SARTHAK KESARWANI

1NT19IS141

**Hybrid Application Development**

Classes & Objects in Dart: Quiz App

main.dart

import 'package:flutter/material.dart';

*// import 'package:quiz/question.dart';*

void main() { runApp(MaterialApp(

home:SafeArea( child: Scaffold(

body:QuizPage(),

),

),

));

}

class Question {

final String questionText; *// Question*

final bool answer; *// Answer*

Question({required this.questionText, required this.answer});

*// making it as named arguments for the Question Constructor*

}

class Questions {

List<Question> questionBank = [ *// create a list of Questions using default*

Question(questionText: "The atomic number for lithium is 17", answer: true),

*// set the Question and Answer*

Question(questionText: "A cross between a horse and a zebra is called a 'Hobra'", answer: true),

Question(questionText: "Alliumphobia is a fear of garlic", answer: false), Question(questionText: "Messi is goat", answer: true), Question(questionText: "Rajesh is not CR", answer: false), Question(questionText: "Had is interesting", answer: true), Question(questionText: "Prateek is topper", answer: true),

];

}

class QuizPage extends StatefulWidget {

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

@override

\_QuizPageState createState() => \_QuizPageState();

}

class \_QuizPageState extends State<QuizPage> { int questionNumber=0;

int currentScore=0;

Questions questions=Questions(); void updateQuestionNumber(){

setState(() {

questionNumber=questionNumber+1; print('$questionNumber');

});

}

void updateCurrentScore(bool choice,int question\_number){ if(questions.questionBank.length==question\_number){

print("end of question");

}else{

if(questions.questionBank[question\_number].answer==choice){ setState(() {

currentScore++;

});

}

}

}

bool checkquestionNumber(int questionNumber){

return questionNumber<questions.questionBank.length?true:false;

}

@override

Widget build(BuildContext context) { return Container(

child: Column( children: [

Center( child: Text(

checkquestionNumber(questionNumber)? questions.questionBank[questionNumber].questionText.toString() : " End",

style: TextStyle(fontSize: 40.0),

),

),

SizedBox(height: 20.0),

if (checkquestionNumber(questionNumber)) ElevatedButton(

onPressed: () {

setState(() {

if (questionNumber == questions.questionBank.length) { *//*

*check the bound*

print("End of questions");

} else {

*// check the user answer against the answer in the list*

updateCurrentScore(true, questionNumber);

*// increment the Question Number*

updateQuestionNumber();

}

});

},

child: Text('True'),

),

SizedBox(width: 20.0),

if (checkquestionNumber(questionNumber)) ElevatedButton(

onPressed: () {

setState(() {

if (questionNumber == questions.questionBank.length) { *//*

*check the bound*

print("End of questions");

} else {

*// check the user answer against the answer in the list*

updateCurrentScore(false, questionNumber);

*// increment the Question Number*

updateQuestionNumber();

}

});

},

child: Text('False'),

),

SizedBox( height: 100.0,

),

SizedBox( height: 50.0,

),

Padding(

padding: const EdgeInsets.all(30.0), child: Center(

child: Text(

"Current Score is:",

style: TextStyle(fontSize: 30),

),

),

),

Padding(

padding: const EdgeInsets.all(30.0), child: Center(

child: Text( '${currentScore}',

style: TextStyle(fontSize: 30),

),

),

),

]

),

);

}

}

OutPut:

