LEARN FLUTTER BY DOING

Personal Expense App

In this hands-on exercise you will learn how different concepts in developing mobile application using Flutter. At the end of the each exercise you are expected to submit the following:

a. The Flutter Program
b. The assignment documentations

The Flutter program is the archived Flutter project folder. You can use Zip or WinRar to archive the whole project folder. Assignment documentations are WORD Documents that contains concepts needed in the exercise.

Exercise 1.

1. **Getting Started**

   Create a flutter project named *ExpenseApp;*
   Change the *pubspec.yaml* file download it here;
   Edit the *main.dart* file with the following basic code:

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

Run | Debug | Profile
void main() => runApp(MyApp());

class MyApp extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'Flutter App',
      home: MyHomePage(),
    ); // MaterialApp
  }
}
```

   Add the MyHomePage class at the end of the main.dart file:

```dart
class MyHomePage extends StatelessWidget {

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: Text('Flutter App'),
      ),
      body: Center(
        child: Text('Widget Playground!'),
      ),
    );
  }
}
```

2. **Adding Cards and Containers**

   Before adding Cards and containers to the project you are required to search for the following:

a. *What is Flutter Layout;*
b. *List Ten (10) widgets with description;*
c. *For each widget give a sample implementation;*
d. *In addition, search for the following widgets: Column, Row, Container, and Card. Give a sample implementation for each; and,*
e. *Save the file in WORD File with the filename **Exercise1.docx**.*

Update the main.dart file by encoding the following code. Observe how the application have changed.

```
body: Column(
  children: <Widget>[
    Container(
      width: double.infinity,
      child: Card(
        color: Colors.blue,
        child: Text('CHART!'),
        elevation: 5,
      ), // Card
    ), // Container
    Card(
      child: Text('LIST OF TX'),
    ), // Card
  ], // <Widget>[]
), // Column
```

Take a screenshot of your app. **Paste** it at the end of the ***Exercise1.docx*** file. Caption it with ***Card Implementation***.

Now enhance the App by modifying again the body using the following code:

```
body: Column(
  mainAxisAlignment: MainAxisAlignment.spaceAround,
  crossAxisAlignment: CrossAxisAlignment.stretch,
  children: <Widget>[
    Container(
      width: double.infinity,
      child: Card(
        color: Colors.blue,
        child: Text('CHART!'),
        elevation: 5,
      ), // Card
    ), // Container
    Card(
      color: Colors.red,
      child: Text('LIST OF TX'),
    ), // Card
  ], // <Widget>[]
), // Column
```

Take a screenshot of your app. **Paste** it at the end of the ***Cards and Containers.docx*** file. Caption it with ***Enhanced Card***.

3. Adding Transactions

You are to search on the following: Text Widget, TextField, InputDecoration, DateFormat and TextStyle. Give samples of its implementation. Paste it at the end of the Exer1.docx.

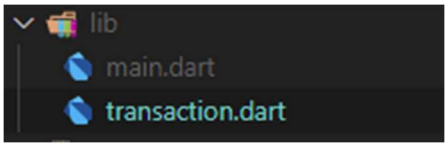
Using the same application, you are now ready to create a transaction class to provide a data structure on the transaction data of the application. Include the following code by creating a separate dart file named ***transaction.dart***

```dart
import 'package:flutter/foundation.dart';

class Transaction {
  final String id;
  final String title;
  final double amount;
  final DateTime date;

  Transaction({
    @required this.id,
    @required this.title,
    @required this.amount,
    @required this.date,
  });
}
```

You should see two (2) files inside your lib folder by now (see below figure).



Now lets implement the transaction class in the main.dart file. First, import it into the main.dart file at the top of the file

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

import './transaction.dart';
```

Then, just below the MyHomePage class declaration add the following code:

```dart
class MyHomePage extends StatelessWidget {
  final List<Transaction> transactions = [
    Transaction(
      id: 't1',
      title: 'New Shoes',
      amount: 69.99,
      date: DateTime.now(),
    ), // Transaction
    Transaction(
      id: 't2',
      title: 'Weekly Groceries',
      amount: 16.53,
      date: DateTime.now(),
    ), // Transaction
  ];
```

Now, let us display the transaction data. Change this section of the body 

with the following code:

Note: Only the highlighted code part

```dart
      Column(
        children: transactions.map((tx) {
          return Card(
            child: Row(
              children: <Widget>[
                Container(
                  margin: EdgeInsets.symmetric(
                    vertical: 10,
                    horizontal: 15,
                  ), // EdgeInsets.symmetric
                  decoration: BoxDecoration(
                    border: Border.all(
                      color: Colors.purple,
                      width: 2,
                    ), // Border.all
                  ), // BoxDecoration
                  padding: EdgeInsets.all(10),
                  child: Text(
                    tx.amount.toString(),
                    style: TextStyle(
                      fontWeight: FontWeight.bold,
                      fontSize: 20,
                      color: Colors.purple,
                    ), // TextStyle
                  ), // Text
                ), // Container
```

Continue below code:

```
            Column(
              crossAxisAlignment: CrossAxisAlignment.start,
              children: <Widget>[
                Text(
                  tx.title,
                  style: TextStyle(
                    fontSize: 16,
                    fontWeight: FontWeight.bold,
                  ), // TextStyle
                ), // Text
                Text(
                  tx.date.toString(),
                  style: TextStyle(
                    color: Colors.grey,
                  ), // TextStyle
                ), // Text
              ], // <Widget>[]
            ), // Column
          ], // <Widget>[]
        ), // Row
      ); // Card
    }).toList(),
  ), // Column
], // <Widget>[]
```

Now, run your application and take a screenshot and add it at the end of the ***Exercise1.docx*** document with the caption ***Transaction Data***.

Further enhance the transaction. Adda the following code:

Note: Only the highlighted code part

```
    children: <Widget>[
      Container(
        width: double.infinity,
        child: Card(
          color: Colors.blue,
          child: Text('CHART!'),
          elevation: 5,
        ), // Card
      ), // Container
      Card(
        elevation: 5,
        child: Container(
          padding: EdgeInsets.all(10),
          child: Column(
            crossAxisAlignment: CrossAxisAlignment.end,
            children: <Widget>[
              TextField(
                decoration: InputDecoration(labelText: 'Title'),
              ), // TextField
              TextField(
                decoration: InputDecoration(labelText: 'Amount'),
              ), // TextField
              FlatButton(
                child: Text('Add Transaction'),
                textColor: Colors.purple,
                onPressed: () {},
              ), // FlatButton
            ], // <Widget>[]
          ), // Column
        ), // Container
      ), // Card
      Column(
        children: transactions.map((tx) {
```

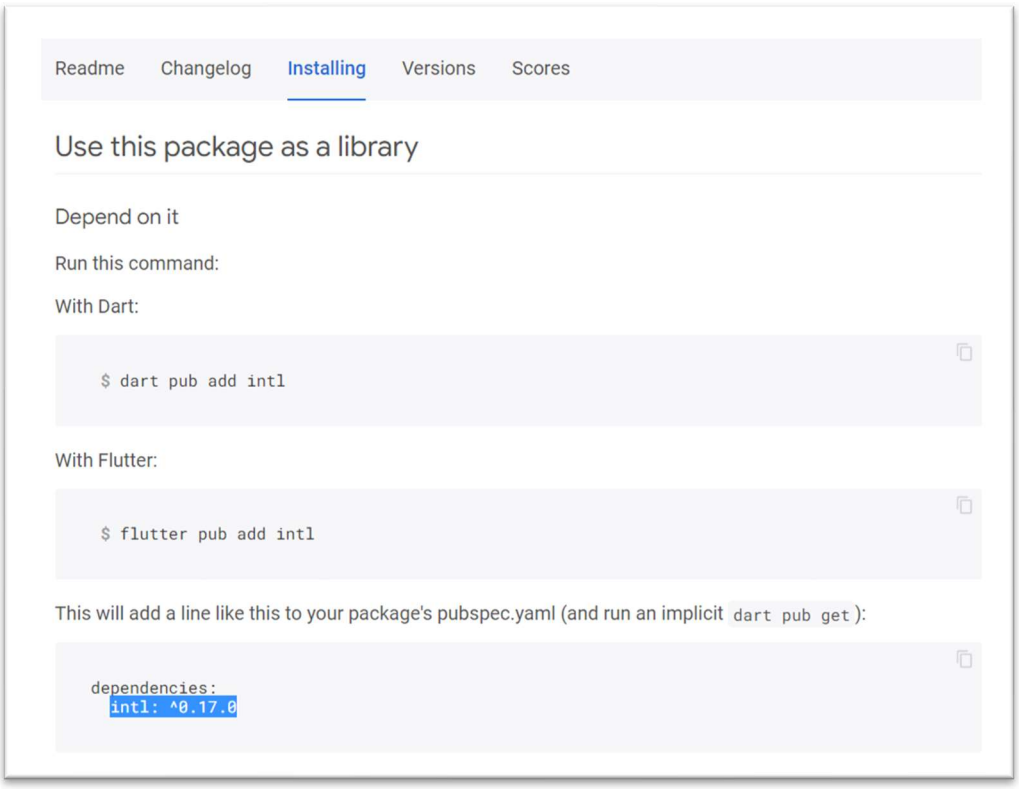
Then add a dollar ($) sign to amount.

```
return Card(
  child: Row(
    children: <Widget>[
      Container(
        margin: EdgeInsets.symmetric(
          vertical: 10,
          horizontal: 15,
        ), // EdgeInsets.symmetric
        decoration: BoxDecoration(
          border: Border.all(
            color: Colors.purple,
            width: 2,
          ), // Border.all
        ), // BoxDecoration
        padding: EdgeInsets.all(10),
        child: Text(
          '\$${tx.amount}',
          style: TextStyle(
            fontWeight: FontWeight.bold,
            fontSize: 20,
            color: Colors.purple,
          ), // TextStyle
```

Next, format the date.

In your browser search ***DART dateformat***. Look for ***intl | Dart Package – Dart Pub*** and open it.

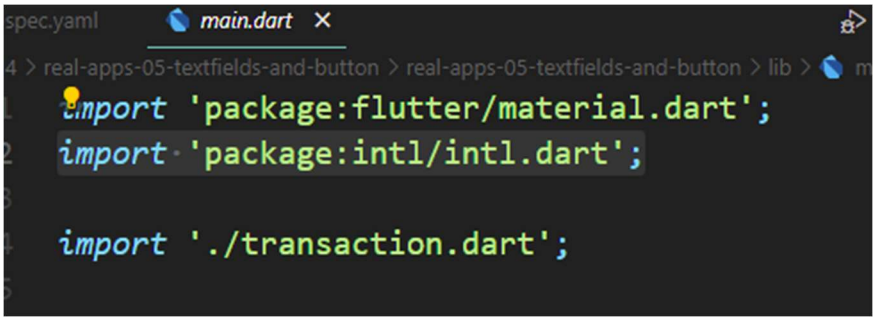
In the installing tab copy the ***dependencies*** text (see highlighted):



In your VSCODE, open the pubspec.yaml file.

Make sure intl is with the same indention like the flutter. Save the yaml file.

Open your main.dart file and import the intl package.



Then finally, format the date.

Now, run your application and take a screenshot and add it at the end of the ***Exercise1.docx*** document with the caption ***Enhanced Transaction Data***.