Logo, company name

Description automatically generated

**NAME**  USAMA HAIDER

**REG NO** 19-NTU-CS-1129

**ASSIGNMENT TITLE**  Assignment # 1

**SUBMIT DATE** 27-04-2022

**SUBMIT TO** MA’AM AYESHA

MA’AM SARA

Sir Kamran

**Q1:**

**Code:**

**Story.dart:**

class Story {

  String storyTitle;

  String choice1;

  String choice2;

  Story(

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

}

**Story\_brain.dart:**

import 'story.dart';

class StoryBrain {

  List<Story> \_storyData = [

    Story(

        storyTitle:

            '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: 'I\'ll hop in. Thanks for the help!',

        choice2: 'Better ask him if he\'s a murderer first.'),

    Story(

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

        choice1: 'At least he\'s honest. I\'ll climb in.',

        choice2: 'Wait, I know how to change a tire.'),

    Story(

        storyTitle:

            '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: 'I love Elton John! Hand him the cassette tape.',

        choice2: 'It\'s him or me! You take the knife and stab him.'),

    Story(

        storyTitle:

            '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:

            '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:

            '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: '')

  ];

  String getStory() {

    return \_storyData[\_storyNumber].storyTitle;

  }

  String getChoice1() {

    return \_storyData[\_storyNumber].choice1;

  }

  String getChoice2() {

    return \_storyData[\_storyNumber].choice2;

  }

  int \_storyNumber = 0;

  void nextStory(int choiceNumber) {

    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;

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

      restart();

    }

  }

  void restart() {

    \_storyNumber = 0;

  }

  bool buttonShouldBeVisible() {

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

      return true;

    } else {

      return false;

    }

  }

}

**Main.dart:**

import 'package:flutter/material.dart';

import 'story\_brain.dart';

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

class Destini extends StatelessWidget {

  Widget build(BuildContext context) {

    return MaterialApp(

      theme: ThemeData.dark(),

      home: StoryPage(),

    );

  }

}

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/1.jpg'),

            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(

                    storyBrain.getStory(),

                    style: TextStyle(

                      fontSize: 25.0,

                    ),

                  ),

                ),

              ),

              Expanded(

                flex: 2,

                child: FlatButton(

                  onPressed: () {

                    setState(() {

                      storyBrain.nextStory(1);

                    });

                  },

                  color: Colors.red,

                  child: Text(

                    storyBrain.getChoice1(),

                    style: TextStyle(

                      fontSize: 20.0,

                    ),

                  ),

                ),

              ),

              SizedBox(

                height: 20.0,

              ),

              Expanded(

                flex: 2,

                child: Visibility(

                  visible: storyBrain.buttonShouldBeVisible(),

                  child: FlatButton(

                    onPressed: () {

                      setState(() {

                        storyBrain.nextStory(2);

                      });

                    },

                    color: Colors.blue,

                    child: Text(

                      storyBrain.getChoice2(),

                      style: TextStyle(

                        fontSize: 20.0,

                      ),

                    ),

                  ),

                ),

              ),

            ],

          ),

        ),

      ),

    );

  }

}

**Output:**

A screenshot of a computer

Description automatically generated

Graphical user interface, text

Description automatically generated

A screenshot of a computer

Description automatically generated

**Q2:**

**Code:**

**Bmi\_brain:**

import 'dart:math';

class bmiBrain {

  bmiBrain({required this.height, required this.weight});

  final int height;

  final int weight;

  // double \_bmi;

  String calculateBMI() {

    double bmi = weight / pow(height / 100, 2);

    return bmi.toStringAsFixed(2);

  }

  String getResult() {

    double bmi = weight / pow(height / 100, 2);

    if (bmi >= 25) {

      return 'Overweight';

    } else if (bmi > 18.5) {

      return 'Normal';

    } else {

      return 'Underweight';

    }

  }

  String getInterpretation() {

    double bmi = weight / pow(height / 100, 2);

    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 can eat a bit more.';

    }

  }

}

**Calculate\_button:**

import 'package:flutter/material.dart';

import 'textstyle.dart';

class calculatebutton extends StatelessWidget {

  calculatebutton({required this.onTap, required this.buttonTitle});

  final Function() onTap;

  final String buttonTitle;

  @override

  Widget build(BuildContext context) {

    return GestureDetector(

      onTap: onTap,

      child: Container(

        child: Center(

          child: Text(

            buttonTitle,

            style: kLargeButtonTextStyle,

          ),

        ),

        color: kBottomContainerColour,

        margin: EdgeInsets.only(top: 10.0),

        width: double.infinity,

        height: kBbottomContainerHeight,

      ),

    );

  }

}

**Child\_i\_c:**

import 'package:flutter/material.dart';

class child\_i\_c extends StatelessWidget {

  child\_i\_c({required this.colour, this.cardChild, this.onPress});

  final Color colour;

  final Widget? cardChild;

  final Function()? onPress;

  @override

  Widget build(BuildContext context) {

    return GestureDetector(

      onTap: onPress,

      child: Container(

        child: cardChild,

        margin: EdgeInsets.all(10.0),

        decoration: BoxDecoration(

          color: colour,

          borderRadius: BorderRadius.circular(7.0),

        ),

      ),

    );

  }

}

**Child\_w\_h:**

import 'package:flutter/material.dart';

class child\_w\_h extends StatelessWidget {

  child\_w\_h({required this.icon, required this.onPressed});

  final IconData icon;

  final Function() onPressed;

  @override

  Widget build(BuildContext context) {

    return RawMaterialButton(

      child: Icon(icon),

      elevation: 3.0,

      constraints: BoxConstraints.tightFor(

        width: 20.0,

        height: 20.0,

      ),

      shape: CircleBorder(),

      fillColor: Color(0xFF4C4F5E),

      onPressed: onPressed,

    );

  }

}

**First screen:**

import 'package:flutter/cupertino.dart';

import 'package:flutter/material.dart';

import 'bmi\_brain.dart';

import 'child\_w\_h.dart';

import 'child\_i\_c.dart';

import 'icon.dart';

import 'textstyle.dart';

import 'second\_screen.dart';

import 'calculate\_button.dart';

enum Gender {

  male,

  female,

}

class InputPage extends StatefulWidget {

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

  @override

  \_InputPageState createState() => \_InputPageState();

}

class \_InputPageState extends State<InputPage> {

  Gender? selectedGender;

  int height = 180;

  int weight = 60;

  int age = 20;

  @override

  Widget build(BuildContext context) {

    return Scaffold(

      appBar: AppBar(

        backgroundColor: Color(0xFF1D1E33),

        title: Text('BMI Calculator'),

      ),

      body: Column(

        crossAxisAlignment: CrossAxisAlignment.stretch,

        children: <Widget>[

          Expanded(

            child: Row(

              children: <Widget>[

                Expanded(

                  child: child\_i\_c(

                    onPress: () {

                      setState(() {

                        selectedGender = Gender.male;

                      });

                    },

                    colour: selectedGender == Gender.male

                        ? kActiveCardColour

                        : kInactiveCardColour,

                    cardChild: IconContent(

                      icon: Icons.male,

                      label: 'MALE',

                    ),

                  ),

                ),

                Expanded(

                  child: child\_i\_c(

                    onPress: () {

                      setState(() {

                        selectedGender = Gender.female;

                      });

                    },

                    colour: selectedGender == Gender.female

                        ? kActiveCardColour

                        : kInactiveCardColour,

                    cardChild: IconContent(

                      icon: Icons.female,

                      label: 'FEMALE',

                    ),

                  ),

                ),

              ],

            ),

          ),

          Expanded(

            child: child\_i\_c(

              colour: kActiveCardColour,

              cardChild: Column(

                mainAxisAlignment: MainAxisAlignment.center,

                children: <Widget>[

                  Text(

                    'HEIGHT',

                    style: kLabelTextStyle,

                  ),

                  Row(

                    mainAxisAlignment: MainAxisAlignment.center,

                    crossAxisAlignment: CrossAxisAlignment.baseline,

                    textBaseline: TextBaseline.alphabetic,

                    children: <Widget>[

                      Text(

                        height.toString(),

                        style: kNumberTextStyle,

                      ),

                      Text(

                        'cm',

                        style: kLabelTextStyle,

                      ),

                    ],

                  ),

                  SliderTheme(

                    data: SliderTheme.of(context).copyWith(

                      thumbColor: Color(0xFFEB1555),

                      activeTrackColor: Colors.white,

                      inactiveTrackColor: Color(0xFF8D8E98),

                      overlayColor: Color(0x29EB1555),

                      thumbShape:

                          RoundSliderThumbShape(enabledThumbRadius: 15.0),

                      overlayShape:

                          RoundSliderOverlayShape(overlayRadius: 30.0),

                    ),

                    child: Slider(

                      value: height.toDouble(),

                      min: 120.0,

                      max: 220.0,

                      onChanged: (double newValue) {

                        setState(() {

                          height = newValue.round();

                        });

                      },

                    ),

                  ),

                ],

              ),

            ),

          ),

          Expanded(

            child: Row(

              children: <Widget>[

                Expanded(

                  child: child\_i\_c(

                    colour: kActiveCardColour,

                    cardChild: Column(

                      mainAxisAlignment: MainAxisAlignment.center,

                      children: <Widget>[

                        Text(

                          'WEIGHT',

                          style: kLabelTextStyle,

                        ),

                        Text(

                          weight.toString(),

                          style: kNumberTextStyle,

                        ),

                        Row(

                          mainAxisAlignment: MainAxisAlignment.center,

                          children: [

                            child\_w\_h(

                              icon: Icons.remove,

                              onPressed: () {

                                setState(() {

                                  weight--;

                                });

                              },

                            ),

                            SizedBox(

                              width: 10.0,

                            ),

                            child\_w\_h(

                              icon: Icons.add,

                              onPressed: () {

                                setState(() {

                                  weight++;

                                });

                              },

                            ),

                          ],

                        ),

                      ],

                    ),

                  ),

                ),

                Expanded(

                  child: child\_i\_c(

                    colour: kActiveCardColour,

                    cardChild: Column(

                      mainAxisAlignment: MainAxisAlignment.center,

                      children: <Widget>[

                        Text(

                          'AGE',

                          style: kLabelTextStyle,

                        ),

                        Text(

                          age.toString(),

                          style: kNumberTextStyle,

                        ),

                        Row(

                          mainAxisAlignment: MainAxisAlignment.center,

                          children: <Widget>[

                            child\_w\_h(

                              icon: Icons.remove,

                              onPressed: () {

                                setState(() {

                                  age--;

                                });

                              },

                            ),

                            SizedBox(

                              width: 10.0,

                            ),

                            child\_w\_h(

                              icon: Icons.add,

                              onPressed: () {

                                setState(() {

                                  age++;

                                });

                              },

                            ),

                          ],

                        ),

                      ],

                    ),

                  ),

                ),

              ],

            ),

          ),

          calculatebutton(

            buttonTitle: 'CALCULATE',

            onTap: () {

              bmiBrain calc = bmiBrain(height: height, weight: weight);

              Navigator.push(

                context,

                MaterialPageRoute(

                    builder: (context) => ResultsPage(

                          bmiResult: calc.calculateBMI(),

                          resultText: calc.getResult(),

                          interpretation: calc.getInterpretation(),

                        )),

              );

            },

          ),

        ],

      ),

    );

  }

}

**Icon:**

import 'package:flutter/material.dart';

import 'textstyle.dart';

class IconContent extends StatelessWidget {

  IconContent({required this.icon, required this.label});

  final IconData icon;

  final String label;

  @override

  Widget build(BuildContext context) {

    return Column(

      mainAxisAlignment: MainAxisAlignment.center,

      children: <Widget>[

        Icon(

          icon,

          size: 80.0,

        ),

        SizedBox(

          height: 15.0,

        ),

        Text(

          label,

          style: kLabelTextStyle,

        ),

      ],

    );

  }

}

**Main:**

import 'package:flutter/material.dart';

import 'first\_screen.dart';

void main() {

  runApp(BMICalculator());

}

class BMICalculator extends StatelessWidget {

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

  @override

  Widget build(BuildContext context) {

    return MaterialApp(

      debugShowCheckedModeBanner: false,

      theme: ThemeData.dark().copyWith(

        primaryColor: Color(0xFF0A0E21),

        scaffoldBackgroundColor: Color(0xFF0A0E21),

      ),

      home: InputPage(),

    );

  }

}

**Second Screen:**

import 'package:flutter/material.dart';

import 'child\_w\_h.dart';

import 'child\_i\_c.dart';

import 'calculate\_button.dart';

import 'textstyle.dart';

class ResultsPage extends StatelessWidget {

  ResultsPage(

      {required this.bmiResult,

      required this.resultText,

      required this.interpretation});

  final String bmiResult;

  final String resultText;

  final String interpretation;

  @override

  Widget build(BuildContext context) {

    return Scaffold(

      appBar: AppBar(

        backgroundColor: Color(0xFF1D1E33),

        title: Text('BMI CALCULATOR'),

      ),

      body: Column(

        mainAxisAlignment: MainAxisAlignment.spaceEvenly,

        crossAxisAlignment: CrossAxisAlignment.stretch,

        children: <Widget>[

          Expanded(

            child: Container(

              padding: EdgeInsets.all(15.0),

              alignment: Alignment.bottomLeft,

              child: Text(

                'Your Result',

                style: kTitleTextStyle,

              ),

            ),

          ),

          Expanded(

            flex: 5,

            child: child\_i\_c(

              colour: kActiveCardColour,

              cardChild: Column(

                crossAxisAlignment: CrossAxisAlignment.center,

                mainAxisAlignment: MainAxisAlignment.spaceEvenly,

                children: <Widget>[

                  Text(

                    resultText.toUpperCase(),

                    style: kResultTextStyle,

                  ),

                  Text(

                    bmiResult,

                    style: kBMITextStyle,

                  ),

                  Text(

                    interpretation,

                    textAlign: TextAlign.center,

                    style: kBodyTextStyle,

                  ),

                ],

              ),

            ),

          ),

          calculatebutton(

            buttonTitle: 'RE-CALCULATE',

            onTap: () {

              Navigator.pop(context);

            },

          ),

        ],

      ),

    );

  }

}

**Textstyle:**

import 'package:flutter/material.dart';

const kBbottomContainerHeight = 80.0;

const kActiveCardColour = Color(0xFF1D1E33);

const kInactiveCardColour = Color(0xFF111328);

const kBottomContainerColour = Color(0xFFEB1555);

const kLabelTextStyle = TextStyle(

  fontSize: 10.0,

  color: Color(0xFF8D8E98),

);

const kNumberTextStyle = TextStyle(

  fontSize: 50.0,

  fontWeight: FontWeight.w900,

);

const kLargeButtonTextStyle = TextStyle(

  fontSize: 25.0,

  fontWeight: FontWeight.bold,

);

const kTitleTextStyle = TextStyle(

  fontWeight: FontWeight.bold,

  fontSize: 25.0,

);

const kResultTextStyle = TextStyle(

  fontSize: 22.0,

  fontWeight: FontWeight.bold,

  color: Color(0xFF24D876),

);

const kBMITextStyle = TextStyle(

  fontSize: 100.0,

  fontWeight: FontWeight.bold,

);

const kBodyTextStyle = TextStyle(

  fontSize: 22.0,

);

**Output:**

A screenshot of a computer

Description automatically generated with medium confidence

Graphical user interface, application, website

Description automatically generated