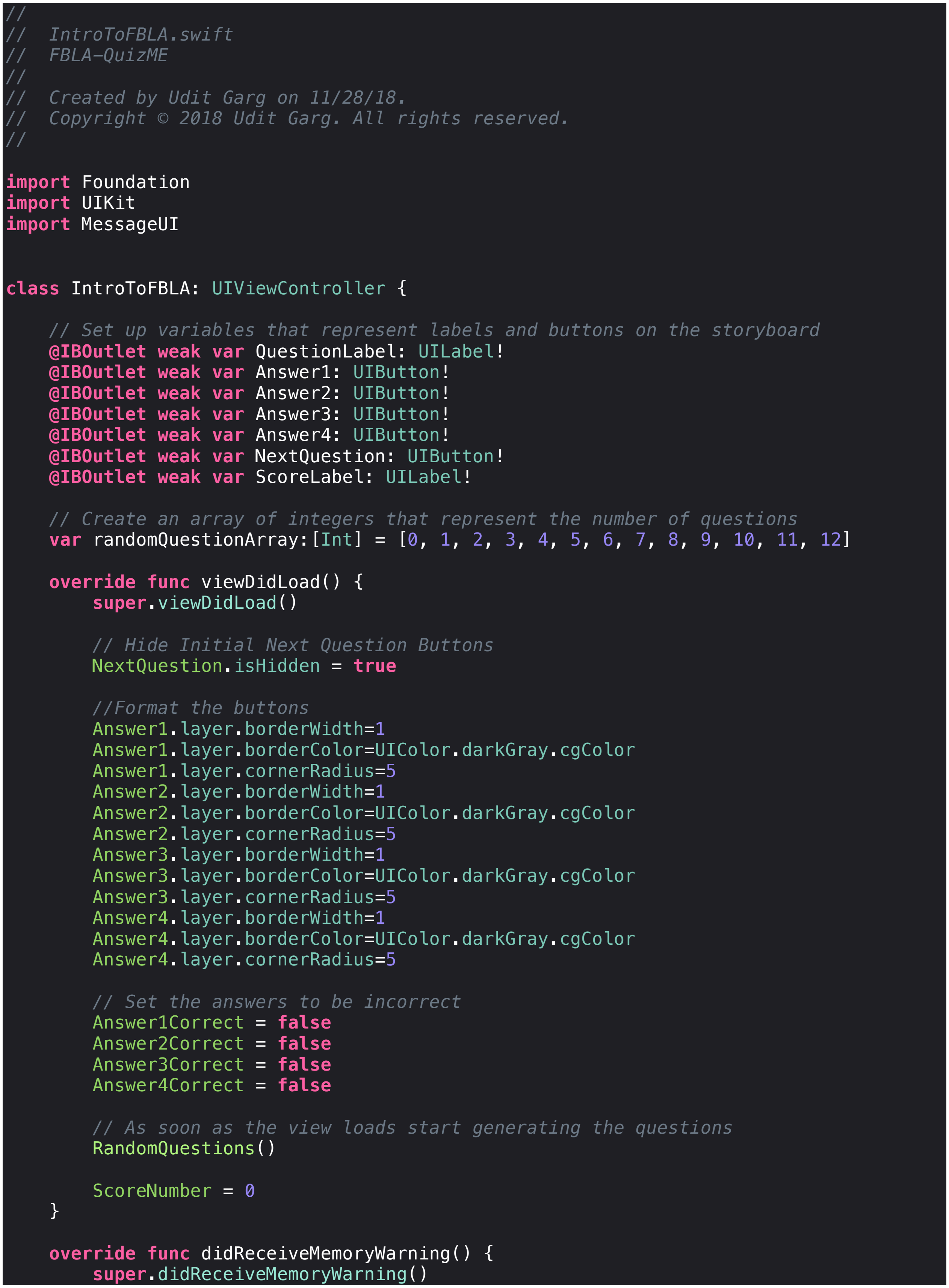
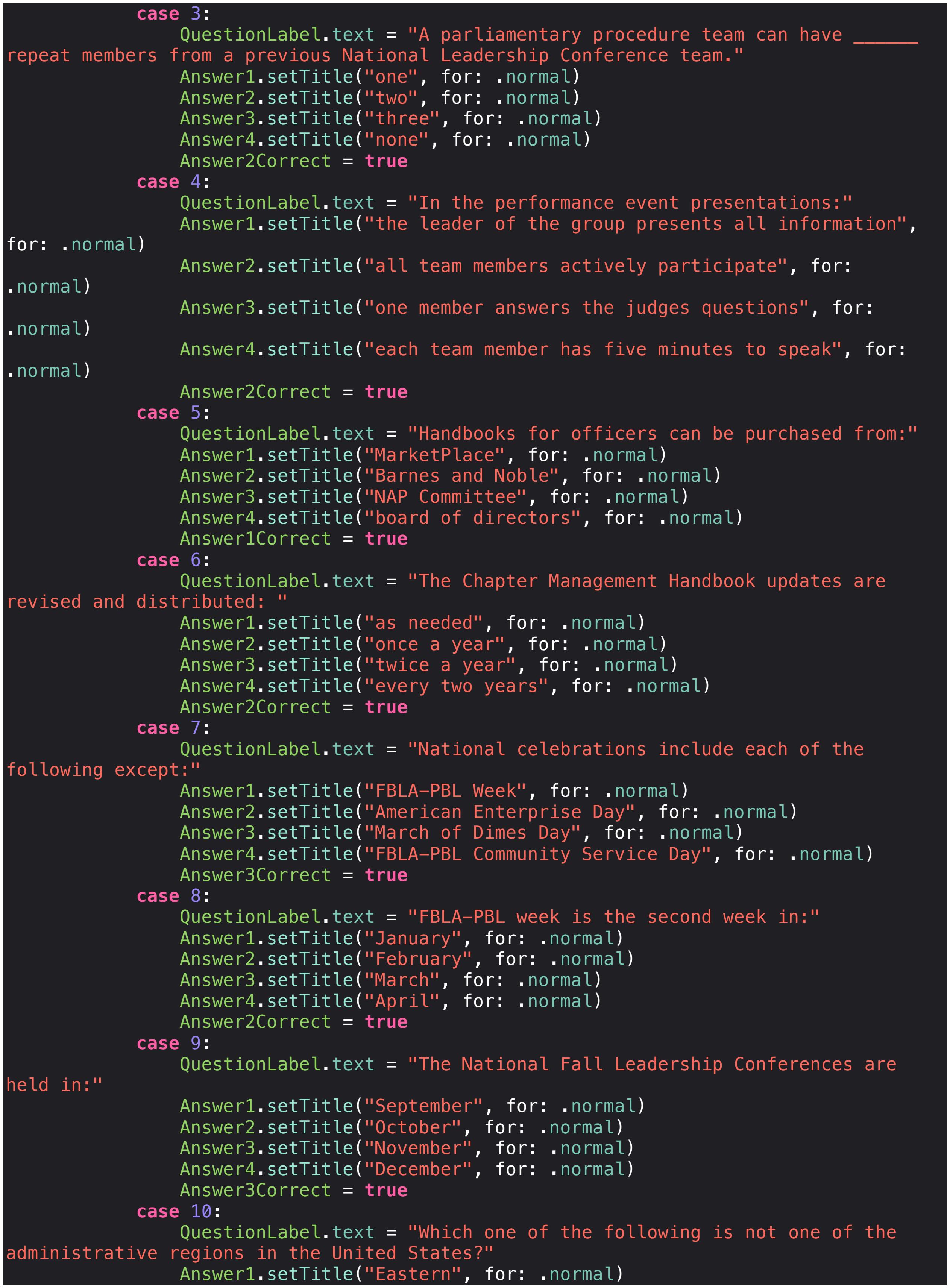
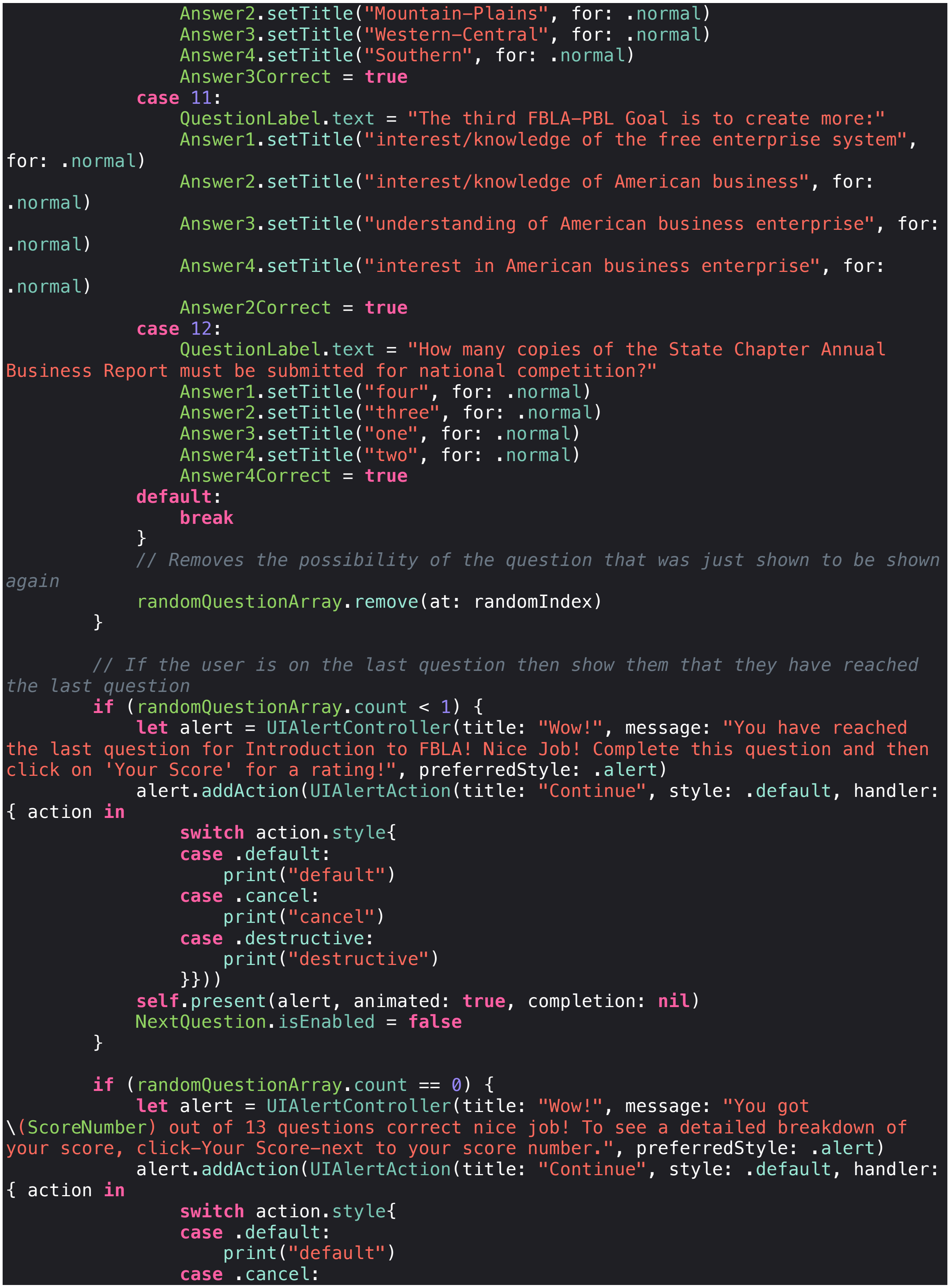


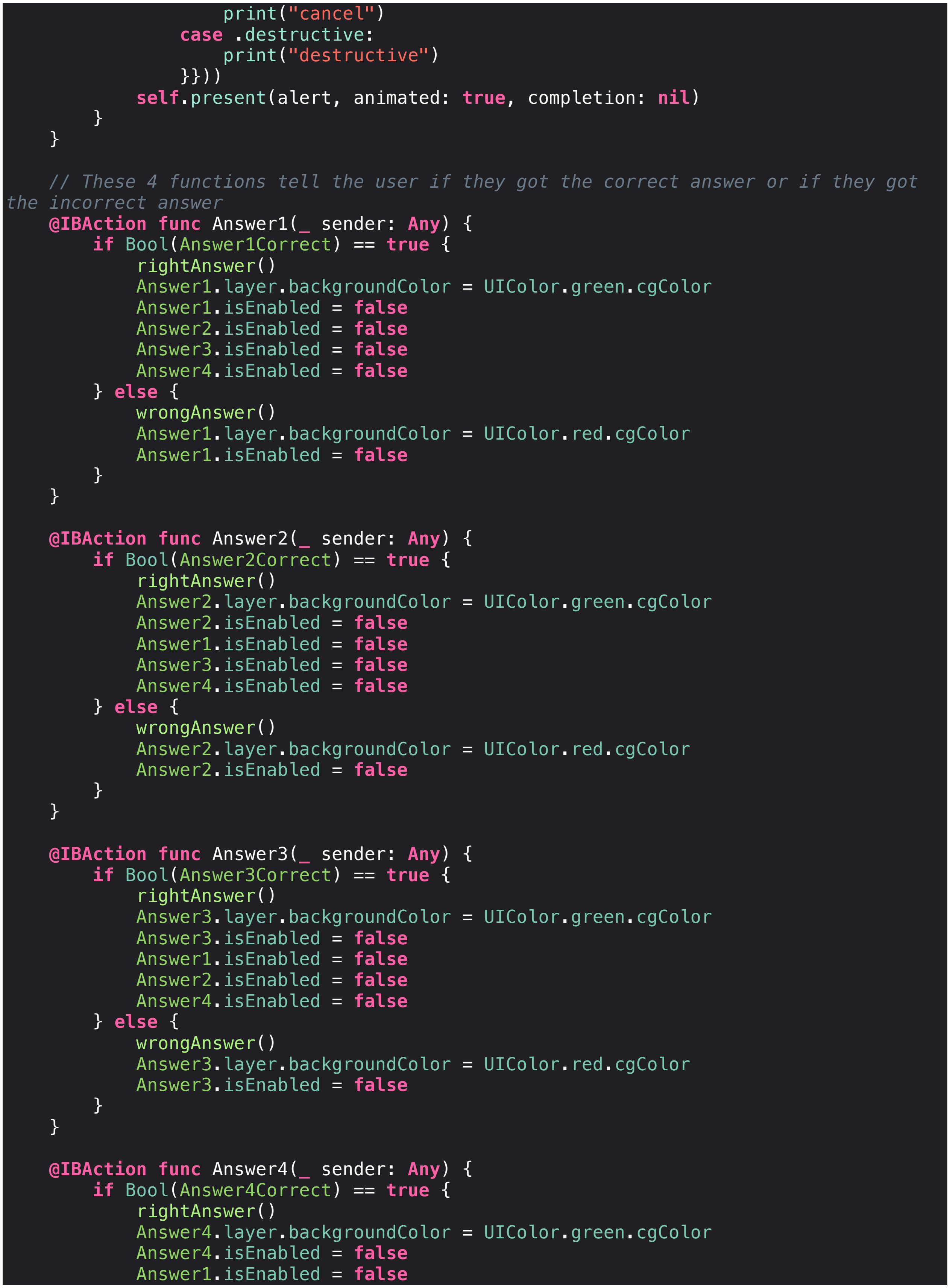
```swift
//
//  IntroToFBLA.swift
//  FBLA-QuizME
//
//  Created by Udit Garg on 11/28/18.
//  Copyright © 2018 Udit Garg. All rights reserved.
//

import Foundation
import UIKit
import MessageUI


class IntroToFBLA: 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 = "Key words found in the FBLA goals include:"
                Answer1.setTitle("confidence, understanding, efficient, scholarship", 
for: .normal)
                Answer2.setTitle("seek, sincere, abide, initiative", for: .normal)
                Answer3.setTitle("promise, aims, qualities, active", for: .normal)
                Answer4.setTitle("manner, directors, regulations, cooperation", for: 
.normal)
                Answer1Correct = true
            case 1:
                QuestionLabel.text = "In the last paragraph of the FBLA Creed, the 
following statement is made:"
                Answer1.setTitle("I promise to use my abilities", for: .normal)
                Answer2.setTitle("I agree to do my utmost", for: .normal)
                Answer3.setTitle("Every person should actively", for: .normal)
                Answer4.setTitle("Every person should prepare", for: .normal)
                Answer1Correct = true
            case 2:
                QuestionLabel.text = "The National Awards Program (NAP) provides 
events based on:"
                Answer1.setTitle("bylaws and goals of FBLA-PBL", for: .normal)
                Answer2.setTitle("guidelines in the Chapter Management Handbook", for: 
.normal)
                Answer3.setTitle("national office requests", for: .normal)
                Answer4.setTitle("goals of FBLA-PBL and educational programs", for: 
.normal)
                Answer4Correct = true
```



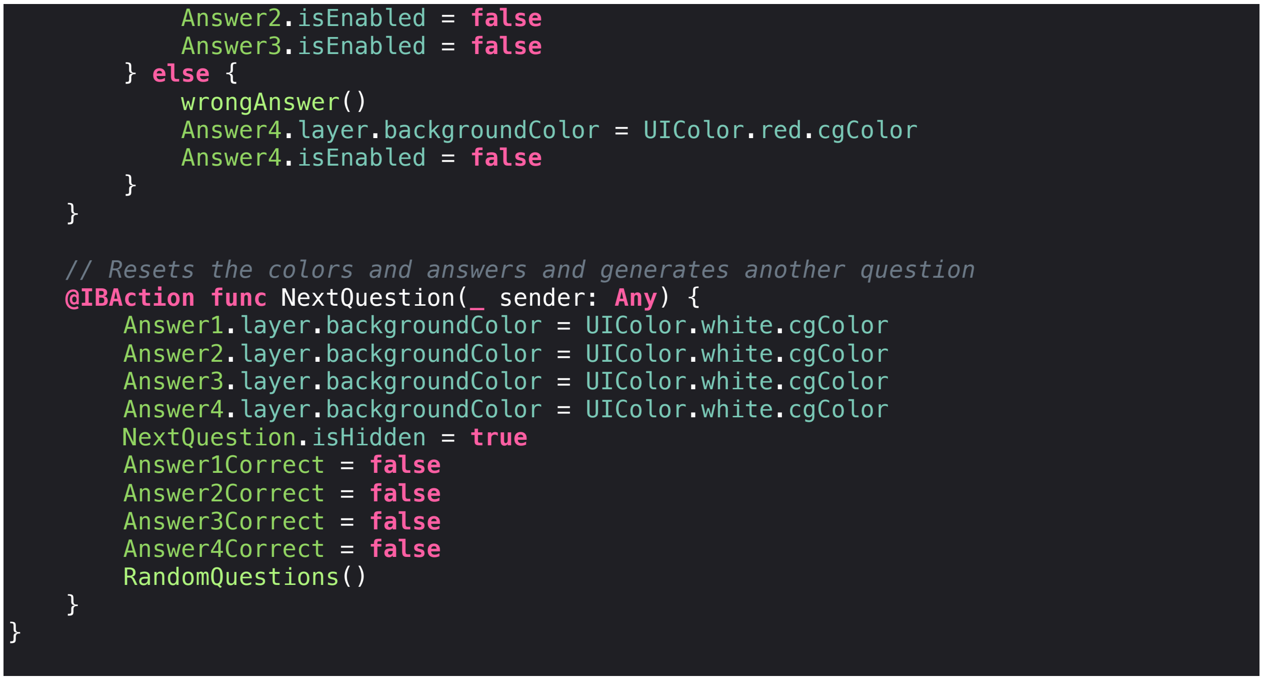
```
            case 3:
                QuestionLabel.text = "A parliamentary procedure team can have ______
repeat members from a previous National Leadership Conference team."
                Answer1.setTitle("one", for: .normal)
                Answer2.setTitle("two", for: .normal)
                Answer3.setTitle("three", for: .normal)
                Answer4.setTitle("none", for: .normal)
                Answer2Correct = true
            case 4:
                QuestionLabel.text = "In the performance event presentations:"
                Answer1.setTitle("the leader of the group presents all information",
for: .normal)
                Answer2.setTitle("all team members actively participate", for:
.normal)
                Answer3.setTitle("one member answers the judges questions", for:
.normal)
                Answer4.setTitle("each team member has five minutes to speak", for:
.normal)
                Answer2Correct = true
            case 5:
                QuestionLabel.text = "Handbooks for officers can be purchased from:"
                Answer1.setTitle("MarketPlace", for: .normal)
                Answer2.setTitle("Barnes and Noble", for: .normal)
                Answer3.setTitle("NAP Committee", for: .normal)
                Answer4.setTitle("board of directors", for: .normal)
                Answer1Correct = true
            case 6:
                QuestionLabel.text = "The Chapter Management Handbook updates are
revised and distributed: "
                Answer1.setTitle("as needed", for: .normal)
                Answer2.setTitle("once a year", for: .normal)
                Answer3.setTitle("twice a year", for: .normal)
                Answer4.setTitle("every two years", for: .normal)
                Answer2Correct = true
            case 7:
                QuestionLabel.text = "National celebrations include each of the
following except:"
                Answer1.setTitle("FBLA-PBL Week", for: .normal)
                Answer2.setTitle("American Enterprise Day", for: .normal)
                Answer3.setTitle("March of Dimes Day", for: .normal)
                Answer4.setTitle("FBLA-PBL Community Service Day", for: .normal)
                Answer3Correct = true
            case 8:
                QuestionLabel.text = "FBLA-PBL week is the second week in:"
                Answer1.setTitle("January", for: .normal)
                Answer2.setTitle("February", for: .normal)
                Answer3.setTitle("March", for: .normal)
                Answer4.setTitle("April", for: .normal)
                Answer2Correct = true
            case 9:
                QuestionLabel.text = "The National Fall Leadership Conferences are
held in:"
                Answer1.setTitle("September", for: .normal)
                Answer2.setTitle("October", for: .normal)
                Answer3.setTitle("November", for: .normal)
                Answer4.setTitle("December", for: .normal)
                Answer3Correct = true
            case 10:
                QuestionLabel.text = "Which one of the following is not one of the
administrative regions in the United States?"
                Answer1.setTitle("Eastern", for: .normal)
```



```
                Answer2.setTitle("Mountain-Plains", for: .normal)
                Answer3.setTitle("Western-Central", for: .normal)
                Answer4.setTitle("Southern", for: .normal)
                Answer3Correct = true
            case 11:
                QuestionLabel.text = "The third FBLA-PBL Goal is to create more:"
                Answer1.setTitle("interest/knowledge of the free enterprise system", 
for: .normal)
                Answer2.setTitle("interest/knowledge of American business", for: 
.normal)
                Answer3.setTitle("understanding of American business enterprise", for: 
.normal)
                Answer4.setTitle("interest in American business enterprise", for: 
.normal)
                Answer2Correct = true
            case 12:
                QuestionLabel.text = "How many copies of the State Chapter Annual 
Business Report must be submitted for national competition?"
                Answer1.setTitle("four", for: .normal)
                Answer2.setTitle("three", for: .normal)
                Answer3.setTitle("one", for: .normal)
                Answer4.setTitle("two", 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 Introduction to FBLA! 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()
    }
}
```