

# WorkshopPLUS – Power Platform – Introduction to Power Apps & Power Automate for Dataverse with Lab

Lab 2 - Module 3: Model-driven App Lab

Student Lab Manual

## Conditions and Terms of Use

Microsoft Confidential - For Internal Use Only

This training package is proprietary and confidential, and is intended only for uses described in the training materials. Content and software is provided to you under a Non-Disclosure Agreement and cannot be distributed. Copying or disclosing all or any portion of the content and/or software included in such packages is strictly prohibited.

The contents of this package are for informational and training purposes only and are provided "as is" without warranty of any kind, whether express or implied, including but not limited to the implied warranties of merchantability, fitness for a particular purpose, and non-infringement.

Training package content, including URLs and other Internet Web site references, is subject to change without notice. Because Microsoft must respond to changing market conditions, the content should not be interpreted to be a commitment on the part of Microsoft, and Microsoft cannot guarantee the accuracy of any information presented after the date of publication. Unless otherwise noted, the companies, organizations, products, domain names, e-mail addresses, logos, people, places, and events depicted herein are fictitious, and no association with any real company, organization, product, domain name, e-mail address, logo, person, place, or event is intended or should be inferred.

© 2023 Microsoft Corporation. All rights reserved.

## Copyright and Trademarks

© 2023 Microsoft Corporation. All rights reserved.

Microsoft may have patents, patent applications, trademarks, copyrights, or other intellectual property rights covering subject matter in this document. Except as expressly provided in written license agreement from Microsoft, the furnishing of this document does not give you any license to these patents, trademarks, copyrights, or other intellectual property.

Complying with all applicable copyright laws is the responsibility of the user. Without limiting the rights under copyright, no part of this document may be reproduced, stored in or introduced into a retrieval system, or transmitted in any form or by any means (electronic, mechanical, photocopying, recording, or otherwise), or for any purpose, without the express written permission of Microsoft Corporation.

For more information, see Use of Microsoft Copyrighted Content at

<http://www.microsoft.com/about/legal/permissions/>

Internet Explorer, Microsoft, Microsoft Dynamics, Microsoft Dynamics logo, and Windows are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries. Other Microsoft products mentioned herein may be either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries. All other trademarks are property of their respective owners.



# Contents

**Lab 2: Create a model-driven app ..... 6**

    Exercise 1: Create and edit forms and views in a Dataverse table ..... 7

    Exercise 2: Create a model-driven app..... 14

## Lab 2: Create a model-driven app

### Introduction

In this lab, you will learn how to create a model-driven app. Additionally, you'll learn how to create and edit a custom view and a custom form on your Dataverse table to see how that's reflected in a model-driven app.

### Objectives

After completing this lab, you will be able to:

- Create a custom form on the Office Reservation table.
- Create a custom view on the Office Reservation table.
- Create a model-driven app for the Office Reservation table.
- Include the User table in the new model-driven app as a relationship.

### Prerequisites

- Basic knowledge of customizing Dataverse forms and views.
- Completion of Module 2 Lab 1.

### Estimated time to complete this lab

30 minutes

### Scenario

Contoso is asking their employees to return to the office. They are looking to build a tool for employees to report their office presence and a tool for front-desk employees at the offices to monitor the office presence.

In this exercise, you will create a model-driven app that is to be used by the front-desk operators to check on the reservations, as well as verify that reservations were approved and register employee presences.

## Exercise 1: Create and edit forms and views in a Dataverse table

### Objectives

After completing this exercise, you will be able to:

- Create and edit forms and views on top of Dataverse table to be used in applications associated with that table.

### Prerequisites

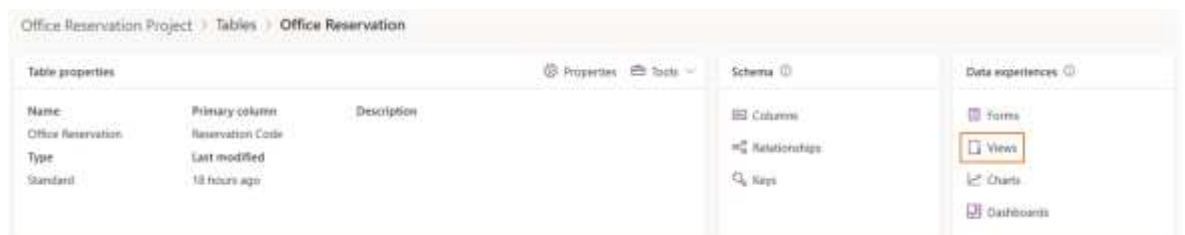
Completion of Module 2 Lab 1.

### Scenario

You have been asked to edit a form and create a view to add in the model-driven app that you will develop in Exercise 2 of this lab.

### Task 1: Edit a form and create new views in a Dataverse table

1. Your new table should still be open on your screen. **If it isn't:**
  - a. Ensure you are logged into your trial account. If you closed the browser, you could open a new In-Private browser and go to <https://make.powerapps.com>. Sign in using your Office365 credentials.
  - b. Make sure you are logged into the correct environment where you just built your solution and Dataverse table.
  - c. In the left pane select **Solutions**.
  - d. From the list of solutions in the environment, select **your solution** (Office Reservation Project).
  - e. Select the “Office Reservation” table inside the solution.
2. Select **Views** under the “Data experiences” section.



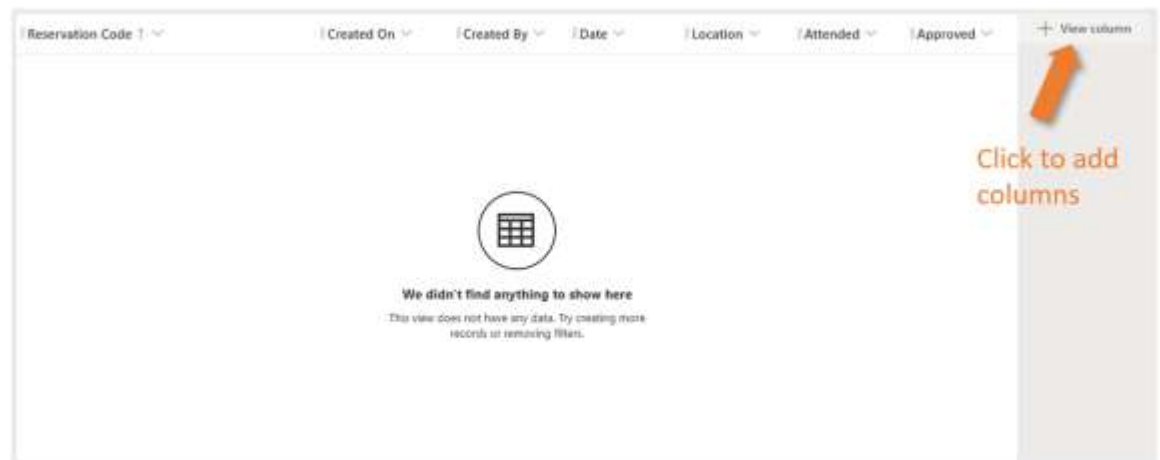
3. Select “Active Office Reservations” from the list.

Office Reservation Project > Tables > Office Reservation > Views ▾

Name ↑ ▾		View type ▾
Active Office Reservations	⋮	Public View default
Inactive Office Reservations	⋮	Public View
Office Reservation Advanced Find View	⋮	Advanced Find View default
Office Reservation Associated View	⋮	Associated View default
Office Reservation Lookup View	⋮	Lookup View default
Office Reservations Copilot View	⋮	Copilot View default
Quick Find Active Office Reservations	⋮	Quick Find View default

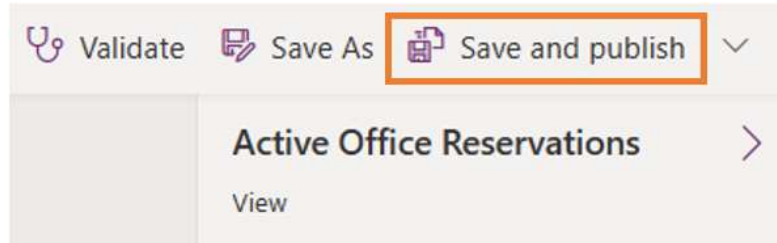
4. On the view designer page, click the **+ View column** button and select the following columns:
- Reservation Code
  - Created On
  - Created By
  - Date
  - Location
  - Attended
  - Approved

Your view should look like this:

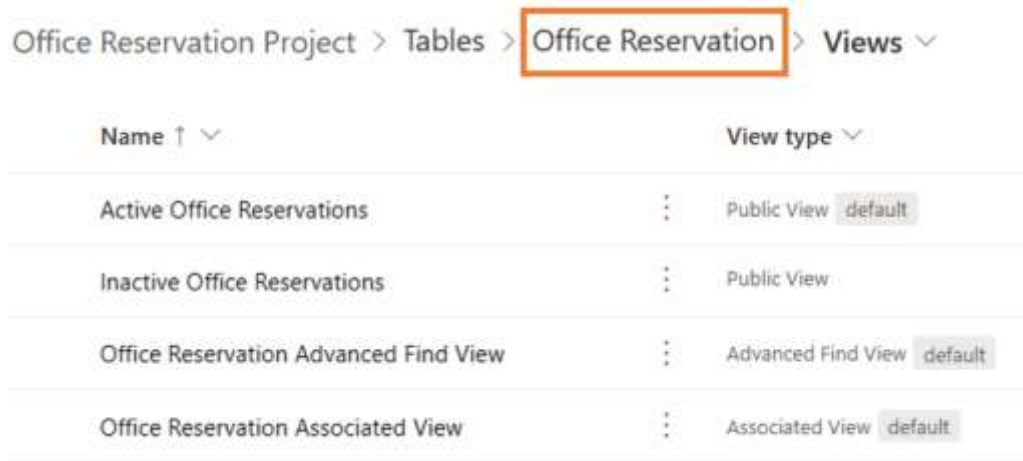




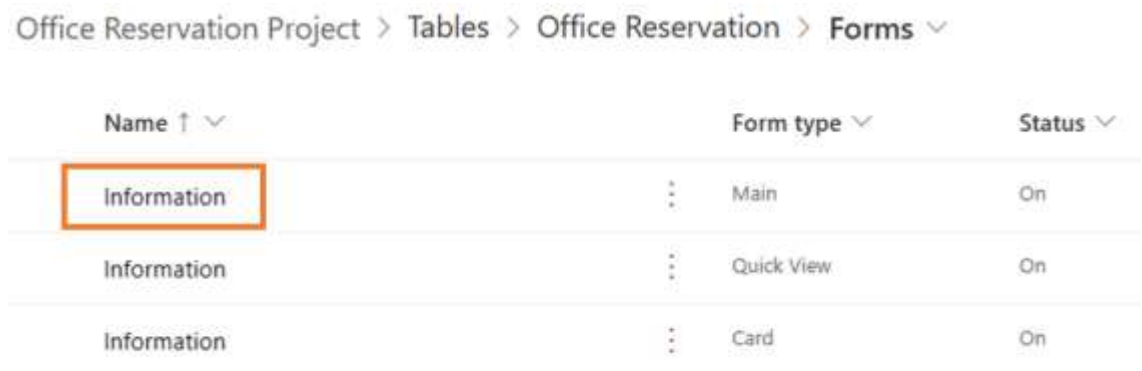
5. In the top right corner of the screen, select **Save and publish**.



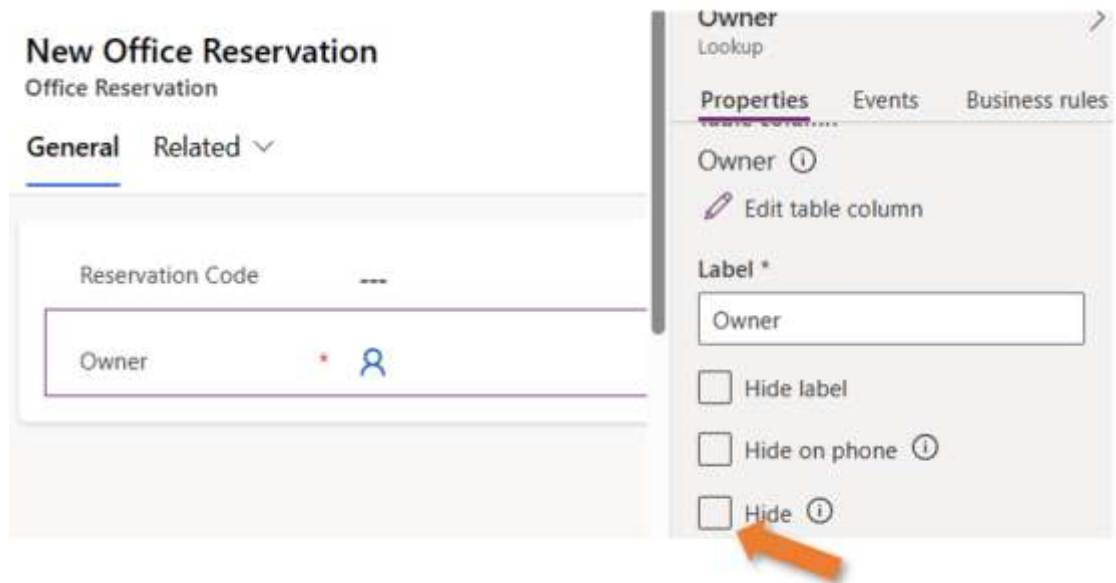
6. Go back to the main table screen by clicking **← Back** in the top left corner of the screen.
7. Select **Office Reservation** at the top of the screen to return to the main table page.



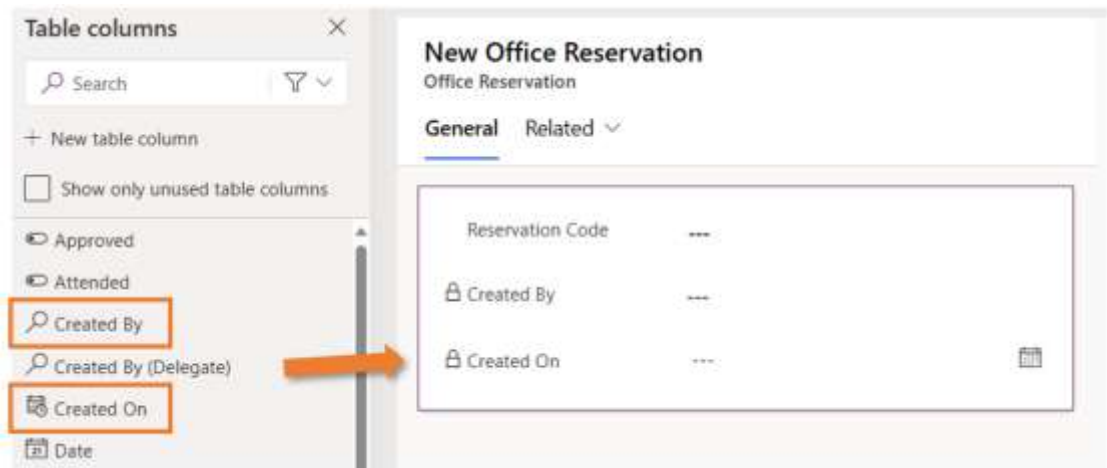
8. Select **Forms** under the “Data experiences” section towards the top of your screen, right above the Views you selected earlier.
9. Select the **Main Information** form from the list of forms.



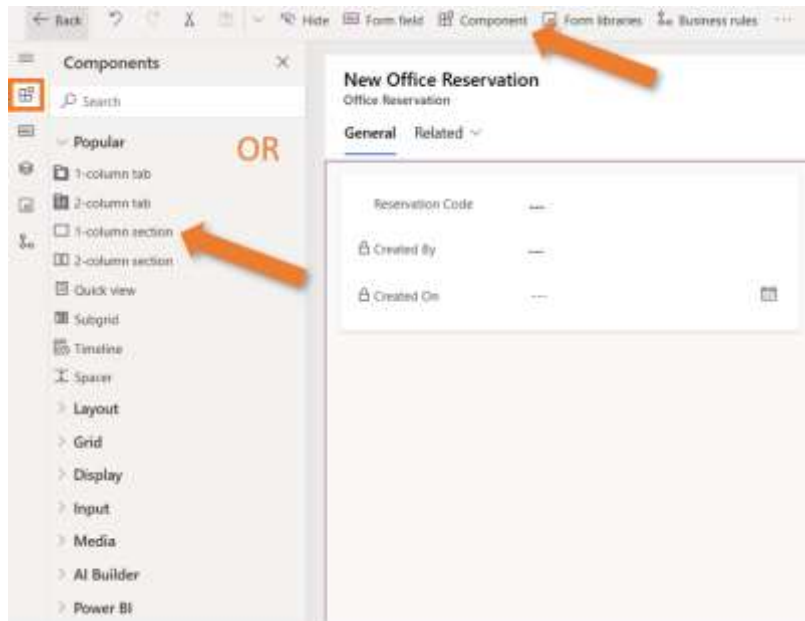
10. You should see the table columns available to you on the left pane. In the general section, click on the “Owner” column and hide it. Your column section may include the “Location” and “Date” fields since they are required. The screenshot below will not match what you see for that reason. You can leave those unhidden for now.



11. Add the “Created By” and “Created On” columns by dragging them from the left pane into the general section.



12. In the form you are editing, create a 1-column section by clicking the Component tab on the left side of your screen or the **Component** button on the top of your screen.



## New Office Reservation

Office Reservation

**General** Related ▾

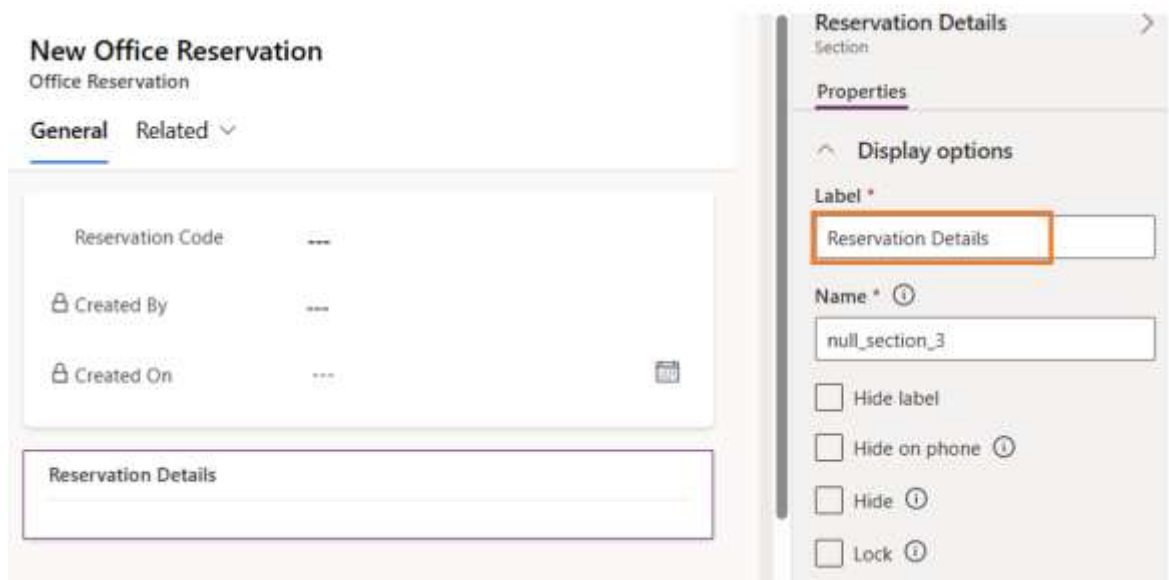
Reservation Code ---

Created By ---

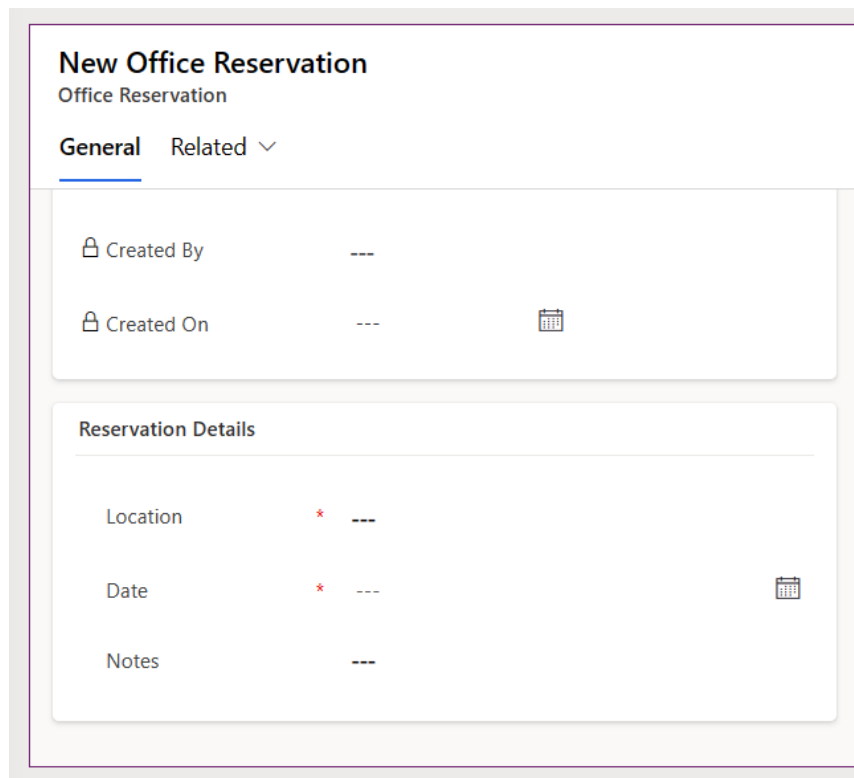
Created On ---

New Section

13. Change the label for this new section to “Reservation Details” by clicking the new section and changing the label in the properties pane on the right side of the screen.



14. Click on the “Form field” tab on the left side of your screen or the **Form field** button at the top of the screen.
15. You can drag “Location” and “Date” from the first section on the page. Drag the “Notes” column to your new “Reservation Details” section on the screen as well. It should now look like this and match the screenshots.



16. Create a new 1-column section under the “Reservation Details” section. Change the label of that section to “Indicate attendance”.
17. Drag the “Approved” and “Attended” columns to your new section.

Your form should look like this:

**New Office Reservation**  
Office Reservation

**General** Related ▾

Reservation Code ---

Created By ---

Created On ---

**Reservation Details**

Location \* ---

Date \* ---

Notes ---

**Indicate Attendance**

Approved ---

Attended ---

18. Click the **Save and publish** button in the upper right corner of the screen.



## Exercise 2: Create a model-driven app

### Objectives

After completing this exercise, you will be able to:

- Create a model-driven app connected to a custom table in Dataverse.
- Customize the model-driven app by adding pages and changing the app structure.

### Prerequisites

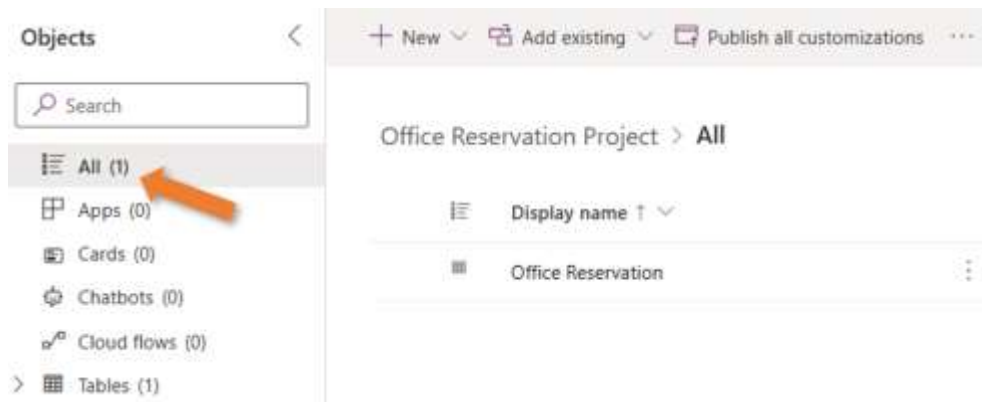
Completion of Lab 2 Exercise 1.

### Scenario

You have been asked to create a new app that is to be used by the front-desk operators to check on reservations, verify which reservations were approved and register employee presences.

### Task 1: Create a new model-driven app

1. Your new form should still be open on your screen. If it is, click the **← Back** button in the top left corner of your screen. **If it isn't:**
  - a. Ensure you are logged into your trial account. If you closed the browser, you could open a new In-Private browser and go to <https://make.powerapps.com>. Sign in using your Office365 credentials.
  - b. Make sure you are logged into the correct environment where you just built your solution and Dataverse table.
  - c. In the left pane select **Solutions**.
  - d. From the list of solutions in the environment, select **your solution** (Office Reservation Project).
2. Under the “Objects” pane of your solution explorer, select “All (1)”.



3. At the top of your screen, click the **+ New → App → Model-driven app** button.
4. Name your model-driven app “Operators Office Reservation” and click on **Create**.

## New model-driven app

Name \*

Operators Office Reservation

Description

✓ Advanced

Create

Cancel

5. Click on the **+ Add page** button at the top of your screen to create a new page in your model-driven app.
6. Select “Dataverse table” for the content and click **Next**.

## New page



This option gives you two pages for each table. You get a view, which displays the rows and columns of the table. You also get a form, which lets people interact with the table by editing existing rows, or by creating new ones.

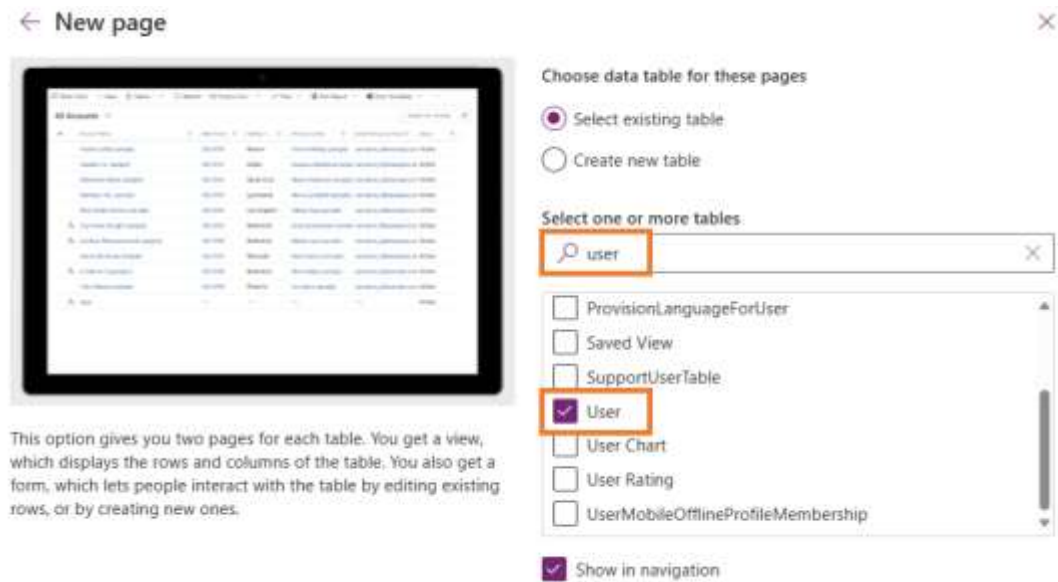
Choose content for the page

- ☒ Dataverse table
- ☐ Dashboard
- ☐ URL
- ☐ Web resource
- ☐ Custom page

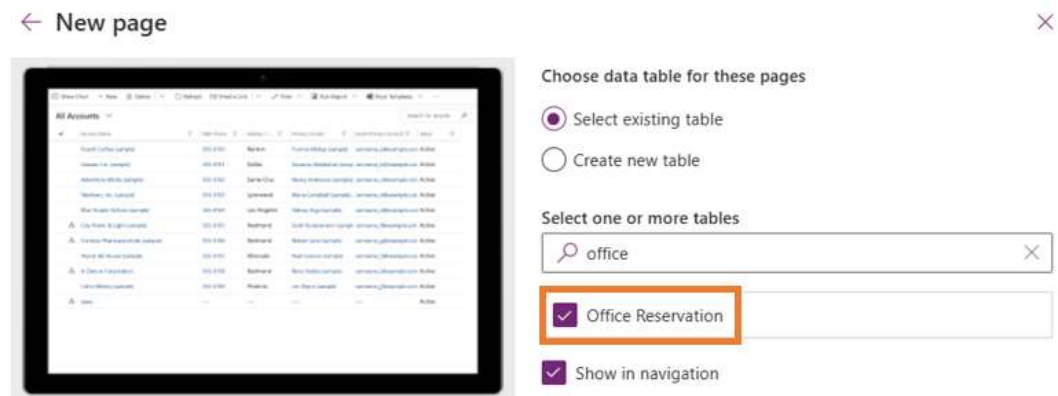
Next

Cancel

7. Select the “User” table by searching for user in the list of tables. Do not click the “Add” button yet.



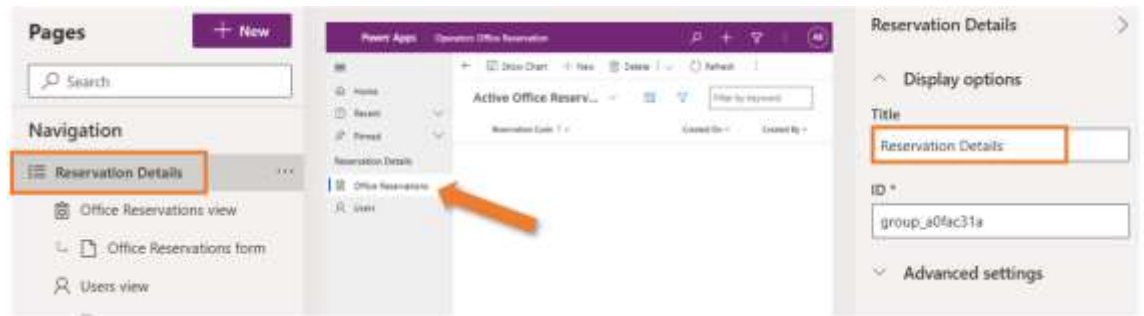
8. Select the “Office Reservation” table by searching for it in the list of tables. Click the **Add** button.



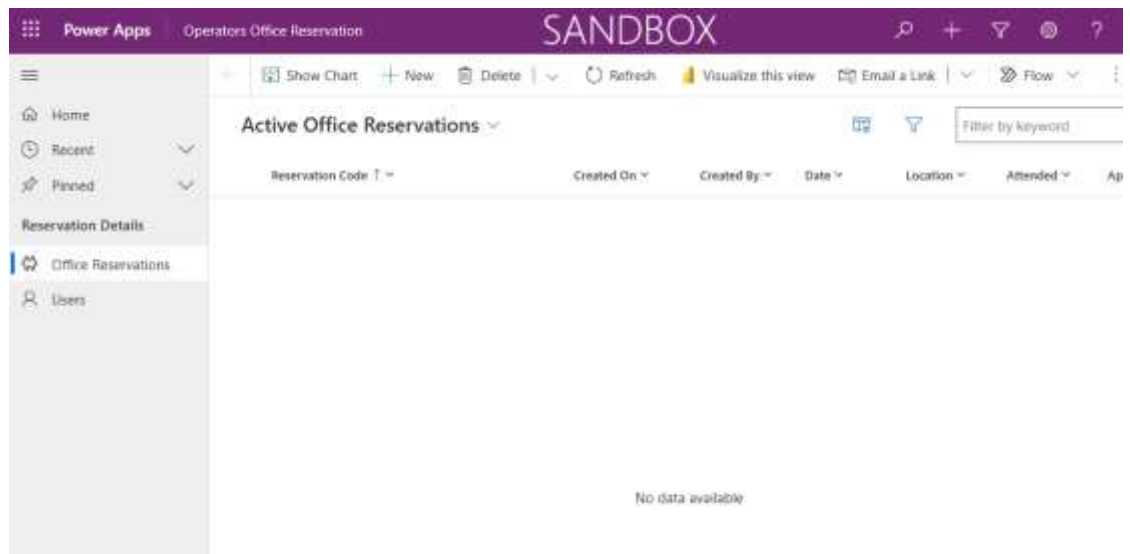
9. You should now see those tables under a navigation menu called “New Group” in your app.



10. Under the Pages tab on the left side of your screen, select “New Group”. Rename it to “**Reservation Details**” in the pane on the right side of your screen.



11. In the top right corner of your screen, select **Save** then select **Publish**.
12. Click the **Play** button in the upper right corner of your screen. You should see your new model-driven app with the tables you added. Note that the default view for “Office Reservations” includes all the fields you added to the view earlier.



13. If you click on **+ New** at the top of the screen, the default form will have the fields you added in this lab. It should look like this:

## New Office Reservation

### General

Reservation Code ---

 Created By ---

### Reservation Details

Location \* ---

Date \* ---

Notes ---

### Indicate Attendance

Approved No

Attended No

### You have successfully completed Lab 2!

In this lab, you edited the default view and form for your table to create an enhanced user experience for users of your model-driven app. Changes you made in the Dataverse table designer were reflected in the app you built automatically.

If you have additional time, you can explore the model-driven app studio to see which types of pages you can add and the settings for one of these applications. If you have any questions, reach out to your instructor.