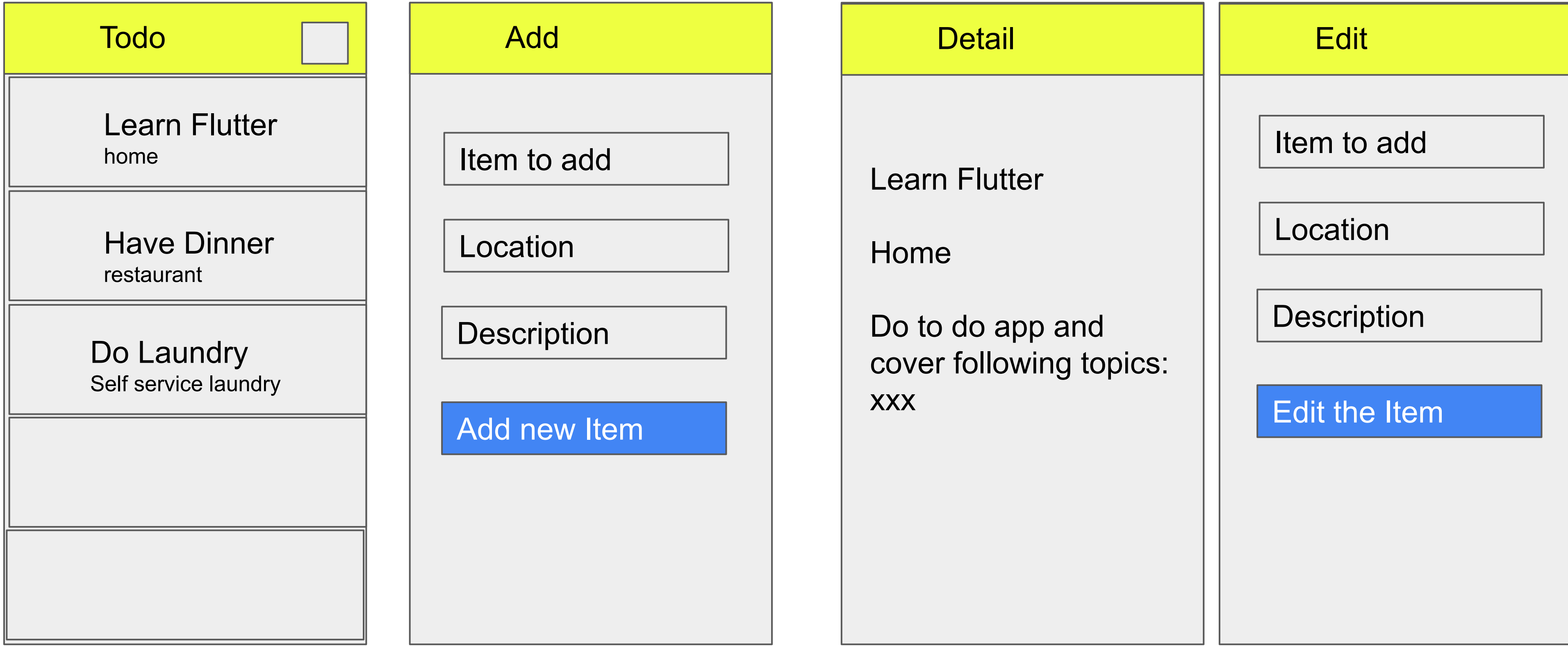


Todo

Learn Flutter
home

Have Dinner
restaurant

Do Laundry
Self service laundry

Add

Item to add

Location

Description

Add new Item

Detail

Learn Flutter

Home

Do to do app and
cover following topics:
xxx

Edit

Item to add

Location

Description

Edit the Item

Topic for today

- Multiple page/ view
- Navigation - how to go from one page to another page
- ListView / known as FlatList in React Native
- List Tile
- Passing data to second page
- Passing data from the second page
- Local Storage / shared preference
-

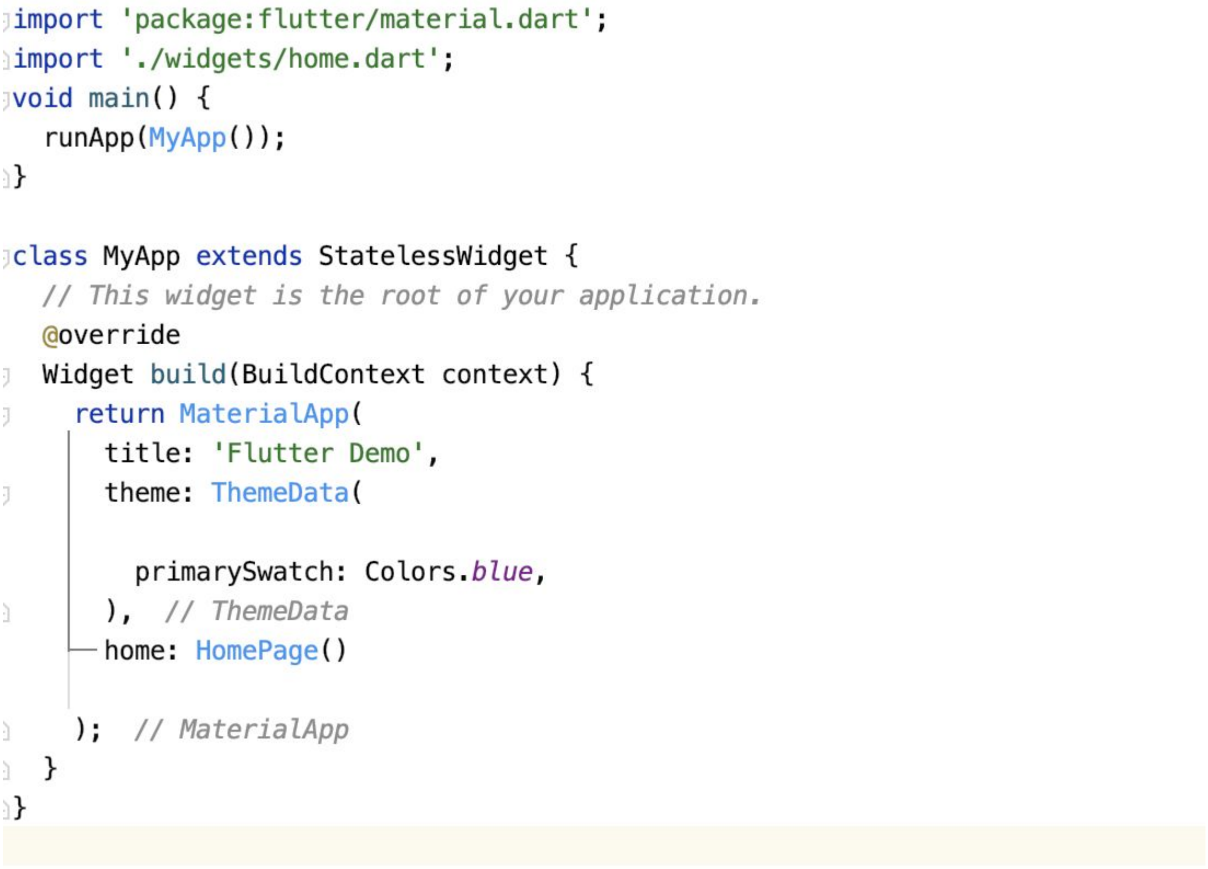
Adding Multiple page

- Create a new directory/package inside **lib** called it widgets
- Add files of all the pages that we want, add.dart, edit.dart, detail.dart, home.dart
- For each of the page do *stless+tab* or *stful + tab*
    - Add.dart - stful + tab - it has a form
    - Edit.dart - stful + tab - ir has a form
    - Home.dart - stful + tab - when the page is loaded, we will get data from ***localstorage***
    - Detail.dart - stless + tab - we only show data passed from the previous page
- ● Get rid of the error by adding import material on top of each files (get from main.dart)
- For each of the page, do a scaffold
- In main.dart you will remove the normal code (boilerplate) but this time around you will import home.dart and call it from ”home”

Stateless vs Stateful

Stateless	Stateful
It’s the page that is only used to show information without having and changing a ***state/ variable***	Page that will show and manipulate information, need to have ***state/variable***
Page show information	API Call Form and you manipulate the data… Getting data from local storage
stless + tab	stful + tab

Main.dart should be as follows



```dart
import 'package:flutter/material.dart';
import './widgets/home.dart';
void main() {
  runApp(MyApp());
}

class MyApp extends StatelessWidget {
  // This widget is the root of your application.
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'Flutter Demo',
      theme: ThemeData(

        primarySwatch: Colors.blue,
      ),  // ThemeData
      home: HomePage()

    );  // MaterialApp
  }
}
```

Navigation to a new page

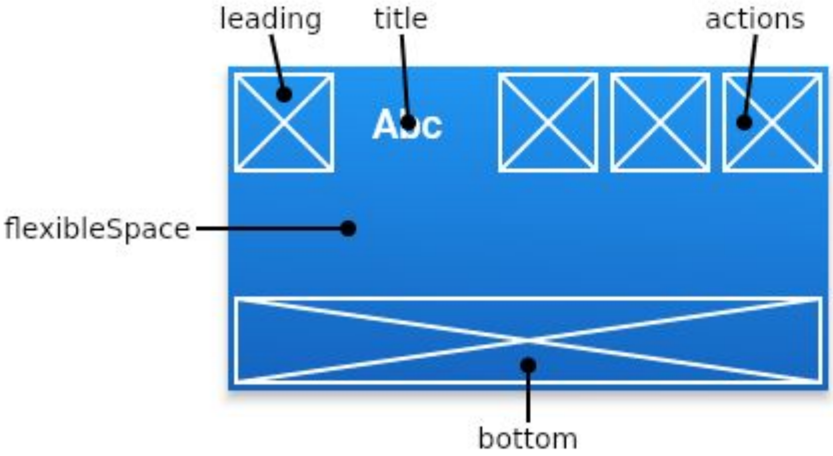
1) Create a button on the AppBar, this button will have an icon “+” on the right (actions), when the button is pressed it will go to Add Page
2) The code to open new page is:

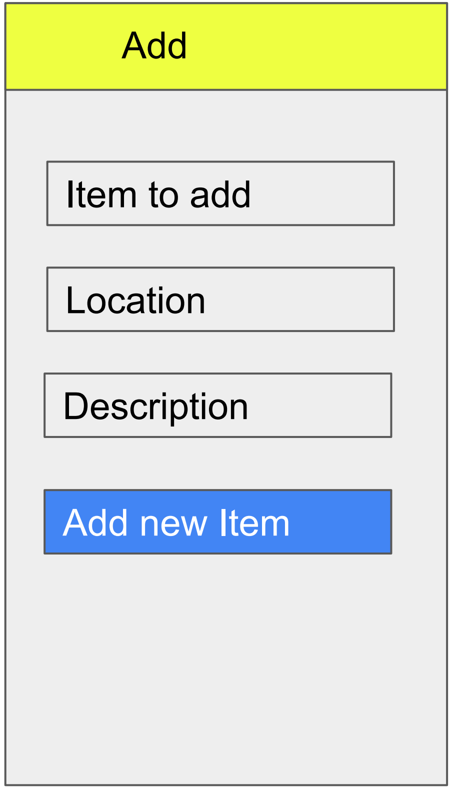
```
Navigator.push(

 context,

 MaterialPageRoute(builder: (context)=>AddPage())

);
```

3) You need to import add on top of the file

Reference:https://docs.flutter.dev/cookbook/navigation/passing-data

Anatomy of an app bar





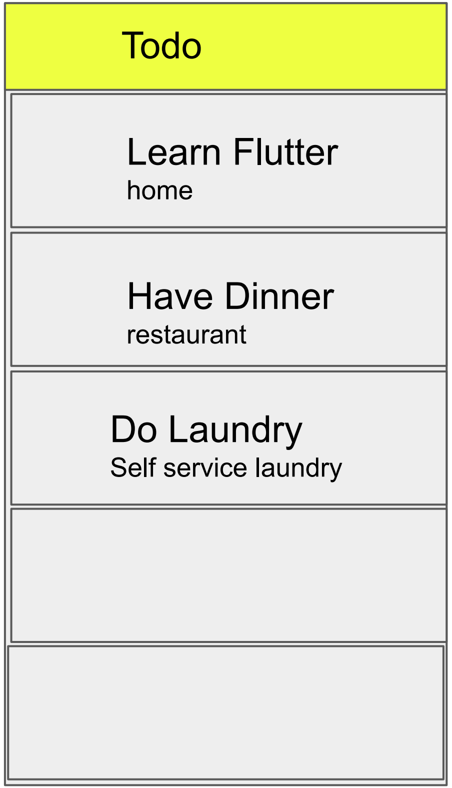
Padding

- Column
  - TextField
  - SizedBox
  - TextField
  - SizedBox
  - TextField
  - SizedBox
  - Button

Revision : Working with TextField

1) Create the TextField widget
2) Create the controller for each of the Textfields
3) Link the Textfield to the controller

ListView



Todo

Learn Flutter
home

Have Dinner
restaurant

Do Laundry
Self service laundry

1) We create mock data = List of object
2) Show the data in ListView
3) Use ListTile to make the list nicer …

ListView

```
ListView.builder(
   itemCount: todos.length,
   itemBuilder: (context,position){
     return Text(todos[position]["name"]!);
   })
```

itemCount - > How many rows are there?

itemBuilder : What to show for each row in the ListView?

https://api.flutter.dev/flutter/widgets/ListView-class.html

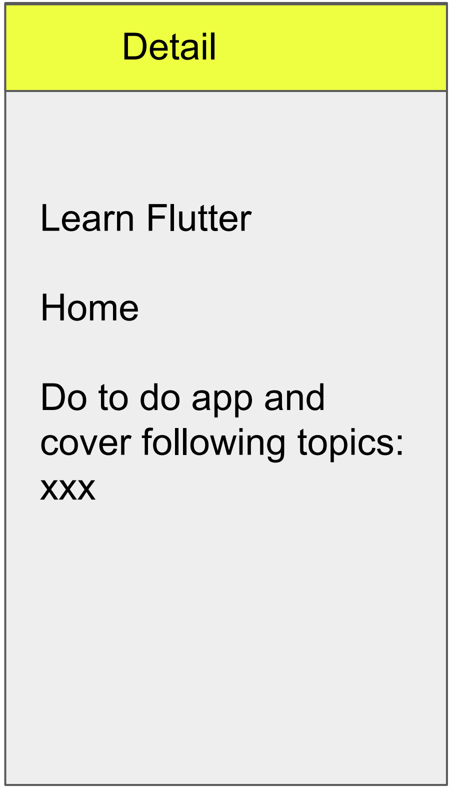
ListTile

ListTile is a widget that helps you to create a row in a ListView nicely

- Title -> title in the ListTile
- subtitle -> Subtitle in the listTile
- leading -> What to show on the right side
- trailing -> What to show on the left side
- onTap -> When the row is pressed what will happens…

https://api.flutter.dev/flutter/material/ListTile-class.html

Detail.dart



Detail

Learn Flutter

Home

Do to do app and cover following topics: xxx

1) Get the data from the home.dart
2) SHow the data on the UI

Column

Text

- Text
- Text

Passing Data to new screen (front to back)

Sender - home.dart

Receiver - detail.dart

1) In receiver, you will create a variable where you are going to store the data and the constructor of the page
2) In sender, you will pass the data as parameter in Navigator.push part…
3) In receiver, create the UI to show the value

Passing data to new screen, back to front

Sender - add.dart

Receiver - home.dart

1) In sender, do Navigator.pop(context, ***data***)
2) In receiver, you will retrieve the data (add async await)

```
IconButton(onPressed: () async {

 final data = await Navigator.push(

   context,

   MaterialPageRoute(builder: (context)=>AddPage())

 );
```

3) then you can manipulate the data

Persistence in programming means the data should always be available upon restart of the application.

Shared Preference

Retrieving the value

```
SharedPreferences prefs = await SharedPreferences.getInstance();
      int counter = (prefs.getInt('counter') ?? 0) + 1;
```

Saving the value

```
    await prefs.setInt('counter', counter);
```

https://pub.dev/packages/shared_preferences

We will load the data in initState, which means after the page is loaded (Home Page)...

```
@override
void initState() {
 // TODO: implement initState
 super.initState();
}
```

We will resave our todos everytome there is a change in the value

After we push it..

Small revision on package

We can add google_fonts inside our project from pub.dev

https://pub.dev/packages/google_fonts

Force unwrap for optional

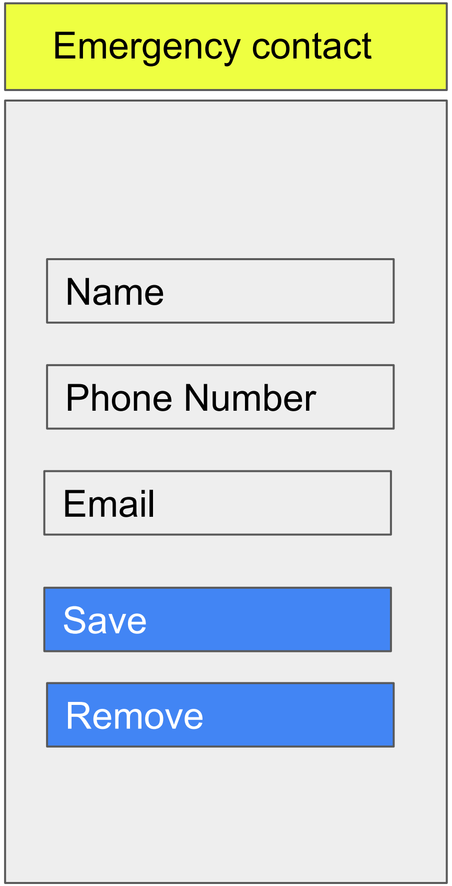
-- remind me to answer this next week

Object / Map

```
var user = {
“name”:”Muzaffar”,
“Age”:30,
“location”:”bangi”
}


print(user[“name”]!)
```

https://github.com/wanmuz86/flutter-todo-jan

Emergency contact app



During the first load, there is no emergency contact the app will show form as shown

-User can enter the emergency contact and press Save. Once pressed show a toast “Info successfully saved”

- Once save the next time app is opened, saved information will be shown

- User can remove the emergency contact by pressing remove. Once pressed show a toast “Info successfully removed”

- Once data is removed the next time app is opened, no information will be shown