

1NT19IS022
ANANYA BILLAVA
C1-BATCH

1.Implement a quiz application using utter that consist of two buttons true/false. Upon answering the question, the result of the answer should be shown and the user will navigate to the next question. Once the questions in the quiz ends, user should be shown the score.

```
import 'package:flutter/material.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});
}

class Questions {
  List<Question> questionBank = [
    Question(questionText: "Ananya is a good dancer", answer: true),
    Question(questionText: "Sunday is a fun day", answer: true),
    Question(questionText: "sahana is screaming", answer: false),
    Question(questionText: "Movies are very bad", answer: true),
    Question(questionText: "putin likes dogs", answer: false),
    Question(questionText: "Japan is anime world", answer: true),
    Question(questionText: "Modi is PM", 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) {

                    print("End of questions");
                  } else {

                    updateCurrentScore(true, questionNumber);

```

```

        updateQuestionNumber();
    }
    });
    },
    child: Text('True'),
),
 SizedBox(width: 20.0),

if (checkquestionNumber(questionNumber))
  ElevatedButton(
    onPressed: () {
      setState(() {
        if (questionNumber == questions.questionBank.length) {

          print("End of questions");
        } else {

          updateCurrentScore(false, questionNumber);

          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),
      ),
    ),
  ),
),

```

```
    },  
    );  
}  
}
```





