

W8-S2 PRACTICE

MODALS & INPUTS

Learning objectives

- ✓ Use **Drop-Down** and **Date Picker** inputs
- ✓ Understand the usage of **Futures**
- ✓ Display a **Snack Bar** with **Scaffold Messenger** to undo the remove action
- ✓ Open and close **Modals**



No AI tools allowed to solve this practice



How to submit?

- ✓ **Push** your final code on **your GitHub repository**
- ✓ Then **attach the GitHub path** to the MS Team assignment and **turn it in**



Before practice, to be prepared!

Read the following documentation to be ready for this practice:

[Modals](#)

[Text Field](#)

[Dropdown Button](#)

[List View / Builders](#)

[List Tile](#)

[Checkbox List Tile](#)

[App bar](#)

[What is the BuildContext?](#)

[TextField](#)

[Dismissible](#)

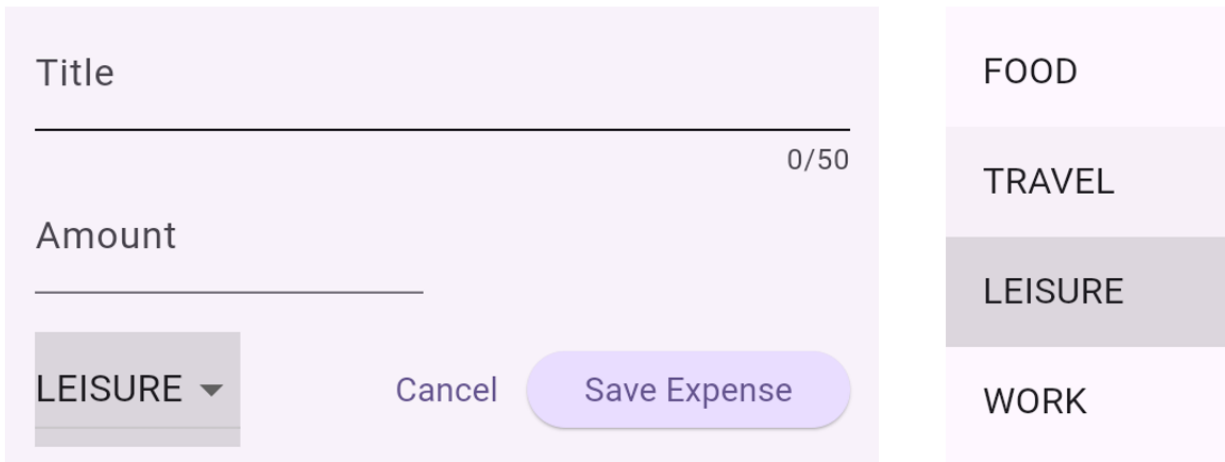
[Snack Bars](#)



EX 1 – Add a drop-down button

For this exercise, you can start from the provided start code, or use your Expenses app current code.

- ✓ You need to add [a dropdown button](#) to display and **select a category** of expense
- ✓ When clicking on Save Expense, you need to create the item **using the selected category**
- ✓ Categories need to be displayed with **uppercase**



The image shows a user interface for adding an expense. On the left is a form with three input fields: 'Title' (with a character count '0/50'), 'Amount', and a dropdown menu currently showing 'LEISURE'. Below the form are two buttons: 'Cancel' and 'Save Expense'. On the right is a vertical list of category options: 'FOOD', 'TRAVEL', 'LEISURE' (which is highlighted with a darker background), and 'WORK'.

When clicking on the drop down, we should be able to select an Expense category



Are you lost?

To manage a dropdown, we cannot use a controller, as we did with the TextField. Instead, we need to manage a state in the Stateful widget to keep the selected category.

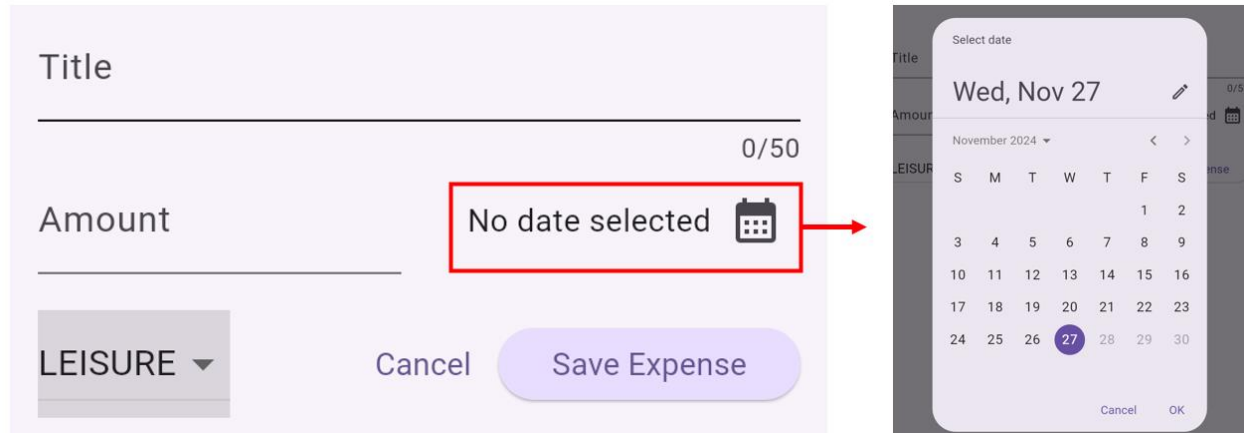
Here is the recommended way:

- Create a new state: the selected category (type Category)
- Create the Dropdown Button widget
- Set the Dropdown items, using the categories from the enum Category
- Implement the Dropdown onChanged event to change the state (setState)

EX 2 – Add a time picker

For this exercise, you can start from the provided start code, or use your Expenses app current code.

- ✓ You need to add a [Date picker](#) to display and **select the date** of expense
 - The date picker can be composed of a **Text** and an **IconButton**
 - When clicking on the icon the date picker shall be displayed
- ✓ When clicking on **Save Expense**, you need to create the expense **using the selected date**



The date picker can be composed of a Text and an IconButton

Are you lost?

Are you lost using a Date picker in Flutter? [Check this code](#) !

You need to learn [how to use Future in Dart](#), as you get the date in an asynchronous way!

For this need, you need to add a **state in your stateful widget** to manage this new field

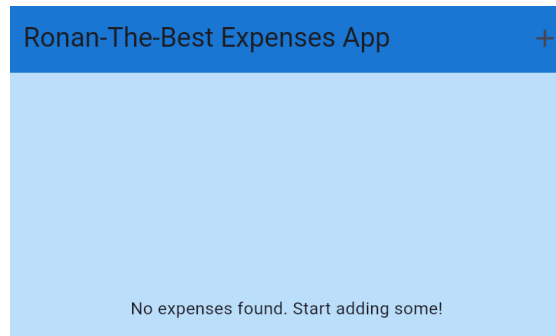
Here is the recommended way:

- Create a new state: the **selected date** (type DateTime)
- Create an **icon button** to select the date
- Display date picker using `showDatePicker`
- Manage the future using `async/await`
- Change the state (`setState`) if a date has been selected

EX 3 – Empty list and Snack bar

Let's make the app a bit more user friendly with 2 small improvements

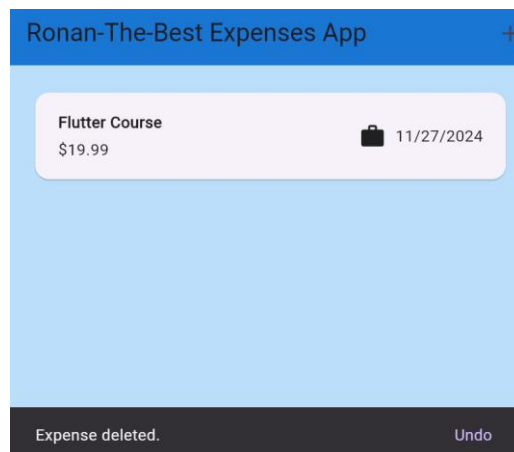
Q1 - When the list of expenses **contains no value**, display a text message:



When the list is empty, display an informative text

Q1 - After removing an item, display a Snack Bar to allow user to undo the operation

- Note: By undo we mean: insert the Expense again in the list!



After removing an item, display a Snack Bar to allow user to undo the operation

🧐 Are you lost?

How to use a snack bar in Flutter? [Read the documentation](#) !

After having removing a Expense from the list, you can do the following:

- First clear the snack bar if needed: `clearSnackBars`
- Then display a snack bar: `showSnackBar`
 - Duration: 3 seconds
 - The snack bar action should be to **insert again** the Expense you have just removed!

EX 4 – Device Converter

You need to complete the start code to manage a device converter

- The dollar TextField shall allow only numbers
- The drop down shall allow to select devices (euro, riels, dong)
- Convert the dollar into the selected device, using a well-defined ratio

TIP

- *Think about the input's listeners: do you need to add states or to use controllers?*
- *You should re-use what you have learnt during the previous exercises !*

100

Converter

Amount in dollars:

\$ 150

Device:

EURO ▼

Amount in selected device:

135