```swift
//
//  BusinessCalculations.swift
//  FBLA-QuizME
//
//  Created by Udit Garg on 11/24/18.
//  Copyright © 2018 Udit Garg. All rights reserved.
//

import Foundation
import UIKit


class BusinessCalculations: UIViewController {

    // Set up variables that represent labels and buttons on the storyboard
    @IBOutlet weak var QuestionLabel: UILabel!
    @IBOutlet weak var Answer1: UIButton!
    @IBOutlet weak var Answer2: UIButton!
    @IBOutlet weak var Answer3: UIButton!
    @IBOutlet weak var Answer4: UIButton!
    @IBOutlet weak var NextQuestion: UIButton!
    @IBOutlet weak var ScoreLabel: UILabel!

    // Create an array of integers that represent the number of questions
    var randomQuestionArray:[Int] = [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12]

    override func viewDidLoad() {
        super.viewDidLoad()

        // Hide Initial Next Question Buttons
        NextQuestion.isHidden = true

        //Format the buttons
        Answer1.layer.borderWidth=1
        Answer1.layer.borderColor=UIColor.darkGray.cgColor
        Answer1.layer.cornerRadius=5
        Answer2.layer.borderWidth=1
        Answer2.layer.borderColor=UIColor.darkGray.cgColor
        Answer2.layer.cornerRadius=5
        Answer3.layer.borderWidth=1
        Answer3.layer.borderColor=UIColor.darkGray.cgColor
        Answer3.layer.cornerRadius=5
        Answer4.layer.borderWidth=1
        Answer4.layer.borderColor=UIColor.darkGray.cgColor
        Answer4.layer.cornerRadius=5

        // Set the answers to be incorrect
        Answer1Correct = false
        Answer2Correct = false
        Answer3Correct = false
        Answer4Correct = false

        // As soon as the view loads start generating the questions
        RandomQuestions()

        ScoreNumber = 0
    }

    override func didReceiveMemoryWarning() {
        super.didReceiveMemoryWarning()
        // Dispose of any resources that can be recreated.
```

```
    }

    // If the answer is right then the Next Question button is enabled
    func rightAnswer() {
        NextQuestion.isHidden = false
        ScoreNumber = Int(ScoreNumber) + 2
        ScoreLabel.text = String(format: "%i", ScoreNumber)
    }

    // If the answer is wrong then the Next Question button is hidden
    func wrongAnswer() {
        ScoreNumber = Int(ScoreNumber) - 1
        ScoreLabel.text = String(format: "%i", ScoreNumber)
    }

    // This function randomly generates questions without repeat
    func RandomQuestions(){

        Answer1.isEnabled = true
        Answer2.isEnabled = true
        Answer3.isEnabled = true
        Answer4.isEnabled = true


        // This makes randomIndex represent the number of questions available for this
question
        let randomIndex = Int(arc4random_uniform(UInt32(randomQuestionArray.count)))

        // Generates questions until all of the questions for this topic have been
answered
        if randomQuestionArray.count > -1 {

            switch (randomQuestionArray[randomIndex]) {
            case 0:
                QuestionLabel.text = "Which of the following is NOT a heading in a
check register?"
                Answer1.setTitle("interest rate", for: .normal)
                Answer2.setTitle("check number", for: .normal)
                Answer3.setTitle("balance", for: .normal)
                Answer4.setTitle("transaction", for: .normal)
                Answer1Correct = true
            case 1:
                QuestionLabel.text = "Manuel is paid $145 each day he guides tourists
on camping trips in the Grand Canyon. What amount did he earn last month if he worked
19 days?"
                Answer1.setTitle("$2755", for: .normal)
                Answer2.setTitle("$3500", for: .normal)
                Answer3.setTitle("$3000", for: .normal)
                Answer4.setTitle("none of the above", for: .normal)
                Answer1Correct = true
            case 2:
                QuestionLabel.text = "By investing in many companies, mutual funds
increase the chance of buying stocks that will be"
                Answer1.setTitle("risky", for: .normal)
                Answer2.setTitle("profitable", for: .normal)
                Answer3.setTitle("equitable", for: .normal)
                Answer4.setTitle("aggressive", for: .normal)
                Answer2Correct = true
            case 3:
                QuestionLabel.text = "Which of the following helps measure the amount
of financial risk a business faces by showing the level of debt the firm is carrying?"
```

```
                Answer1.setTitle("debt to equity ratio", for: .normal)
                Answer2.setTitle("current ratio", for: .normal)
                Answer3.setTitle("income statement", for: .normal)
                Answer4.setTitle("return on equity", for: .normal)
                Answer1Correct = true
            case 4:
                QuestionLabel.text = "Randie’s monthly budget allows for 23% of her
income to go towards food costs. If she made $5,364 last month, what was her food
budget?"
                Answer1.setTitle("$4130.28", for: .normal)
                Answer2.setTitle("$1341", for: .normal)
                Answer3.setTitle("$1233.72", for: .normal)
                Answer4.setTitle("none of the above", for: .normal)
                Answer3Correct = true
            case 5:
                QuestionLabel.text = "Mark Snyder deposited $2,200 in a six month CD
that pays 4.5% simple annual interest. How much interest did he earn at the end of the
term?"
                Answer1.setTitle("$49.50", for: .normal)
                Answer2.setTitle("$4.90", for: .normal)
                Answer3.setTitle("$495", for: .normal)
                Answer4.setTitle("none of the above", for: .normal)
                Answer1Correct = true
            case 6:
                QuestionLabel.text = "Fran’s Internet connection has a true speed of
350 kbps. How long would it take her to download a 4.35 MB file to the nearest tenth
of a second?"
                Answer1.setTitle("99.4 seconds", for: .normal)
                Answer2.setTitle("89.4 seconds", for: .normal)
                Answer3.setTitle("109.4 seconds", for: .normal)
                Answer4.setTitle("none of the above", for: .normal)
                Answer1Correct = true
            case 7:
                QuestionLabel.text = "A homeowners policy has a face value of $57,000
with a $975 deductible. How much will the insurance company pay on a $2,410.19 loss?"
                Answer1.setTitle("$2410.19", for: .normal)
                Answer2.setTitle("$57,00", for: .normal)
                Answer3.setTitle("$1435.19", for: .normal)
                Answer4.setTitle("none of the above", for: .normal)
                Answer3Correct = true
            case 8:
                QuestionLabel.text = "A mutual fund has a net asset value of $8.23 and
an offer price of $8.75. What is the rate of commission, to the nearest tenth
percent?"
                Answer1.setTitle("3.9%", for: .normal)
                Answer2.setTitle("5.9%", for: .normal)
                Answer3.setTitle("4.9%", for: .normal)
                Answer4.setTitle("6.9%", for: .normal)
                Answer2Correct = true
            case 9:
                QuestionLabel.text = "Hank bought 25, $1,000 Hampton Railroad bonds at
100.823. The broker charged a commission of $4 per bond. What total investment did
Hank make in the bonds? "
                Answer1.setTitle("$23,505.75", for: .normal)
                Answer2.setTitle("$25,305.75", for: .normal)
                Answer3.setTitle("$20,305.75", for: .normal)
                Answer4.setTitle("none of the above", for: .normal)
                Answer2Correct = true
            case 10:
```

```
                QuestionLabel.text = "If it took 4 programmers 12 days to develop a
program containing 1,680 lines of code, how many lines of code per day did each
programmer average?"
                Answer1.setTitle("25", for: .normal)
                Answer2.setTitle("30", for: .normal)
                Answer3.setTitle("35", for: .normal)
                Answer4.setTitle("40", for: .normal)
                Answer3Correct = true
            case 11:
                QuestionLabel.text = "A newspaper has an estimated 103,000 daily
readers. What is the cost per reader of an ad that costs $9,750, to the nearest tenth
of a cent?"
                Answer1.setTitle(".95", for: .normal)
                Answer2.setTitle(".095", for: .normal)
                Answer3.setTitle(".0095", for: .normal)
                Answer4.setTitle("none of the above", for: .normal)
                Answer2Correct = true
            case 12:
                QuestionLabel.text = "Find the value of cin this equation: c= (1 m −15
cm) ÷ 0.5"
                Answer1.setTitle("10 m", for: .normal)
                Answer2.setTitle("17 m", for: .normal)
                Answer3.setTitle("1.3 m", for: .normal)
                Answer4.setTitle("1.7 m", for: .normal)
                Answer4Correct = true
            default:
                break
            }
            // Removes the possibility of the question that was just shown to be shown
again
            randomQuestionArray.remove(at: randomIndex)
        }

        // If the user is on the last question then show them that they have reached
the last question
        if (randomQuestionArray.count < 1) {
            let alert = UIAlertController(title: "Wow!", message: "You have reached
the last question for Business Calculations! Nice Job! Complete this question and then
click on 'Your Score' for a rating!", preferredStyle: .alert)
            alert.addAction(UIAlertAction(title: "Continue", style: .default, handler:
{ action in
                switch action.style{
                case .default:
                    print("default")
                case .cancel:
                    print("cancel")
                case .destructive:
                    print("destructive")
                }}))
            self.present(alert, animated: true, completion: nil)
            NextQuestion.isEnabled = false
        }

        if (randomQuestionArray.count == 0) {
            let alert = UIAlertController(title: "Wow!", message: "You got
\(ScoreNumber) out of 13 questions correct nice job! To see a detailed breakdown of
your score, click-Your Score-next to your score number.", preferredStyle: .alert)
            alert.addAction(UIAlertAction(title: "Continue", style: .default, handler:
{ action in
                switch action.style{
                case .default:
```

```
                print("default")
            case .cancel:
                print("cancel")
            case .destructive:
                print("destructive")
            }}))
        self.present(alert, animated: true, completion: nil)
    }
}

// These 4 functions tell the user if they got the correct answer or if they got
the incorrect answer
@IBAction func Answer1(_ sender: Any) {
    if Bool(Answer1Correct) == true {
        rightAnswer()
        Answer1.layer.backgroundColor = UIColor.green.cgColor
        Answer1.isEnabled = false
        Answer2.isEnabled = false
        Answer3.isEnabled = false
        Answer4.isEnabled = false
    } else {
        wrongAnswer()
        Answer1.layer.backgroundColor = UIColor.red.cgColor
        Answer1.isEnabled = false
    }
}

@IBAction func Answer2(_ sender: Any) {
    if Bool(Answer2Correct) == true {
        rightAnswer()
        Answer2.layer.backgroundColor = UIColor.green.cgColor
        Answer2.isEnabled = false
        Answer1.isEnabled = false
        Answer3.isEnabled = false
        Answer4.isEnabled = false
    } else {
        wrongAnswer()
        Answer2.layer.backgroundColor = UIColor.red.cgColor
        Answer2.isEnabled = false
    }
}

@IBAction func Answer3(_ sender: Any) {
    if Bool(Answer3Correct) == true {
        rightAnswer()
        Answer3.layer.backgroundColor = UIColor.green.cgColor
        Answer3.isEnabled = false
        Answer1.isEnabled = false
        Answer2.isEnabled = false
        Answer4.isEnabled = false
    } else {
        wrongAnswer()
        Answer3.layer.backgroundColor = UIColor.red.cgColor
        Answer3.isEnabled = false
    }
}

@IBAction func Answer4(_ sender: Any) {
    if Bool(Answer4Correct) == true {
        rightAnswer()
        Answer4.layer.backgroundColor = UIColor.green.cgColor
```

```
            Answer4.isEnabled = false
            Answer1.isEnabled = false
            Answer2.isEnabled = false
            Answer3.isEnabled = false
        } else {
            wrongAnswer()
            Answer4.layer.backgroundColor = UIColor.red.cgColor
            Answer4.isEnabled = false
        }
    }

    // Resets the colors and answers and generates another question
    @IBAction func NextQuestion(_ sender: Any) {
        Answer1.layer.backgroundColor = UIColor.white.cgColor
        Answer2.layer.backgroundColor = UIColor.white.cgColor
        Answer3.layer.backgroundColor = UIColor.white.cgColor
        Answer4.layer.backgroundColor = UIColor.white.cgColor
        NextQuestion.isHidden = true
        Answer1Correct = false
        Answer2Correct = false
        Answer3Correct = false
        Answer4Correct = false
        RandomQuestions()
    }
}
```