

**Noman Ali Shah**

**19-NTU-CS-1118**



**Mobile App Development**

**CSE-4078**

**Assignment # 1**

**Q1:**

**Main. Dart file**

import 'package:flutter/material.dart';

import 'story\_brain.dart';

//TODO: Step 15 - Run the app and see if you can see the screen update with the first story. Delete this TODO if it looks as you expected.

void main() => runApp(Destini());

class Destini extends StatelessWidget {

Widget build(BuildContext context) {

return MaterialApp(

theme: ThemeData.dark(),

home: StoryPage(),

);

}

}

//TODO: Step 9 - Create a new storyBrain object from the StoryBrain class.

StoryBrain storyBrain = StoryBrain();

class StoryPage extends StatefulWidget {

\_StoryPageState createState() => \_StoryPageState();

}

class \_StoryPageState extends State<StoryPage> {

@override

Widget build(BuildContext context) {

return Scaffold(

body: Container(

decoration: BoxDecoration(

image: DecorationImage(

image: AssetImage('images/background.png'),

fit: BoxFit.cover,

),

),

padding: EdgeInsets.symmetric(vertical: 50.0, horizontal: 15.0),

constraints: BoxConstraints.expand(),

child: SafeArea(

child: Column(

crossAxisAlignment: CrossAxisAlignment.stretch,

children: <Widget>[

Expanded(

flex: 12,

child: Center(

child: Text(

//TODO: Step 10 - use the storyBrain to get the first story title and display it in this Text Widget.

storyBrain.getStory(),

style: TextStyle(

fontSize: 25.0,

),

),

),

),

Expanded(

flex: 2,

child: FlatButton(

onPressed: () {

//Choice 1 made by user.

//TODO: Step 18 - Call the nextStory() method from storyBrain and pass the number 1 as the choice made by the user.

//TODO: Step 24 - Run the app and try to figure out what code you need to add to this file to make the story change when you press on the choice buttons.

setState(() {

storyBrain.nextStory(1);

});

},

color: Colors.red,

child: Text(

//TODO: Step 13 - Use the storyBrain to get the text for choice 1.

storyBrain.getChoice1(),

style: TextStyle(

fontSize: 20.0,

),

),

),

),

SizedBox(

height: 20.0,

),

Expanded(

flex: 2,

//TODO: Step 26 - Use a Flutter Visibility Widget to wrap this FlatButton.

//TODO: Step 28 - Set the "visible" property of the Visibility Widget to equal the output from the buttonShouldBeVisible() method in the storyBrain.

child: Visibility(

visible: storyBrain.buttonShouldBeVisible(),

child: FlatButton(

onPressed: () {

//Choice 2 made by user.

//TODO: Step 19 - Call the nextStory() method from storyBrain and pass the number 2 as the choice made by the user.

setState(() {

storyBrain.nextStory(2);

});

},

color: Colors.blue,

child: Text(

//TODO: Step 14 - Use the storyBrain to get the text for choice 2.

storyBrain.getChoice2(),

style: TextStyle(

fontSize: 20.0,

),

),

),

),

),

],

),

),

),

);

}

}

**StoryBrain. Dart file**

}

//TODO: Step 6 - import the story.dart file into this file.

import 'story.dart';

//TODO: Step 5 - Create a new class called StoryBrain.

class StoryBrain {

//TODO: Step 7 - Uncomment the lines below to include storyData as a private property in StoryBrain. Hint: You might need to change something in story.dart to make this work.

  List<Story> \_storyData = [

    Story(

        storyTitle:

            'Story 0: Your car has blown a tire on a winding road in the middle of nowhere with no cell phone reception. You decide to hitchhike. A rusty pickup truck rumbles to a stop next to you. A man with a wide brimmed hat with soulless eyes opens the passenger door for you and asks: "Need a ride, boy?".',

        choice1: 'Will take you to story 2',

        choice2: 'Will take you to story 1.'),

    Story(

        storyTitle: 'Story 1: He nods slowly, unphased by the question.',

        choice1: 'Will take you to story 2',

        choice2: 'Will take you to story 3'),

    Story(

        storyTitle:

            'Story 2: As you begin to drive, the stranger starts talking about his relationship with his mother. He gets angrier and angrier by the minute. He asks you to open the glovebox. Inside you find a bloody knife, two severed fingers, and a cassette tape of Elton John. He reaches for the glove box.',

        choice1: 'Will take you to story 5',

        choice2: 'Will take you to story 4'),

    Story(

        storyTitle:

            'Story 3: What? Such a cop out! Did you know traffic accidents are the second leading cause of accidental death for most adult age groups?',

        choice1: 'Restart',

        choice2: ''),

    Story(

        storyTitle:

            'Story 4: As you smash through the guardrail and careen towards the jagged rocks below you reflect on the dubious wisdom of stabbing someone while they are driving a car you are in.',

        choice1: 'Restart',

        choice2: ''),

    Story(

        storyTitle:

            'Story 5: You bond with the murderer while crooning verses of "Can you feel the love tonight". He drops you off at the next town. Before you go he asks you if you know any good places to dump bodies. You reply: "Try the pier".',

        choice1: 'Restart',

        choice2: '')

  ];

  //TODO: Step 23 - Use the storyNumber property inside getStory(), getChoice1() and getChoice2() so that it gets the updated story and choices rather than always just the first (0th) one.

//TODO: Step 8 - Create a method called getStory() that returns the first storyTitle from \_storyData.

  String getStory() {

    return \_storyData[\_storyNumber].storyTitle;

  }

//TODO: Step 11 - Create a method called getChoice1() that returns the text for the first choice1 from \_storyData.

  String getChoice1() {

    return \_storyData[\_storyNumber].choice1;

  }

//TODO: Step 12 - Create a method called getChoice2() that returns the text for the first choice2 from \_storyData.

  String getChoice2() {

    return \_storyData[\_storyNumber].choice2;

  }

//TODO: Step 16 - Create a property called storyNumber which starts with a value of 0. This will be used to track which story the user is currently viewing.

  //TODO: Step 25 - Change the storyNumber property into a private property so that only story\_brain.dart has access to it. You can do this by right clicking on the name (storyNumber) and selecting Refactor -> Rename to make the change across all the places where it's used.

  int \_storyNumber = 0;

//TODO: Step 17 - Create a method called nextStory(), it should not have any outputs but it should have 1 input called choiceNumber which will be the choice number (int) made by the user.

  void nextStory(int choiceNumber) {

    //TODO: Step 21 - Using the story plan, update nextStory to change the storyNumber depending on the choice made by the user.

    //When user is on story0 and they chose choice1, the story should progress to story2.

    if (choiceNumber == 1 && \_storyNumber == 0) {

      \_storyNumber = 2;

    } else if (choiceNumber == 2 && \_storyNumber == 0) {

      \_storyNumber = 1;

    } else if (choiceNumber == 1 && \_storyNumber == 1) {

      \_storyNumber = 2;

    } else if (choiceNumber == 2 && \_storyNumber == 1) {

      \_storyNumber = 3;

    } else if (choiceNumber == 1 && \_storyNumber == 2) {

      \_storyNumber = 5;

    } else if (choiceNumber == 2 && \_storyNumber == 2) {

      \_storyNumber = 4;

    }

    //TODO: Step 22 - In nextStory() if the storyNumber is equal to 3 or 4 or 5, that means it's the end of the game and it should call a method called restart() that resets the storyNumber to 0.

    else if (\_storyNumber == 3 || \_storyNumber == 4 || \_storyNumber == 5) {

      restart();

    }

  }

  void restart() {

    \_storyNumber = 0;

  }

//TODO: Step 20 - Download the story plan here: https://drive.google.com/uc?export=download&id=1KU6EghkO9Hf2hRM0756xFHgNaZyGCou3

//TODO: Step 27 - Create a method called buttonShouldBeVisible() which checks to see if storyNumber is 0 or 1 or 2 (when both buttons should show choices) and return true if that is the case, else it should return false.

  bool buttonShouldBeVisible() {

    //You could also just check if (\_storyNumber < 3)

    if (\_storyNumber == 0 || \_storyNumber == 1 || \_storyNumber == 2) {

      return true;

    } else {

      return false;

    }

  }

}

**Story. Dart file**

//TODO: Step 2 - Create a new class called Story.

class Story {

  //TODO: Step 3 - Create 3 properties for this class, A. storyTitle to store the story text. B. choice1 to store the text for choice 1, C. choice2 to store the text for choice 2.

  String storyTitle = '';

  String choice1 = '';

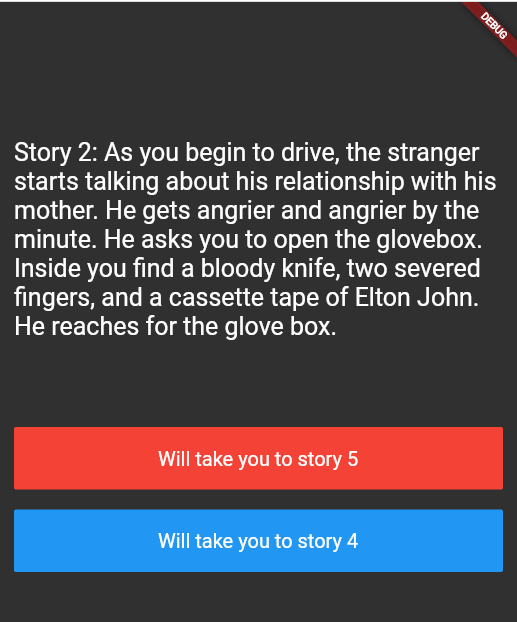
  String choice2 = '';

  //TODO: Step 4 - Create a Constructor for this class to be able to initialise the properties created in step 3.

  Story(

      {required this.storyTitle, required this.choice1, required this.choice2});

}



**Q2:**

**Main. Dart file**

import 'package:flutter/material.dart';

import 'homePage.dart';

import 'resultPage.dart';

void main() {

  runApp(const MyApp());

}

class MyApp extends StatelessWidget {

  const MyApp({Key? key}) : super(key: key);

  @override

  Widget build(BuildContext context) {

    return MaterialApp(

      debugShowCheckedModeBanner: false,

      theme: ThemeData.dark().copyWith(

        primaryColor: Color(0xFF0A0E21),

        scaffoldBackgroundColor: Color(0xFF0A0E21),

      ),

      title: "BMI Calculator",

      home: HomePage(),

    );

  }

}

**Resultpage.dart file**

import 'bottomButton.dart';

import 'calculatorBrain.dart';

import 'cardChild.dart';

import 'constants.dart';

import 'homePage.dart';

import 'reuseablecard.dart';

import 'package:flutter/material.dart';

class ResultPage extends StatelessWidget {

  //const ResultPage({ Key? key }) : super(key: key);

  ResultPage(this.resultText, this.interpretation, this.bmiResult);

  String resultText;

  String interpretation;

  String bmiResult;

  @override

  Widget build(BuildContext context) {

    return Scaffold(

      appBar: AppBar(

        title: Text('BMI Calculator'),

        backgroundColor: Color(0xFF0A0E21),

      ),

      body: Column(

        crossAxisAlignment: CrossAxisAlignment.stretch,

        children: [

          Text(

            "Your Result",

            style: TextStyle(fontSize: 50, color: Colors.white),

          ),

          Expanded(

              child: ReusableCard(

            colour: kActiveColor,

            cardChild: Column(

              mainAxisAlignment: MainAxisAlignment.spaceEvenly,

              crossAxisAlignment: CrossAxisAlignment.center,

              children: [

                Text(

                  resultText.toUpperCase(),

                  style: TextStyle(color: Colors.green, fontSize: 16),

                ),

                Text(

                  bmiResult,

                  style: TextStyle(fontSize: 36),

                ),

                Text(interpretation, style: TextStyle(fontSize: 16)),

              ],

            ),

          )),

          GestureDetector(

            onTap: (() {

              Navigator.pop(context);

            }),

            child: BottomButton(

              text: 'RE-CALCULATE YOUR BMI',

            ),

          )

        ],

      ),

    );

  }

}

**ResuableCard. Dart file**

import 'package:flutter/material.dart';

class ReusableCard extends StatelessWidget {

  final Color? colour;

  final Widget? cardChild;

  ReusableCard({

    @required this.colour,

    this.cardChild,

  });

  @override

  Widget build(BuildContext context) {

    return Container(

      child: this.cardChild,

      margin: EdgeInsets.all(15.0),

      decoration: BoxDecoration(

        borderRadius: BorderRadius.circular(10),

        color: this.colour,

      ),

    );

  }

}

**ResuableCard. Dart file**

**HomePage. Dart file**

import 'package:flutter/material.dart';

import 'constants.dart';

import 'cardChild.dart';

import 'reuseablecard.dart';

import 'resultPage.dart';

import 'calculatorBrain.dart';

import 'package:font\_awesome\_flutter/font\_awesome\_flutter.dart';

import 'bottomButton.dart';

enum Gender { Male, Female, none }

class HomePage extends StatefulWidget {

  const HomePage({Key? key}) : super(key: key);

  @override

  State<HomePage> createState() => \_HomePageState();

}

class \_HomePageState extends State<HomePage> {

  Gender selectedGender = Gender.none;

  int height = 180;

  int weight = 70;

  int age = 20;

  @override

  Widget build(BuildContext context) {

    return Scaffold(

      appBar: AppBar(

        title: Text("BMI Calculator"),

        backgroundColor: Color(0xFF0A0E21),

      ),

      body: Column(

        crossAxisAlignment: CrossAxisAlignment.stretch,

        children: [

          Expanded(

            child: Row(children: [

              Expanded(

                child: GestureDetector(

                  onTap: (() {

                    setState(() {

                      selectedGender = Gender.Male;

                    });

                  }),

                  child: ReusableCard(

                    colour: selectedGender == Gender.Male

                        ? kActiveColor

                        : kInactiveColor,

                    cardChild: CardChild(

                      icon: FontAwesomeIcons.mars,

                      label: 'Male',

                    ),

                  ),

                ),

              ),

              Expanded(

                child: GestureDetector(

                  onTap: (() {

                    setState(() {

                      selectedGender = Gender.Female;

                    });

                  }),

                  child: ReusableCard(

                    colour: selectedGender == Gender.Female

                        ? kActiveColor

                        : kInactiveColor,

                    cardChild: CardChild(

                      icon: FontAwesomeIcons.venus,

                      label: 'Female',

                    ),

                  ),

                ),

              ),

            ]),

          ),

          Expanded(

            child: Row(

              children: [

                Expanded(

                  child: ReusableCard(

                    colour: kInactiveColor,

                    cardChild: Column(

                      mainAxisAlignment: MainAxisAlignment.center,

                      children: [

                        Text(

                          "Height",

                          style: kLabelTextStyle,

                        ),

                        Row(

                          mainAxisAlignment: MainAxisAlignment.center,

                          crossAxisAlignment: CrossAxisAlignment.baseline,

                          textBaseline: TextBaseline.alphabetic,

                          children: [

                            Text(

                              height.toString(),

                              style: TextStyle(

                                fontSize: 50,

                              ),

                            ),

                            Text(

                              'cm',

                              style: TextStyle(

                                fontSize: 20,

                              ),

                            ),

                          ],

                        ),

                        SliderTheme(

                          data: SliderTheme.of(context).copyWith(

                            thumbColor: Colors.pink,

                            activeTrackColor: Colors.pink,

                            inactiveTrackColor: Colors.grey,

                            thumbShape:

                                RoundSliderThumbShape(enabledThumbRadius: 8),

                            overlayShape:

                                RoundSliderOverlayShape(overlayRadius: 15),

                            overlayColor: Color.fromARGB(255, 243, 142, 218),

                          ),

                          child: Slider(

                            min: 120,

                            max: 220,

                            value: height.toDouble(),

                            onChanged: (number) {

                              setState(() {

                                height = number.round();

                              });

                            },

                          ),

                        ),

                      ],

                    ),

                  ),

                ),

              ],

            ),

          ),

          Expanded(

            child: Row(

              children: [

                Expanded(

                  child: ReusableCard(

                    colour: kInactiveColor,

                    cardChild: Column(

                      mainAxisAlignment: MainAxisAlignment.center,

                      children: [

                        Text(

                          'Weight',

                          style: kLabelTextStyle,

                        ),

                        Row(

                          mainAxisAlignment: MainAxisAlignment.center,

                          crossAxisAlignment: CrossAxisAlignment.baseline,

                          textBaseline: TextBaseline.alphabetic,

                          children: [

                            Text(

                              weight.toString(),

                              style: TextStyle(fontSize: 50),

                            ),

                          ],

                        ),

                        Row(

                          mainAxisAlignment: MainAxisAlignment.spaceEvenly,

                          children: [

                            GestureDetector(

                              onTap: () {

                                setState(() {

                                  weight--;

                                });

                              },

                              child: CircleAvatar(

                                backgroundColor:

                                    Color.fromARGB(255, 57, 58, 90),

                                child: Icon(

                                  FontAwesomeIcons.minus,

                                  color: Colors.white,

                                ),

                              ),

                            ),

                            GestureDetector(

                              onTap: () {

                                setState(() {

                                  weight++;

                                });

                              },

                              child: CircleAvatar(

                                backgroundColor:

                                    Color.fromARGB(255, 57, 58, 90),

                                child: Icon(

                                  FontAwesomeIcons.plus,

                                  color: Colors.white,

                                ),

                              ),

                            ),

                          ],

                        ),

                      ],

                    ),

                  ),

                ),

                Expanded(

                  child: ReusableCard(

                    colour: kInactiveColor,

                    cardChild: Column(

                      mainAxisAlignment: MainAxisAlignment.center,

                      children: [

                        Text(

                          'Age',

                          style: kLabelTextStyle,

                        ),

                        Row(

                          mainAxisAlignment: MainAxisAlignment.center,

                          crossAxisAlignment: CrossAxisAlignment.baseline,

                          textBaseline: TextBaseline.alphabetic,

                          children: [

                            Text(

                              age.toString(),

                              style: TextStyle(fontSize: 50),

                            ),

                          ],

                        ),

                        Row(

                          mainAxisAlignment: MainAxisAlignment.spaceEvenly,

                          children: [

                            GestureDetector(

                              onTap: () {

                                setState(() {

                                  age--;

                                });

                              },

                              child: CircleAvatar(

                                backgroundColor:

                                    Color.fromARGB(255, 57, 58, 90),

                                child: Icon(

                                  FontAwesomeIcons.minus,

                                  color: Colors.white,

                                ),

                              ),

                            ),

                            GestureDetector(

                              onTap: () {

                                setState(() {

                                  age++;

                                });

                              },

                              child: CircleAvatar(

                                backgroundColor:

                                    Color.fromARGB(255, 57, 58, 90),

                                //radius: 30,

                                child: Icon(

                                  FontAwesomeIcons.plus,

                                  color: Colors.white,

                                ),

                              ),

                            ),

                          ],

                        ),

                      ],

                    ),

                  ),

                ),

              ],

            ),

          ),

          GestureDetector(

            onTap: () {

              CalculatorBrain calc = CalculatorBrain(height, weight);

              Navigator.push(

                  context,

                  MaterialPageRoute(

                      builder: (context) => ResultPage(calc.getResult(),

                          calc.getInterpretation(), calc.calculateBMI())));

            },

            child: BottomButton(

              text: 'CALCULATE',

            ),

          )

        ],

      ),

    );

  }

}

**Constants. Dart file**

import 'package:flutter/material.dart';

const kLabelTextStyle = TextStyle(

  fontSize: 16.0,

  color: Color(0xFF8D8E98),

);

const kActiveColor = Color(0xFF1D1E33);

const kInactiveColor = Color(0xFF111328);

**CardChild. Dart file**

import 'package:flutter/material.dart';

import 'constants.dart';

const double iconSize = 90.0;

const sizedBoxHeight = 15.0;

class CardChild extends StatelessWidget {

  final IconData? icon;

  final String label;

  CardChild({@required this.icon, this.label = ''});

  @override

  Widget build(BuildContext context) {

    return Column(

      mainAxisAlignment: MainAxisAlignment.center,

      children: <Widget>[

        Icon(

          this.icon,

          size: iconSize,

        ),

        SizedBox(height: sizedBoxHeight),

        Text(

          this.label,

          style: kLabelTextStyle,

        ),

      ],

    );

  }

}

**CalculatorBrain. Dart file**

import 'dart:math';

import 'main.dart';

class CalculatorBrain {

  final int height;

  final int weight;

  double \_bmi = 0;

  CalculatorBrain(this.height, this.weight);

  String calculateBMI() {

    \_bmi = weight / pow(height / 100, 2);

    return \_bmi.toStringAsFixed(1);

  }

  String getResult() {

    if (\_bmi >= 25) {

      return 'OVERWEIGHT';

    } else if (\_bmi > 18.5) {

      return 'NORMAL';

    } else {

      return 'UNDERWEIGHT';

    }

  }

  String getInterpretation() {

    if (\_bmi >= 25) {

      return 'You have a higher than normal body weight. Try to exercise more!';

    } else if (\_bmi > 18.5) {

      return 'You have a normal body weight. Good job!';

    } else {

      return 'You have a lower than normal body weight. You should eat more!';

    }

  }

}

**BottomButton. Dart file**

import 'package:flutter/material.dart';

import 'constants.dart';

class BottomButton extends StatelessWidget {

  const BottomButton({this.text});

  final String? text;

  @override

  Widget build(BuildContext context) {

    return Container(

      decoration: BoxDecoration(

          color: Colors.pink, borderRadius: BorderRadius.circular(25)),

      height: 40,

      width: double.infinity,

      child: Center(child: Text('$text')),

    );

  }

}

