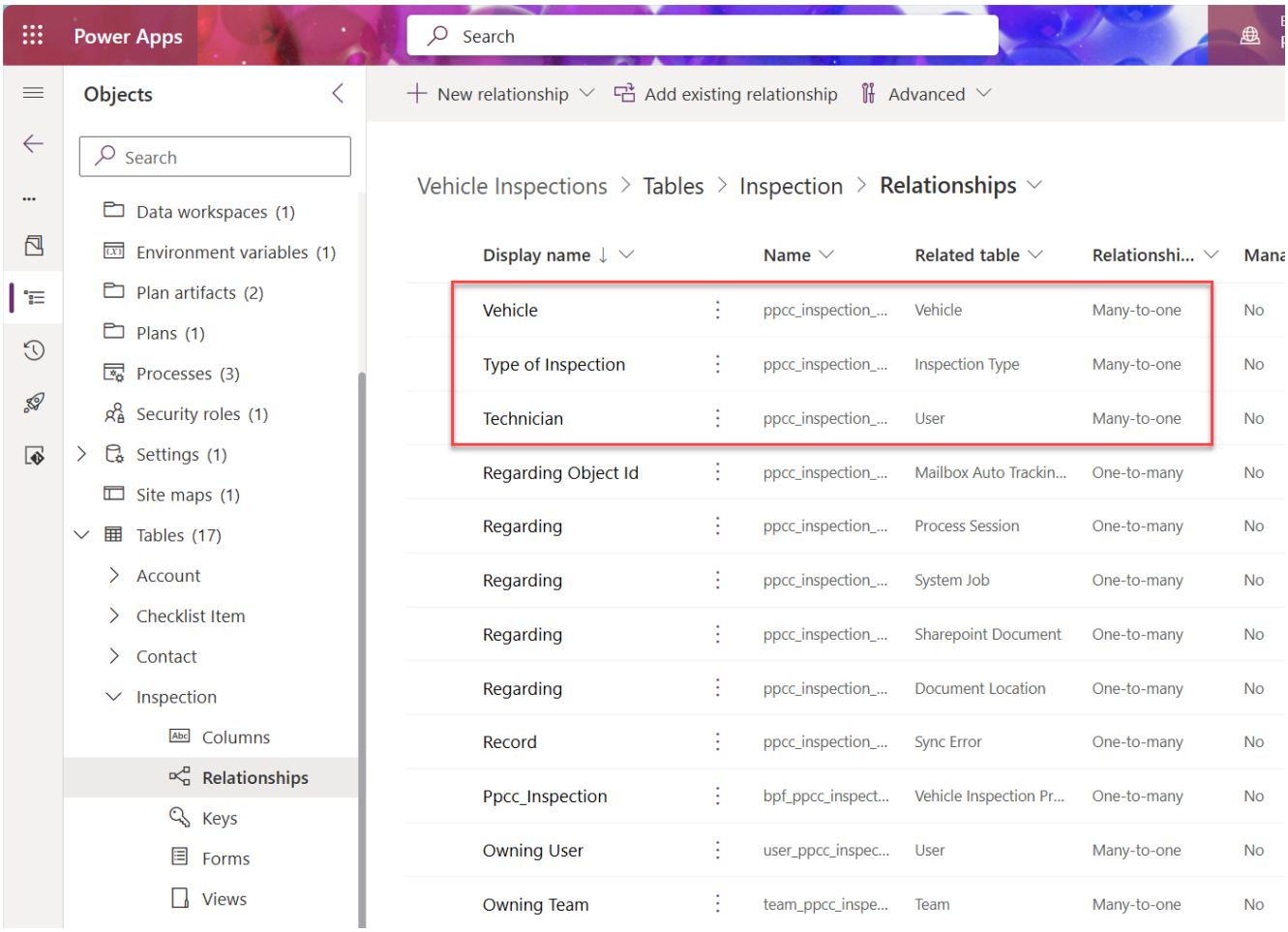
When a Lookup column is created, a many-to-one (N:1) relationship is created between the table the column was added to and the related table option in the column setting.

You will now explore the other way of creating a relationship.

37. Navigate to the **Relationships** page for the *Inspection* table



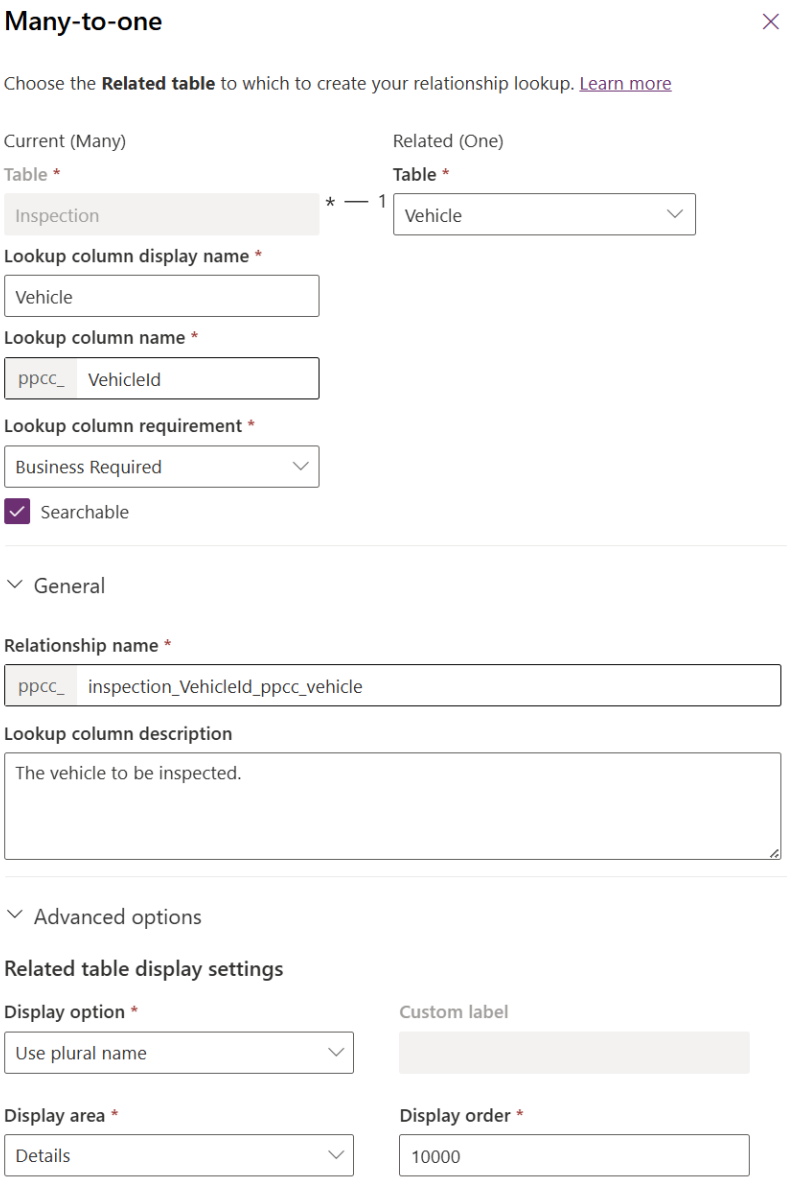
38. You will see the many-to-one relationships that were automatically created when the Lookup columns were created.



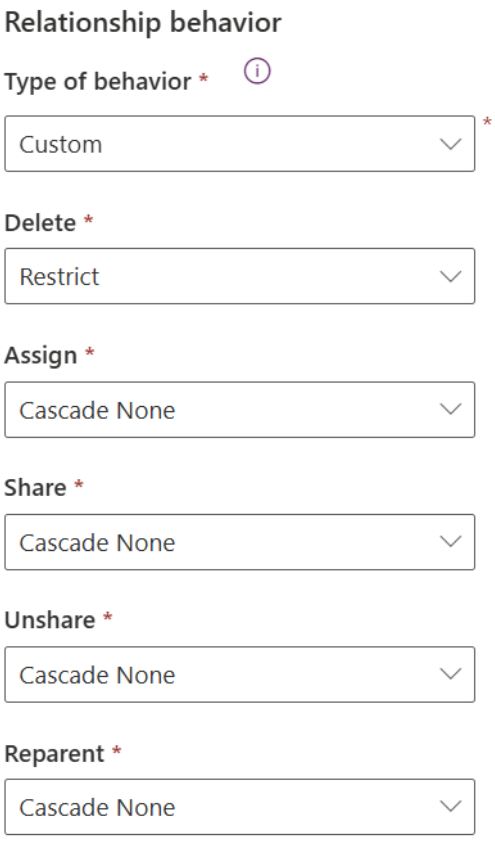
Let's look at how we would create a many-to-one relationship between *Inspection* and Vehicle.

Note: The relationship already exists – do not click Create/Save at the end

39. At the top of the screen, select **+New relationship**, then select **Many-to-one**
40. Enter in the relationship details:
    - Table: *Vehicle*
    - Lookup column display name: *Vehicle*
    - Lookup column name: *VehicleId*
    - Lookup column requirement: **Business Required**
    - Relationship name: **inspection_VehicleId_ppcc_vehicle**



41. Under Relationship behavior, configure the following:
    - Type of behavior: **Custom**
    - Delete: **Restrict**
    - Assign: **Cascade None**
    - Share: **Cascade None**
    - Unshare: **Cascade None**
    - Reparent: **Cascade None**

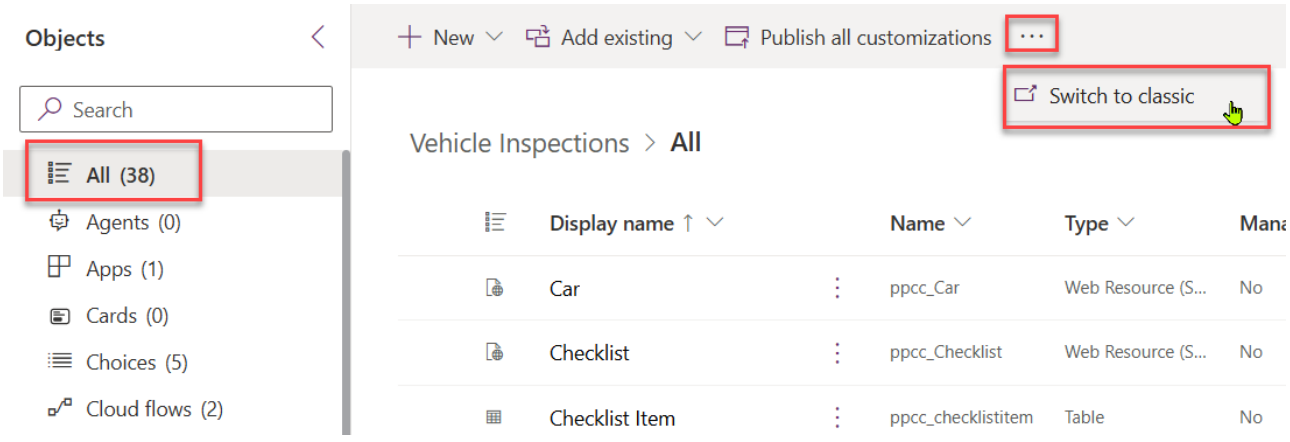


42. Click **Cancel** and **OK** in the confirmation dialog.

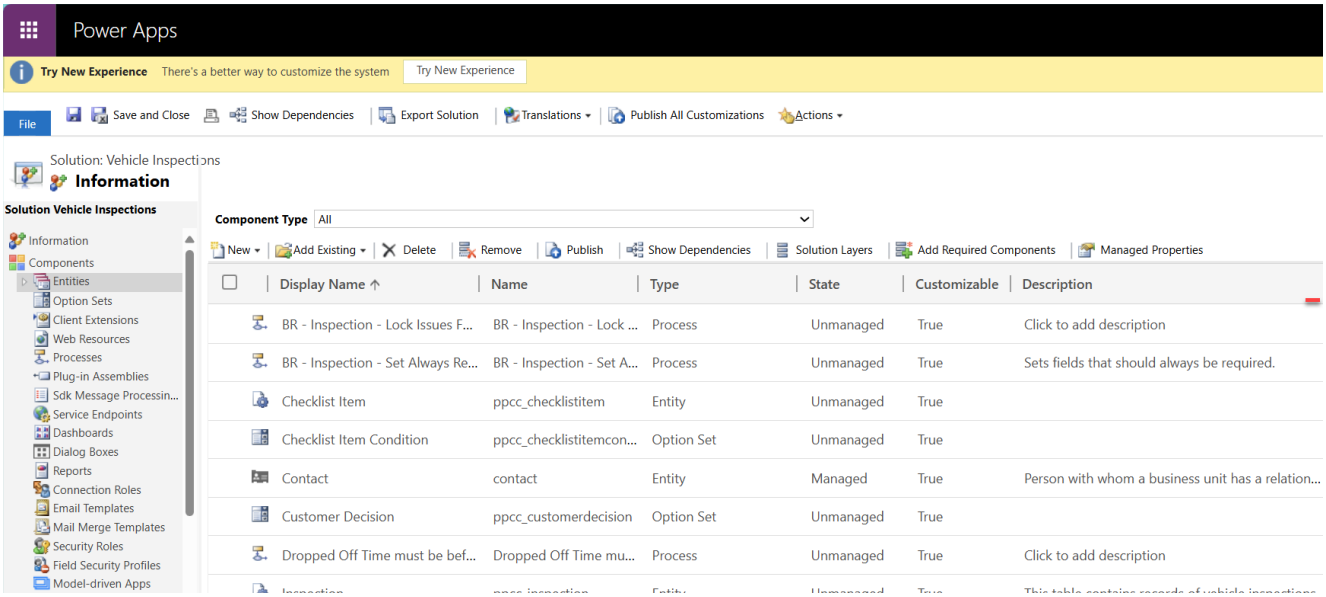
Column Mappings

To view column mappings, we must switch to the classic solution explorer as it has not been brought over to the modern maker experience.

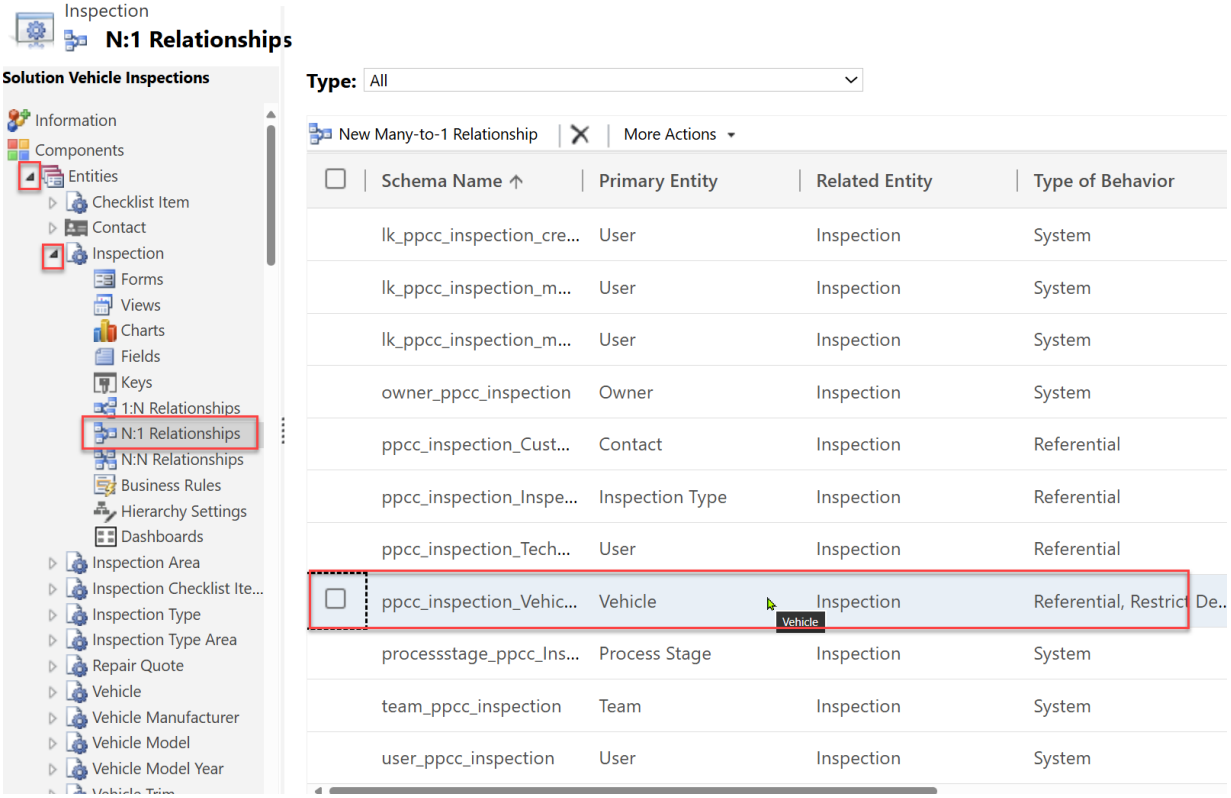
43. At the top of the solution object tree, select All.
44. Select the **3 ellipses (...)** on the command bar, then **Switch to classic**.



45. A new browser tab will open to the legacy solution explorer.

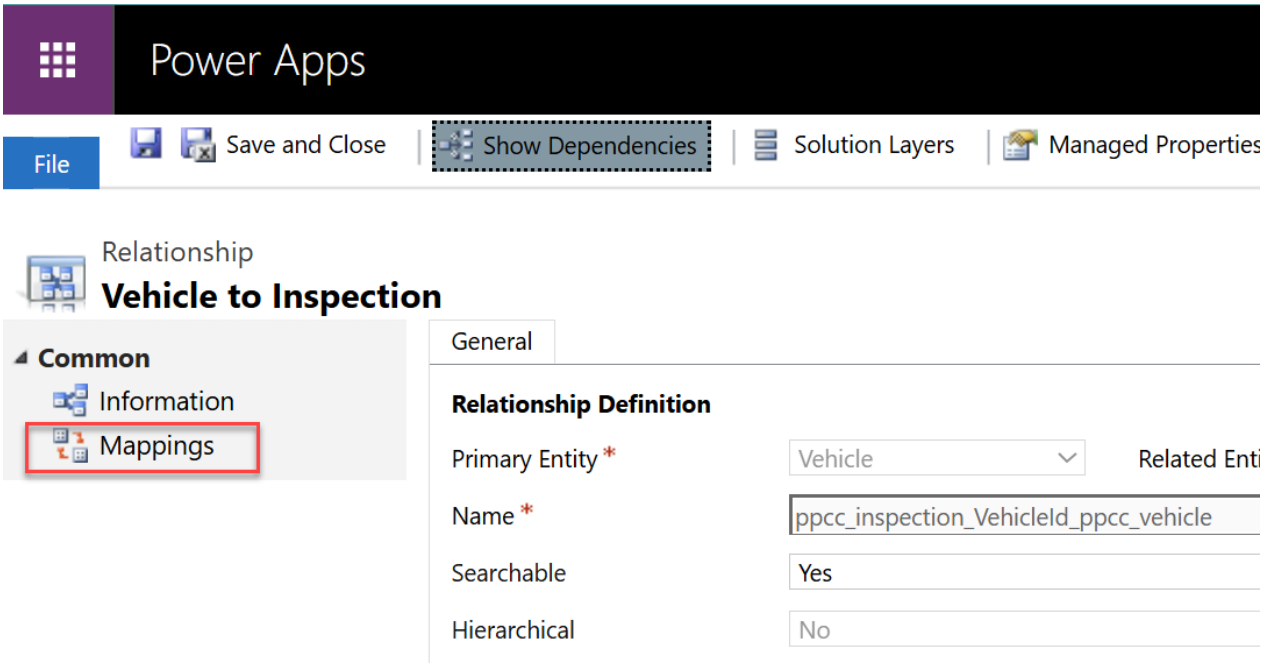


46. Under **Components** > **Entities** > **Inspection** > **N:1 Relationships**
47. Double click on the row with the **Display Name** of **Vehicle**

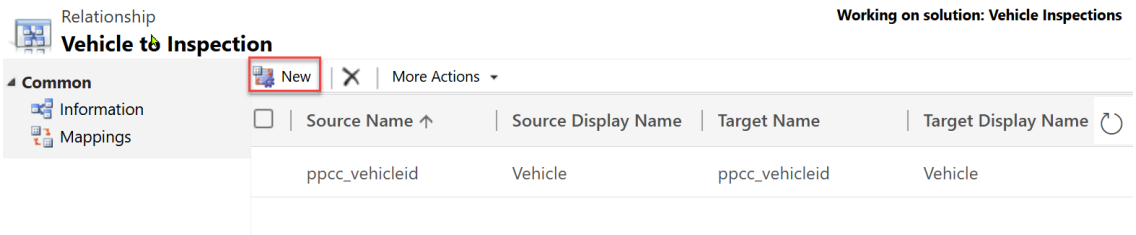


48. A dialog window will open with the relationship details.

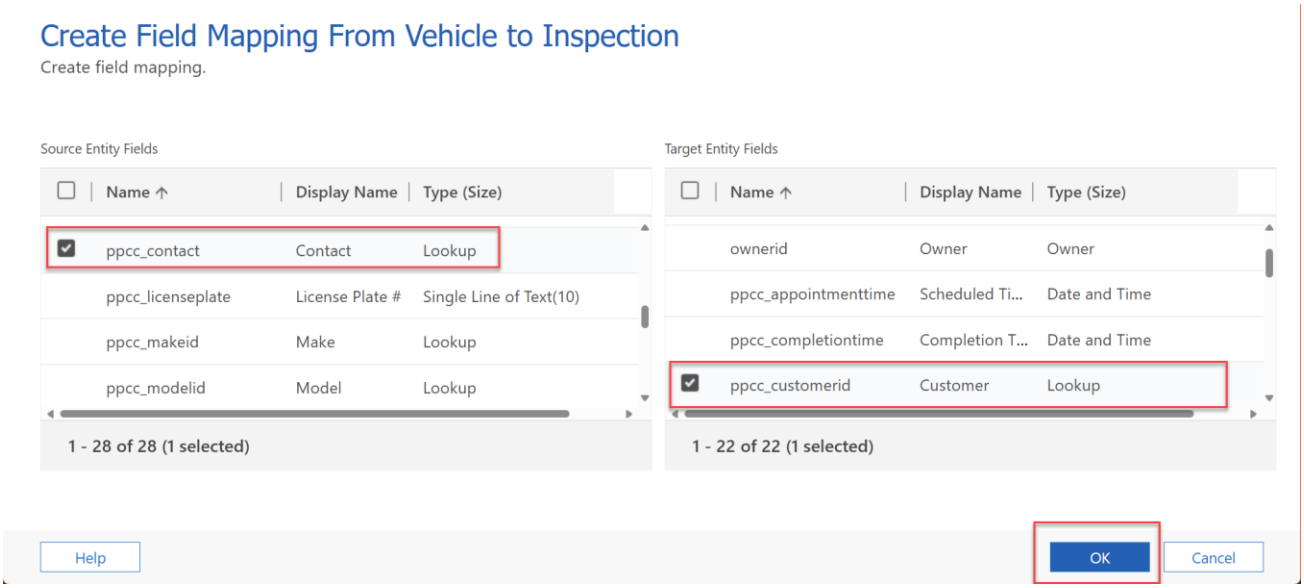
49. On the left hand navigation, select **Mappings**



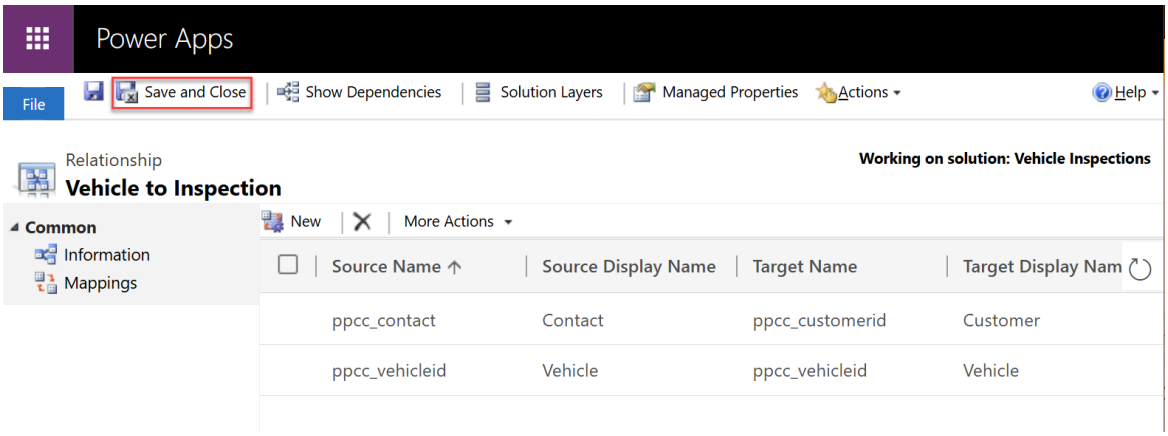
50. Existing mappings will be listed. By default a mapping is added between the Id column of the parent table in the relationship and the lookup column on the child table.
51. Click **New** to open the dialog to set up a new mapping.



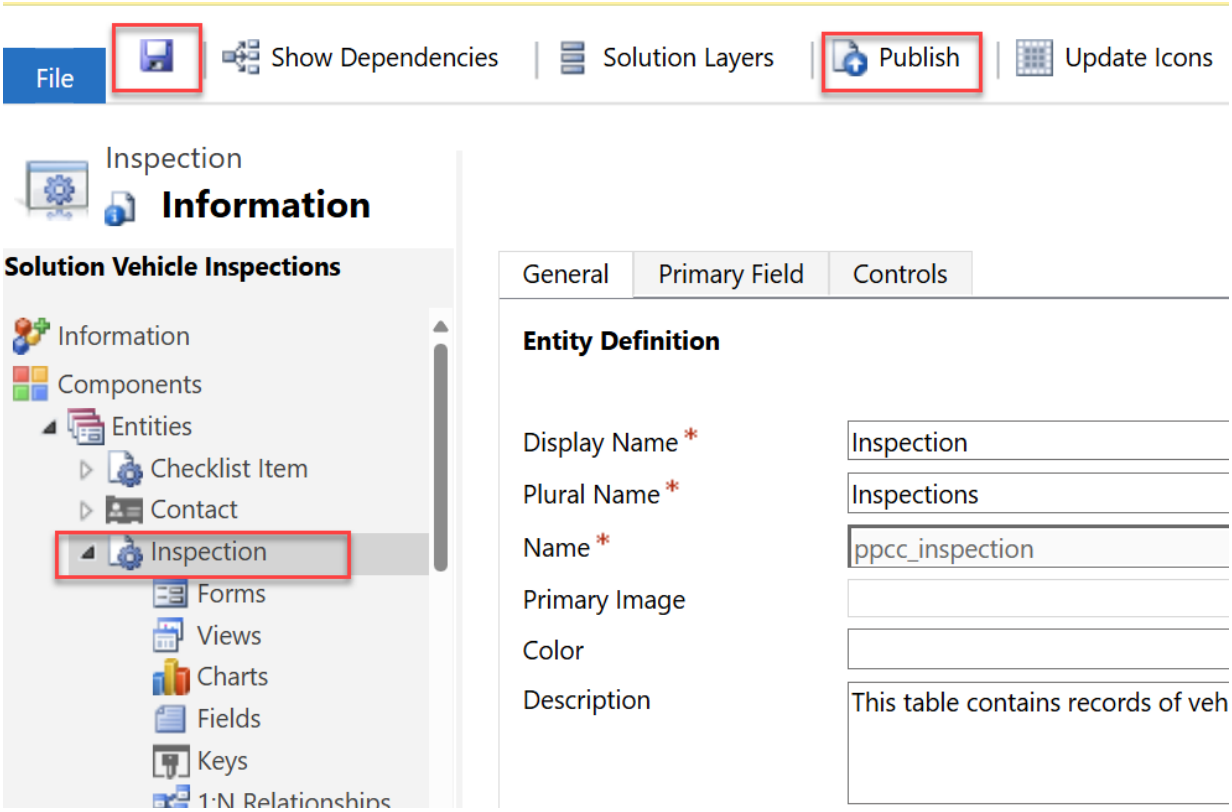
52. Select the following
    - Source: **Contact**
    - Target: **Customer**
    - Click **OK** to create the mapping



53. You will see the new mapping show in the list.
54. Click **Save & Close**.

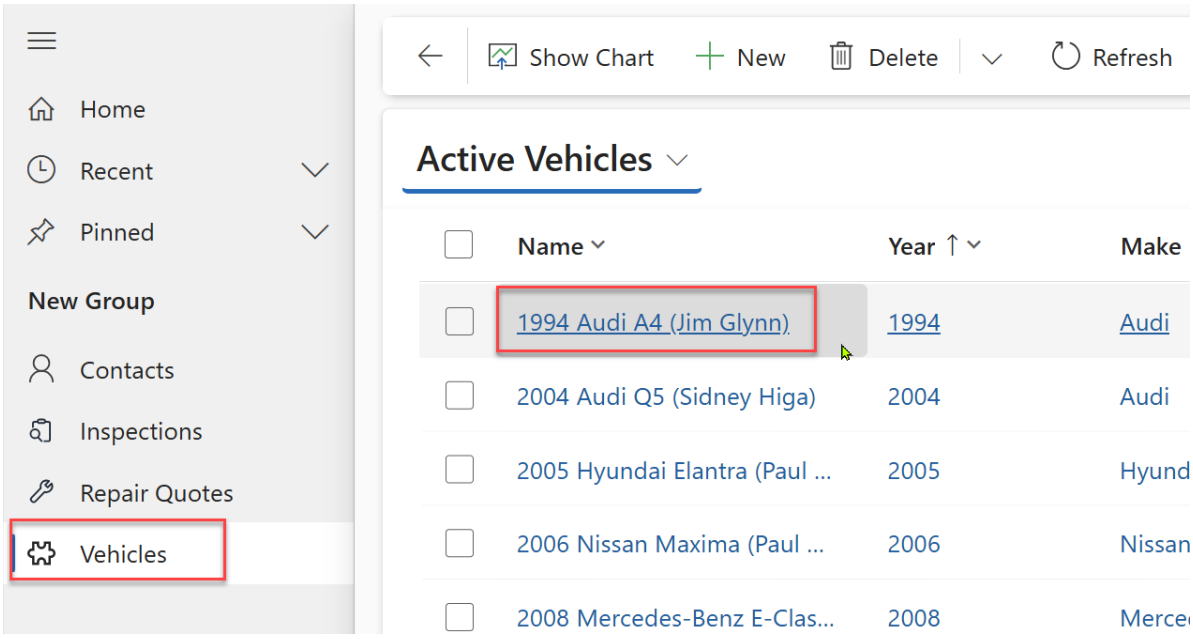


55. Back in the solution explorer, select the **Inspection** entity.
56. Click the **Save** (floppy disk) icon.
57. Select **Publish**.

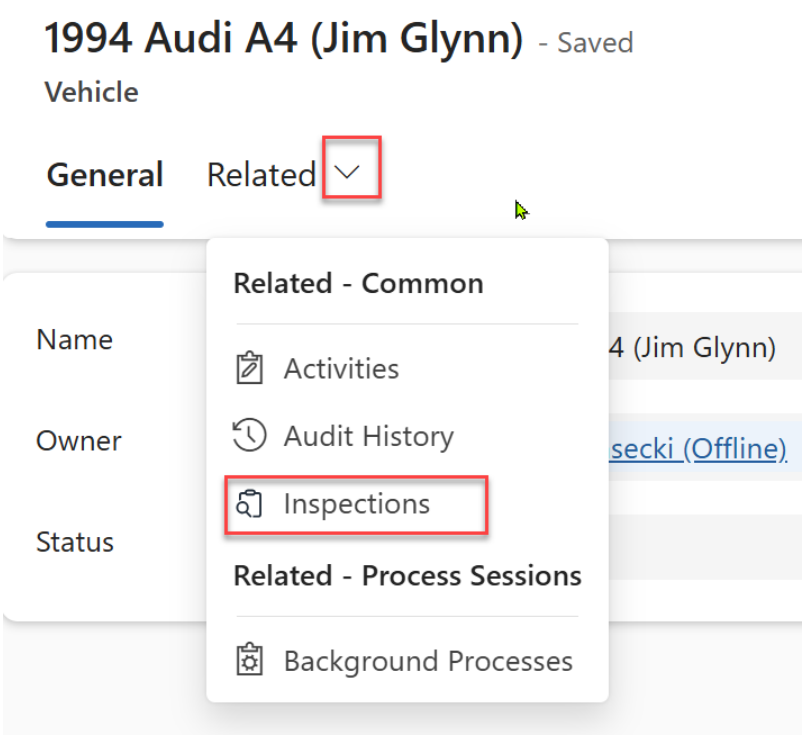


58. Once the **Publishing Customizations** message **clears**, you can close the solution explorer.
59. In the modern solution select **Apps**.
60. Select the row for **Vehicle Inspection Management**.
61. Click **Play** on the top of the command bar.
62. The model-driven app will open in a new browser tab.
63. Navigate to **Vehicles.**

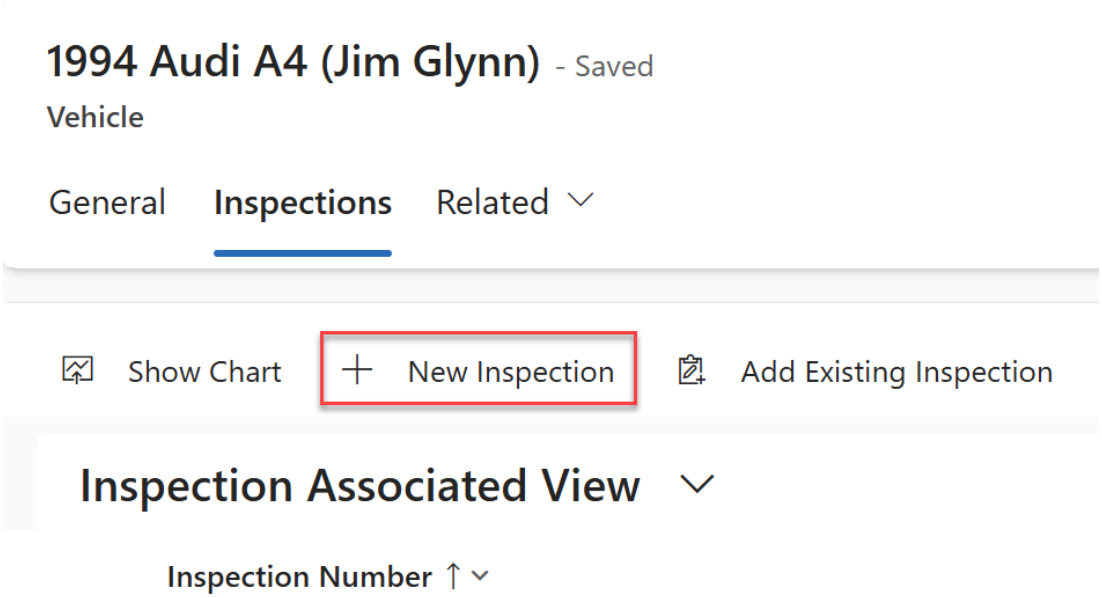
64. Open any existing row.



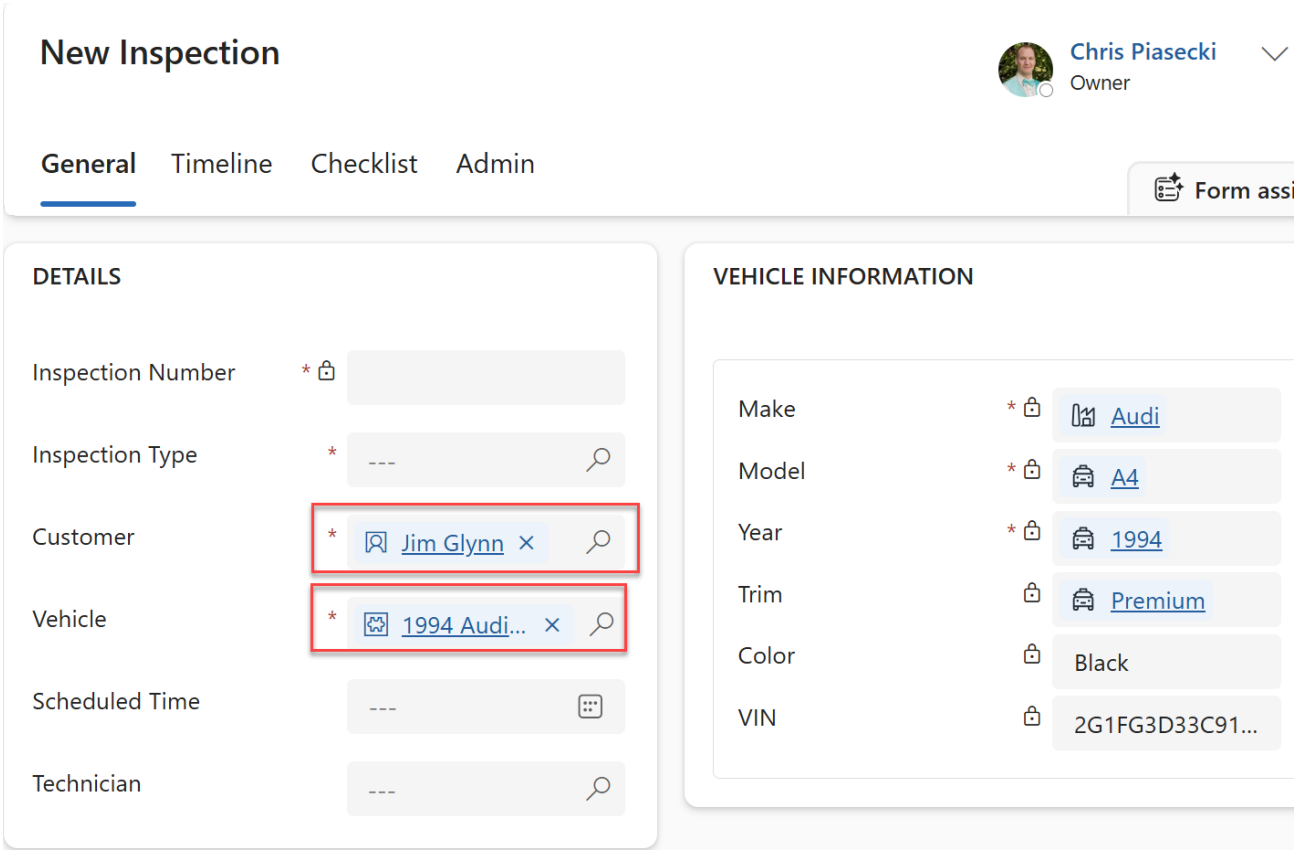
65. Select the **Related** tab, then **Inspections**.



66. Click **+New Inspection**.



67. The target fields from the two column mappings are automatically populated with the value from the parent Vehicle.

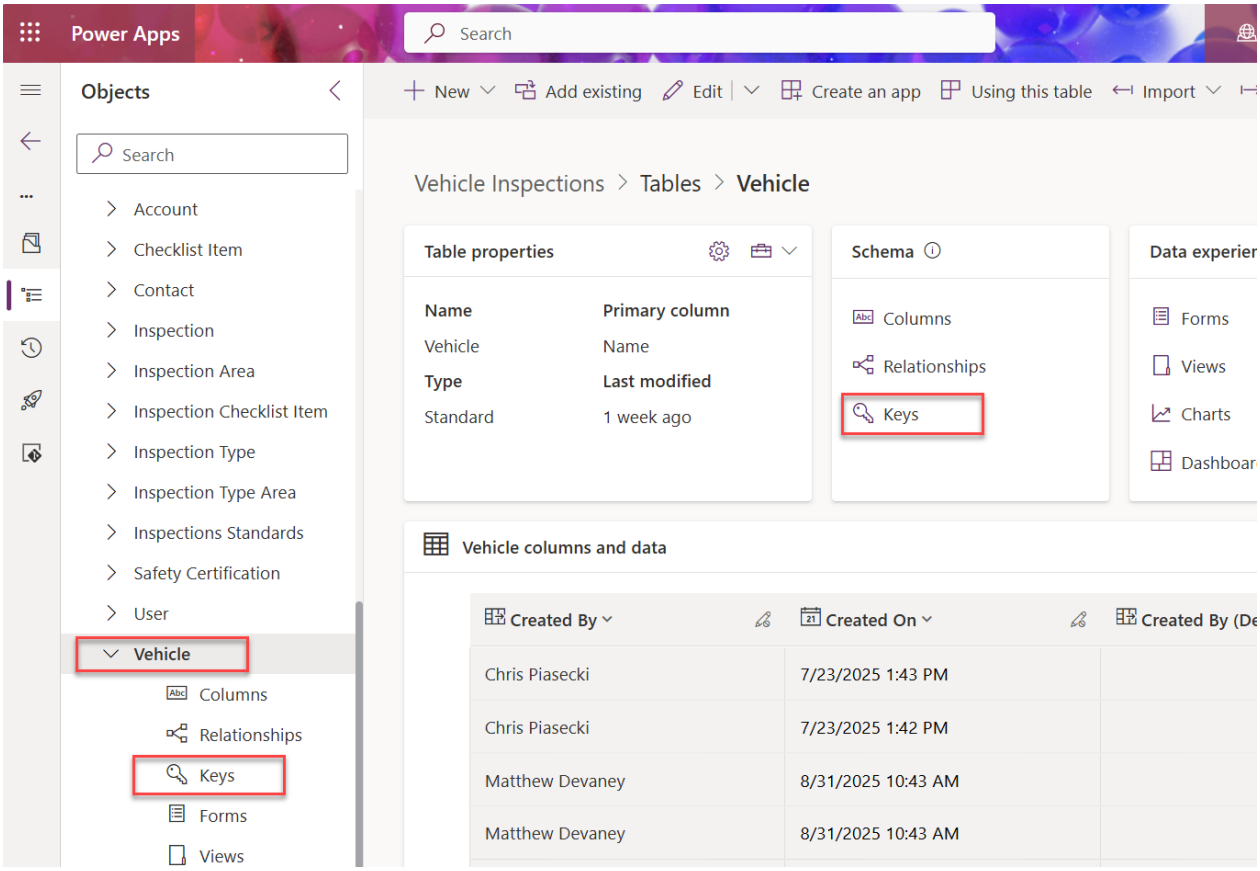


68. Click the back button without saving.

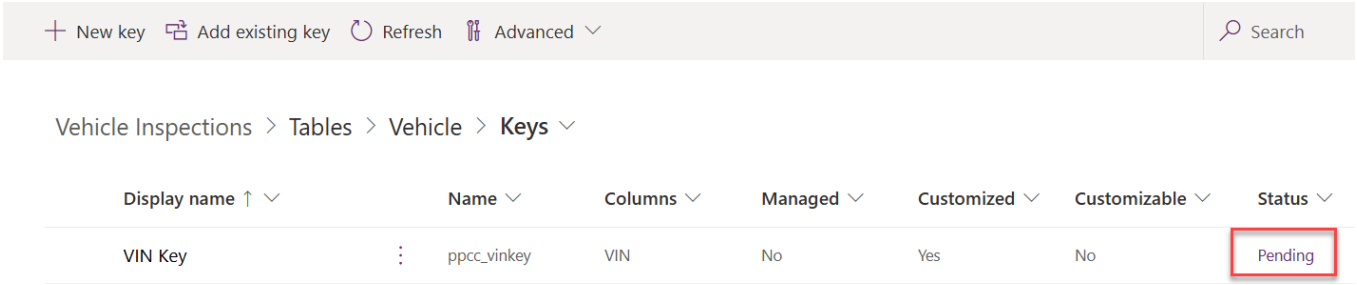
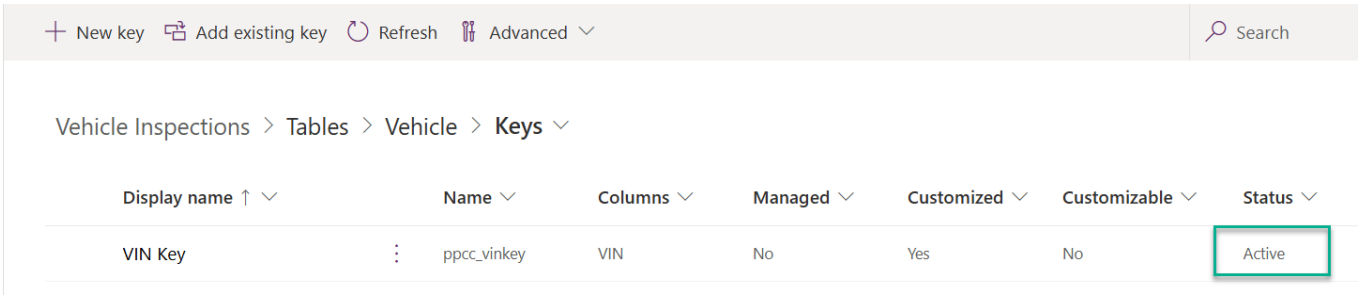
(Alternate) Keys

Now we want to add an alternate key to enforce uniqueness across records, prevent duplication, and overall improve data quality.

69. Under the Solution Objects tree, navigate to **Vehicle** > **Keys**.



70. At the top of the screen, select **+New Key**
71. Enter the following details then click save.
    - Display Name: **VIN Key**
    - Name: **VINKey**
    - Columns: **VIN**
72. Once the key is saved, it will initially have a status of **Pending.** A background job creates the index in the database and indexes the rows.



73. Once the indexing job is completed successfully, the status should change to **Active**.