**Exercise: Render a large list using FlatList**

**Overview**

By now, you should be familiar with the **FlatList** component and how to use it to render large lists. Recall that **FlatList** renders items lazily, which means that it only renders items as they're needed rather than the entire list. This makes it ideal for rendering large lists without slowing down app performance.

In this exercise, you will update the Little Lemon app to display a large list of menu items and each item’s price. You will accomplish this by using a **FlatList** component.

**Scenario**

The Little Lemon app needs to display a large list of menu items on the screen, along with the price of each item. You have been asked to use the **FlatList** component to render these menu items efficiently. You should also make sure that the component you write is readable and clean.

Below is the array of menu items and the price for each item. Note that each item also has an **id** value.

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

const menuItemsToDisplay = [

  { name: 'Hummus', price: '$5.00', id: '1A' },

  { name: 'Moutabal', price: '$5.00', id: '2B' },

  { name: 'Falafel', price: '$7.50', id: '3C' },

  { name: 'Marinated Olives', price: '$5.00', id: '4D' },

  { name: 'Kofta', price: '$5.00', id: '5E' },

  { name: 'Eggplant Salad', price: '$8.50', id: '6F' },

  { name: 'Lentil Burger', price: '$10.00', id: '7G' },

  { name: 'Smoked Salmon', price: '$14.00', id: '8H' },

  { name: 'Kofta Burger', price: '$11.00', id: '9I' },

  { name: 'Turkish Kebab', price: '$15.50', id: '10J' },

  { name: 'Fries', price: '$3.00', id: '11K' },

  { name: 'Buttered Rice', price: '$3.00', id: '12L' },

  { name: 'Bread Sticks', price: '$3.00', id: '13M' },

  { name: 'Pita Pocket', price: '$3.00', id: '14N' },

  { name: 'Lentil Soup', price: '$3.75', id: '15O' },

  { name: 'Greek Salad', price: '$6.00', id: '16Q' },

  { name: 'Rice Pilaf', price: '$4.00', id: '17R' },

  { name: 'Baklava', price: '$3.00', id: '18S' },

  { name: 'Tartufo', price: '$3.00', id: '19T' },

  { name: 'Tiramisu', price: '$5.00', id: '20U' },

  { name: 'Panna Cotta', price: '$5.00', id: '21V' },

];

You will use this array to render the items within the screen. Also, use the **id** for each item as the key to the **FlatList** component.

The screenshots below demonstrate how your app should look after you complete this exercise:



The colors displayed in the images above can be applied with the following values:

*#F4CE14, #EE9972, #333333, #*EDEFEE, *black* and *white.*

**Starter Code**

Use the starter code and build upon it to complete this exercise. You can download the code from this zipped folder:

[flatlist---starter-code](https://d3c33hcgiwev3.cloudfront.net/rWnBn4owSSqjZaUCps06gA_a5412bcad29a454db514bf6a85f8caf1_flatlist---starter-code.zip?Expires=1714176000&Signature=Xq-ORM7OvaeJ78GAqB69dsjLSI6HP~6JECDX~Ltog~9KPbXam~IH4FSsGJrg-QajLuOWLrNdXykDPot8YHnwVNkD~dcb9mnsT3FRQ2N-0GSseDN3OTL3Eb2dfDvOikZSzPrYKcDh1cJh6VSA4zh35TntB8f5kioMBoE132fffRU_&Key-Pair-Id=APKAJLTNE6QMUY6HBC5A" \t "_blank)

[ZIP File](https://d3c33hcgiwev3.cloudfront.net/rWnBn4owSSqjZaUCps06gA_a5412bcad29a454db514bf6a85f8caf1_flatlist---starter-code.zip?Expires=1714176000&Signature=Xq-ORM7OvaeJ78GAqB69dsjLSI6HP~6JECDX~Ltog~9KPbXam~IH4FSsGJrg-QajLuOWLrNdXykDPot8YHnwVNkD~dcb9mnsT3FRQ2N-0GSseDN3OTL3Eb2dfDvOikZSzPrYKcDh1cJh6VSA4zh35TntB8f5kioMBoE132fffRU_&Key-Pair-Id=APKAJLTNE6QMUY6HBC5A" \t "_blank)

**Instructions**

**Step 1: Create a component to display menu items and use FlatList**

To begin, create a new component to display the menu items and prices. You can call this component **MenuItems**.

Create this file within the existing components folder as follows: **components/MenuItems.js**

Within this component, use the array **menuItemsToDisplay** provided in the scenario to pass to the **data** prop of the **FlatList** component.

Then, configure the **renderItem** prop of the **FlatList** component to render each item’s name and the price per item. Provide the **id** as the key to the **FlatList**.

**Hint***:* You can create multiple components within the same file to keep code clean.

**Step 2: Style the component**

Next, you will style the new component you created to match the screenshots shown above. It is best to do this by adding styles to the StyleSheet, and then calling them within the component. Make sure to provide meaningful names to all your styles, making them easier to reference later. For example, **headerText** is a more useful name than **style1**.

**Hint**: You may want to use the following style props: **backgroundColor** and **fontSize**. When using a StyleSheet, don’t forget to define it within the file first by using **const**.

**Step 3: Render the component from App.js**

To finish this exercise, you’ll need to render the component you just created so that it renders the menu items on the screen. Render this component from the **App** component and check the results in the emulator to verify that it renders correctly.

**Conclusion**

By completing this exercise, you will demonstrate your understanding and ability to configure and utilize the **FlatList** component in order to render a large list of items.

Mark as completed

Like

Dislike

Report an issue