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

import Foundation
import UIKit
import MessageUI

class BusinessLaw: 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 = "The laws of the United States are virtually all
based on"
                Answer1.setTitle("the Roman Code", for: .normal)
                Answer2.setTitle("administrative law", for: .normal)
                Answer3.setTitle("common law", for: .normal)
                Answer4.setTitle("moral law", for: .normal)
                Answer3Correct = true
            case 1:
                QuestionLabel.text = "Charging more than the maximum legal interest
rate is"
                Answer1.setTitle("fraud", for: .normal)
                Answer2.setTitle("usury", for: .normal)
                Answer3.setTitle("extortion", for: .normal)
                Answer4.setTitle("price-fixing", for: .normal)
                Answer2Correct = true
            case 2:
                QuestionLabel.text = "Charge accounts are examples of"
                Answer1.setTitle("usury", for: .normal)
                Answer2.setTitle("open-end credit", for: .normal)
                Answer3.setTitle("secured loans", for: .normal)
                Answer4.setTitle("closed-end credit", for: .normal)
                Answer2Correct = true
            case 3:
                QuestionLabel.text = "Most partnership law is set forth in the"
                Answer1.setTitle("Articles of Partnership", for: .normal)
                Answer2.setTitle("Uniform Commercial Code", for: .normal)
                Answer3.setTitle("Uniform Partnership Act", for: .normal)
```

```
                Answer4.setTitle("Tenancy in Partnership Act", for: .normal)
                Answer3Correct = true
            case 4:
                QuestionLabel.text = "All corporations issue"
                Answer1.setTitle("preferred stock", for: .normal)
                Answer2.setTitle("par value stock", for: .normal)
                Answer3.setTitle("common stock", for: .normal)
                Answer4.setTitle("no-par value stock", for: .normal)
                Answer3Correct = true
            case 5:
                QuestionLabel.text = "Breaking and entering with the intent to commit
a felony is"
                Answer1.setTitle("burglary", for: .normal)
                Answer2.setTitle("robbery", for: .normal)
                Answer3.setTitle("larceny", for: .normal)
                Answer4.setTitle("embezzlement", for: .normal)
                Answer1Correct = true
            case 6:
                QuestionLabel.text = "Insurance that protects the insured's car from
acts of nature, vandalism, or theft is"
                Answer1.setTitle("comprehensive insurance", for: .normal)
                Answer2.setTitle("property damage liability insurance", for: .normal)
                Answer3.setTitle("collision insurance", for: .normal)
                Answer4.setTitle("no-fault insurance", for: .normal)
                Answer1Correct = true
            case 7:
                QuestionLabel.text = "Making false statements under oath in court is"
                Answer1.setTitle("perjury", for: .normal)
                Answer2.setTitle("usury", for: .normal)
                Answer3.setTitle("breach of evidence", for: .normal)
                Answer4.setTitle("jeopardy", for: .normal)
                Answer1Correct = true
            case 8:
                QuestionLabel.text = "Fulfilling all terms of a contract properly and
totally is called"
                Answer1.setTitle("satisfactory performance", for: .normal)
                Answer2.setTitle("substantial performance", for: .normal)
                Answer3.setTitle("complete performance", for: .normal)
                Answer4.setTitle("tender of performance", for: .normal)
                Answer3Correct = true
            case 9:
                QuestionLabel.text = "A low down payment and smaller monthly payments
are advantages of"
                Answer1.setTitle("buying", for: .normal)
                Answer2.setTitle("financing", for: .normal)
                Answer3.setTitle("leasing", for: .normal)
                Answer4.setTitle("loaning.", for: .normal)
                Answer3Correct = true
            case 10:
                QuestionLabel.text = "Goods that presently exist and are the subject
matter of a particular contract are"
                Answer1.setTitle("specific goods", for: .normal)
                Answer2.setTitle("real goods", for: .normal)
                Answer3.setTitle("tangible goods", for: .normal)
                Answer4.setTitle("identified goods", for: .normal)
                Answer4Correct = true
            case 11:
                QuestionLabel.text = "The detailed view of a stock offering that must
be filed with the SEC is known as a"
                Answer1.setTitle("prospectusc", for: .normal)
                Answer2.setTitle("registration statement", for: .normal)
```

```
                Answer3.setTitle("financial statement", for: .normal)
                Answer4.setTitle("commerce clause", for: .normal)
                Answer1Correct = true
            case 12:
                QuestionLabel.text = "A mortgage with a fixed interest rate and
increasing monthly payments is a"
                Answer1.setTitle("fixed-rate mortgage", for: .normal)
                Answer2.setTitle("balloon mortgage", for: .normal)
                Answer3.setTitle("graduated-payment mortgage", for: .normal)
                Answer4.setTitle("variable-rate mortgage", for: .normal)
                Answer3Correct = 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 Law! 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()
    }
}
```