

NAME : MAKVANA NEEL

ROLL NO : CE064

ID : 20CEUOS086

BATCH : A4

Lab 9

Tutorial 1:

Here we have created special stateless widget called QuoteCard for printing our quotes in special pattern.

Earlier we have done this using widget in `_EchoListState` class.

```
import 'package:flutter/material.dart';
import 'quote.dart';

void main() => runApp(MaterialApp(
  home: EchoList(),
));

class EchoList extends StatefulWidget {
  const EchoList({Key? key}) : super(key: key);

  @override
  State<EchoList> createState() => _EchoListState();
}
```

```

}

class _EchoListState extends State<EchoList> {
  List<Quote> quotes = [
    Quote(author: 'author1', text: 'I see humans but no
humanity'),
    Quote(text: 'The truth is really pure and never simple',
author: 'author2'),
    Quote(
      text: 'The time is always right to do what is right',
      author: 'author3'),
  ];
  Widget quoteTemplate(quote) {
    return QuoteCard(quote:quote);
  }

  @override
  Widget build(BuildContext context) {
return Scaffold(
  backgroundColor: Colors.lightBlueAccent[100],
  appBar: AppBar(
    title: Text('Quotes'),
    centerTitle: true,
    backgroundColor: Colors.orangeAccent,
  ),
  body: Column(
    // children: quotes.map((quote) =>
quoteTemplate(quote)).toList(),
    children: quotes.map((quote) => QuoteCard(quote:
quote)).toList(),
  ),
);
  }
}

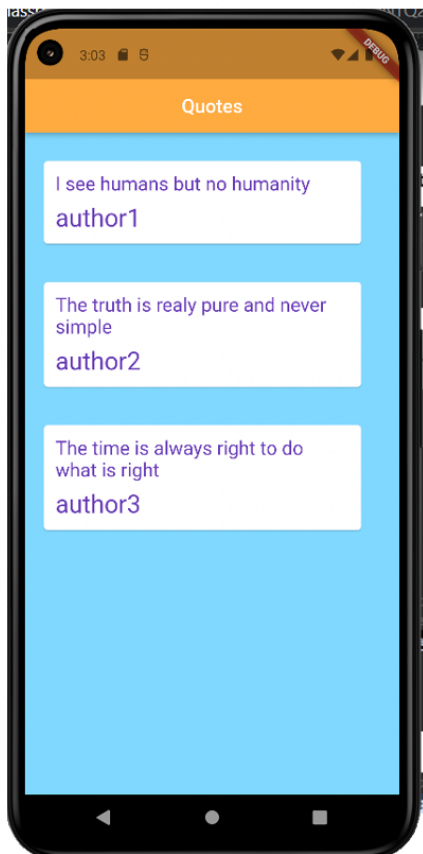
class QuoteCard extends StatelessWidget {
  final Quote quote;
  QuoteCard({required this.quote});

```

```

@override
Widget build(BuildContext context) {
  return Card(
    margin: EdgeInsets.fromLTRB(20.0, 30.0, 40.0, 10.0),
    child: Padding(
      padding: const EdgeInsets.all(12.0),
      child: Column(
        crossAxisAlignment: CrossAxisAlignment.stretch,
        children: [
          Text(
            quote.text,
            style: TextStyle(
              fontSize: 20,
              color: Colors.deepPurple,
            ),
          ),
          SizedBox(height: 10),
          Text(
            quote.author,
            style: TextStyle(
              fontSize: 26,
              color: Colors.deepPurple,
            ),
          ),
        ],
      ),
    ),
  );
}

```



Now we “move” our “QuotCard” widget to another file called “quote_card.dart” so we can use this widget at any place by just “importing” its related file and reusability is also increased.

We have to import “material.dart” and “quote.dart” to newly created file “quote_card.dart” because in QuoteCard class we have property object of Quote class and all widgets are in material.dart so also need to import it.

TextButton widget:

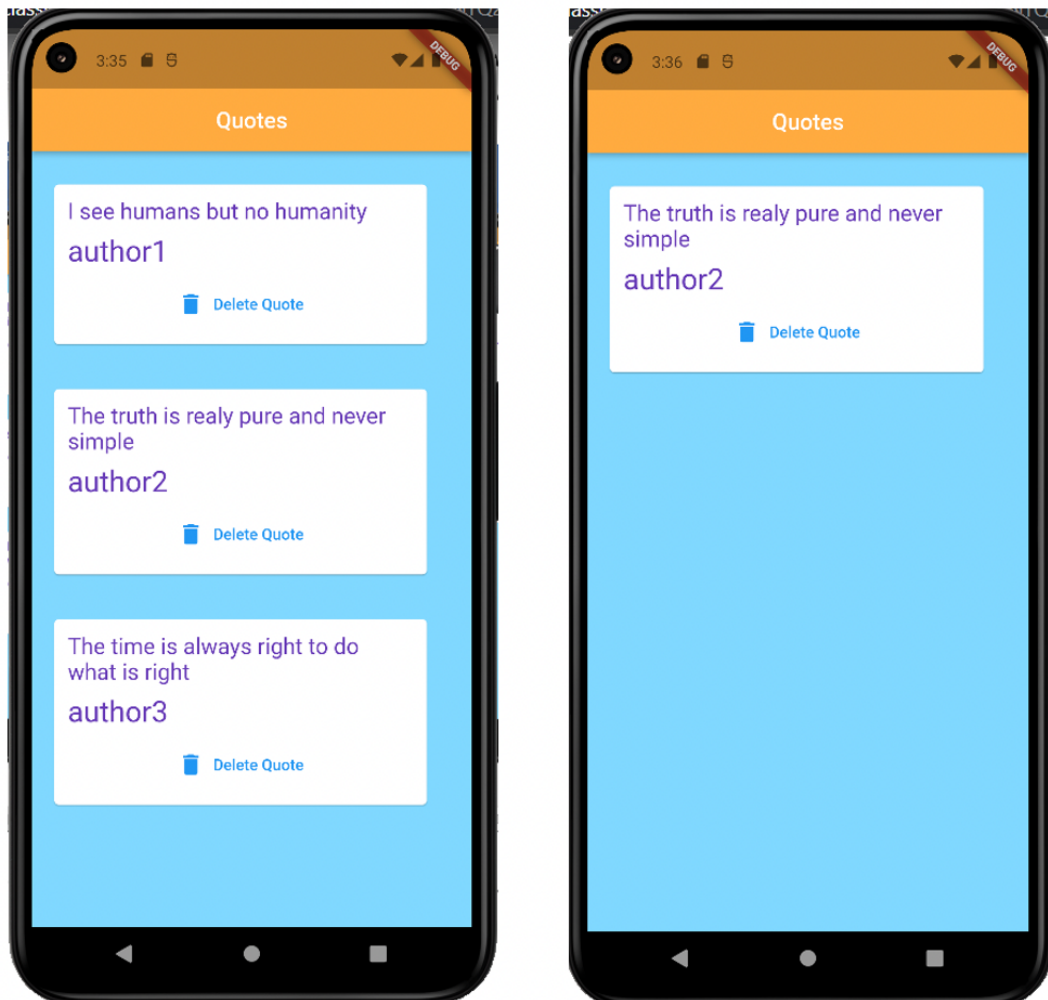
The autofocus and clipBehavior arguments must not be null.

TextButton.icon:

Create a text button from a pair of widgets that serve as the button's icon and label.

The icon and label are arranged in a row and padded by 8 logical pixels at the ends, with an 8 pixel gap in between.

The icon and label arguments must not be null.



Tutorial-2

In main.dart

```
import 'package:flutter/material.dart';
import 'package:lab9_2/pages/home.dart';
void main() => runApp(MaterialApp(
  home: Home(),
));
```

In home.dart:

```
import 'package:flutter/material.dart';
class Home extends StatefulWidget {
  @override
  State<Home> createState() => _HomeState();
}
class _HomeState extends State<Home> {
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      body: SafeArea(child: Text('HOME SCREEN')),
    );
  }
}
```



Route widget:

- The application's top-level routing table.
- When a named route is pushed with `Navigator.pushNamed()`, the route name is looked up in this map. If the name is present, the associated `widgets.WidgetBuilder()` is used to construct a `MaterialPageRoute` that performs an appropriate transition, including Hero animations, to the new route.
- If the app only has one page, then you can specify it using `home` instead.
- If `home` is specified, then it implies an entry in this table for the `Navigator.defaultRouteName()` route (`/`), and it is an error to redundantly provide such a route in the routes table.
- If a route is requested that is not specified in this table (or by `home`), then the `onGenerateRoute()` callback is called to build the page instead.
- The Navigator is only built if routes are provided (either via `home`, `routes`, `onGenerateRoute`, or `onUnknownRoute`); if they are not, builder must not be null.

Main.dart:

```
import 'package:flutter/material.dart';
// import 'package:lab9_2/pages/home.dart';
import './pages/choose_location.dart';
import './pages/home.dart';
import './pages/loading.dart';
```

```

// void main() => runApp(MaterialApp(
//   home: Home(),
// ));
/*
void main() => runApp(MaterialApp(
// home: Home(),
    routes: { // routes is Map variable with 'key:value' pairs
// '/about' ....eg....routes the screen through 'about' widgets
// '/contact'....
    '/': (context) => Loading(), // base routes....base widget
file...main
    file of project
// above statement will creates error...because it conflict with
'home:
    Home(),
// because both statements tell flutter to initialize the app from
their
    given location
    '/home': (context) => Home(),
    '/location': (context) => ChooseLocation(),
    }
));
*/
void main() => runApp(MaterialApp(
// home: Home(),
// instead of making home: property to make any page to initialize a
beginning...
// we can use following code ....
    initialRoute: '/home',
    routes: {

        '/': (context) => Loading(),
        '/home': (context) => Home(),
        '/location': (context) => ChooseLocation(),
    }
));

```


Home.dart:

```
import 'package:flutter/material.dart';
class Home extends StatefulWidget {
  @override
  State<Home> createState() => _HomeState();
}
class _HomeState extends State<Home> {
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      body: SafeArea(
        child: Column(
          children: [
            TextButton.icon(
              onPressed: (){
                Navigator.pushNamed(context, '/location');
              },
              icon: Icon(Icons.edit_location),
              label: Text('EDIT LOCATION'),
            ),
          ],
        ),
      ),
    );
  }
}
```

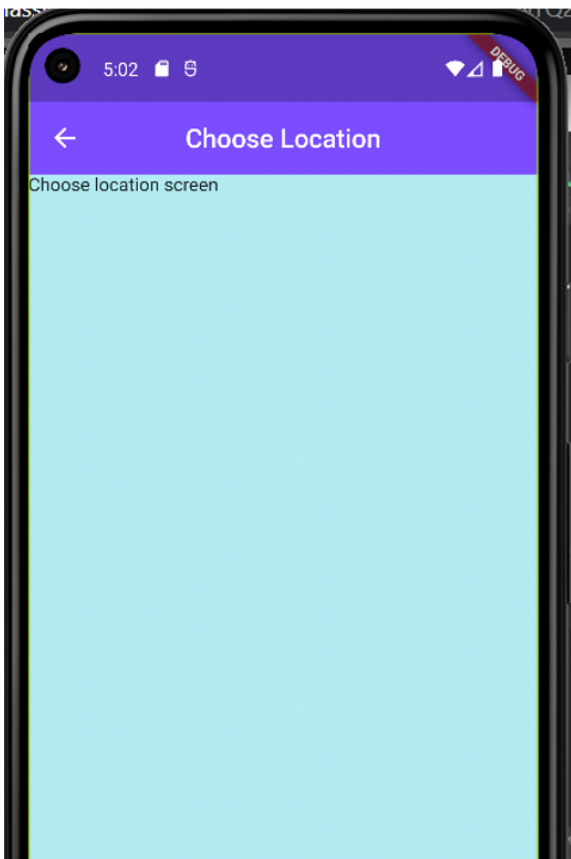
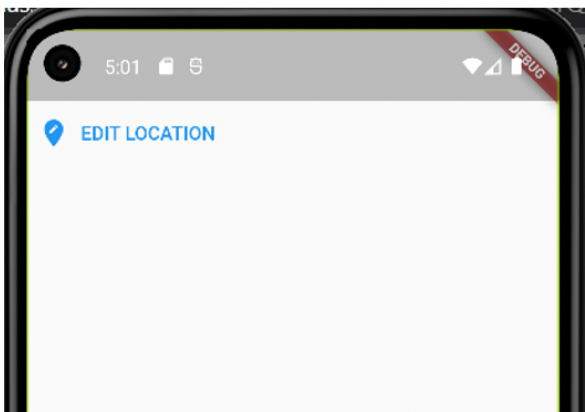
choose_location.dart:

```
import 'package:flutter/material.dart';
class ChooseLocation extends StatefulWidget {
  // const ChooseLocation({Key? key}) : super(key:
  key);
  @override
  State<ChooseLocation> createState() =>
  _ChooseLocationState();
}
class _ChooseLocationState extends
State<ChooseLocation> {
```

```

@override
Widget build(BuildContext context) {
  return Scaffold(
    backgroundColor: Colors.cyan[100],
    appBar: AppBar(
      backgroundColor: Colors.deepPurpleAccent,
      title: Text('Choose Location'),
      centerTitle: true,
      elevation: 0,
    ),
    body: Text('Choose location screen'),
  );
}

```



- The Navigator class provides all the navigation capabilities in a Flutter app.
- Navigator provides methods to mutate the stack by a push to stack or by popping from the stack. The Navigator.push() method is for navigating to a newer page and Navigator.pop() is for going back from the current page.
- Here is a basic example of pop and push: the push method takes BuildContext as the first argument and the second argument is a PageBuilder