```swift
//
//  PersonalFinance.swift
//  FBLA-QuizME
//
//  Created by Udit Garg on 11/28/18.
//  Copyright © 2018 Udit Garg. All rights reserved.
//

import Foundation
import UIKit
import MessageUI


class PersonalFinance: 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 = "Earning a degree at a four-year college is a ___
goal for a high school sophomore"
                Answer1.setTitle("short-term", for: .normal)
                Answer2.setTitle("long-term", for: .normal)
                Answer3.setTitle("mid-term", for: .normal)
                Answer4.setTitle("future financial", for: .normal)
                Answer2Correct = true
            case 1:
                QuestionLabel.text = "One example of non-taxable income is:"
                Answer1.setTitle("salary", for: .normal)
                Answer2.setTitle("child support", for: .normal)
                Answer3.setTitle("interest earned on savings", for: .normal)
                Answer4.setTitle("dividends earned on stocks", for: .normal)
                Answer2Correct = true
            case 2:
                QuestionLabel.text = "The sales tax is an example of which type of
tax?"
                Answer1.setTitle("regressive", for: .normal)
                Answer2.setTitle("progressive", for: .normal)
                Answer3.setTitle("excise", for: .normal)
                Answer4.setTitle("proportional", for: .normal)
                Answer1Correct = true
            case 3:
                QuestionLabel.text = "___ is not a commodity sold on the futures
market"
                Answer1.setTitle("Coal", for: .normal)
```

```
                Answer2.setTitle("Corn", for: .normal)
                Answer3.setTitle("Beef", for: .normal)
                Answer4.setTitle("Stock.", for: .normal)
                Answer4Correct = true
            case 4:
                QuestionLabel.text = "The ___ is the legally established ownership of
a home."
                Answer1.setTitle("lie", for: .normal)
                Answer2.setTitle("deed", for: .normal)
                Answer3.setTitle("title n", for: .normal)
                Answer4.setTitle("mortgage", for: .normal)
                Answer3Correct = true
            case 5:
                QuestionLabel.text = "Because of an error in the pricing bar code,
everyone who bought a pair of jeans at a nationwide clothing store last month was
overcharged by three dollars. The legal action to take would be:"
                Answer1.setTitle("a class action lawsuit", for: .normal)
                Answer2.setTitle("arbitration", for: .normal)
                Answer3.setTitle("negotiation", for: .normal)
                Answer4.setTitle("mediation", for: .normal)
                Answer1Correct = true
            case 6:
                QuestionLabel.text = "The ___ endorsement on a paycheck is the safest
for the consumer"
                Answer1.setTitle("blank", for: .normal)
                Answer2.setTitle("special", for: .normal)
                Answer3.setTitle("restrictive", for: .normal)
                Answer4.setTitle("two-party", for: .normal)
                Answer3Correct = true
            case 7:
                QuestionLabel.text = "___ can solve a sudden need for cash with a pre-
established amount that can be borrowed on demand with no collateral"
                Answer1.setTitle("Line of credit", for: .normal)
                Answer2.setTitle("Deferred billing", for: .normal)
                Answer3.setTitle("Collateral", for: .normal)
                Answer4.setTitle("Overdraft protection", for: .normal)
                Answer1Correct = true
            case 8:
                QuestionLabel.text = "The laws in each state that set the limits for
interest rates are called"
                Answer1.setTitle("Mastercard", for: .normal)
                Answer2.setTitle("Loan Sharks", for: .normal)
                Answer3.setTitle("Usury Laws", for: .normal)
                Answer4.setTitle("Better Business Bureau", for: .normal)
                Answer3Correct = true
            case 9:
                QuestionLabel.text = "Each of the following are ways to evaluate
possible investments EXCEPT"
                Answer1.setTitle("The chaos theory", for: .normal)
                Answer2.setTitle("The fundamental theory", for: .normal)
                Answer3.setTitle("The technical theory ", for: .normal)
                Answer4.setTitle("The efficient market theory", for: .normal)
                Answer1Correct = true
            case 10:
                QuestionLabel.text = "An example of a short term investment strategy
is "
                Answer1.setTitle("the buy and hold technique ", for: .normal)
                Answer2.setTitle("Buying stock on margin", for: .normal)
                Answer3.setTitle("Dollar cost averaging", for: .normal)
                Answer4.setTitle("A dividend reinvestment plan", for: .normal)
                Answer2Correct = true
```

```
            case 11:
                QuestionLabel.text = "Financial planes that are more than five years
off"
                Answer1.setTitle("Opportunity plans ", for: .normal)
                Answer2.setTitle("Long term goals", for: .normal)
                Answer3.setTitle("Intermediate goals", for: .normal)
                Answer4.setTitle("Attainable goals", for: .normal)
                Answer2Correct = true
            case 12:
                QuestionLabel.text = "A series of equal regular deposits is called "
                Answer1.setTitle("Serial deposits", for: .normal)
                Answer2.setTitle("An annuity", for: .normal)
                Answer3.setTitle("A personal financial plan", for: .normal)
                Answer4.setTitle("Paycheck", for: .normal)
                Answer2Correct = 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 Personal Finance! 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)
        }
    }
```

```swift
    // 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()
    }
}
```