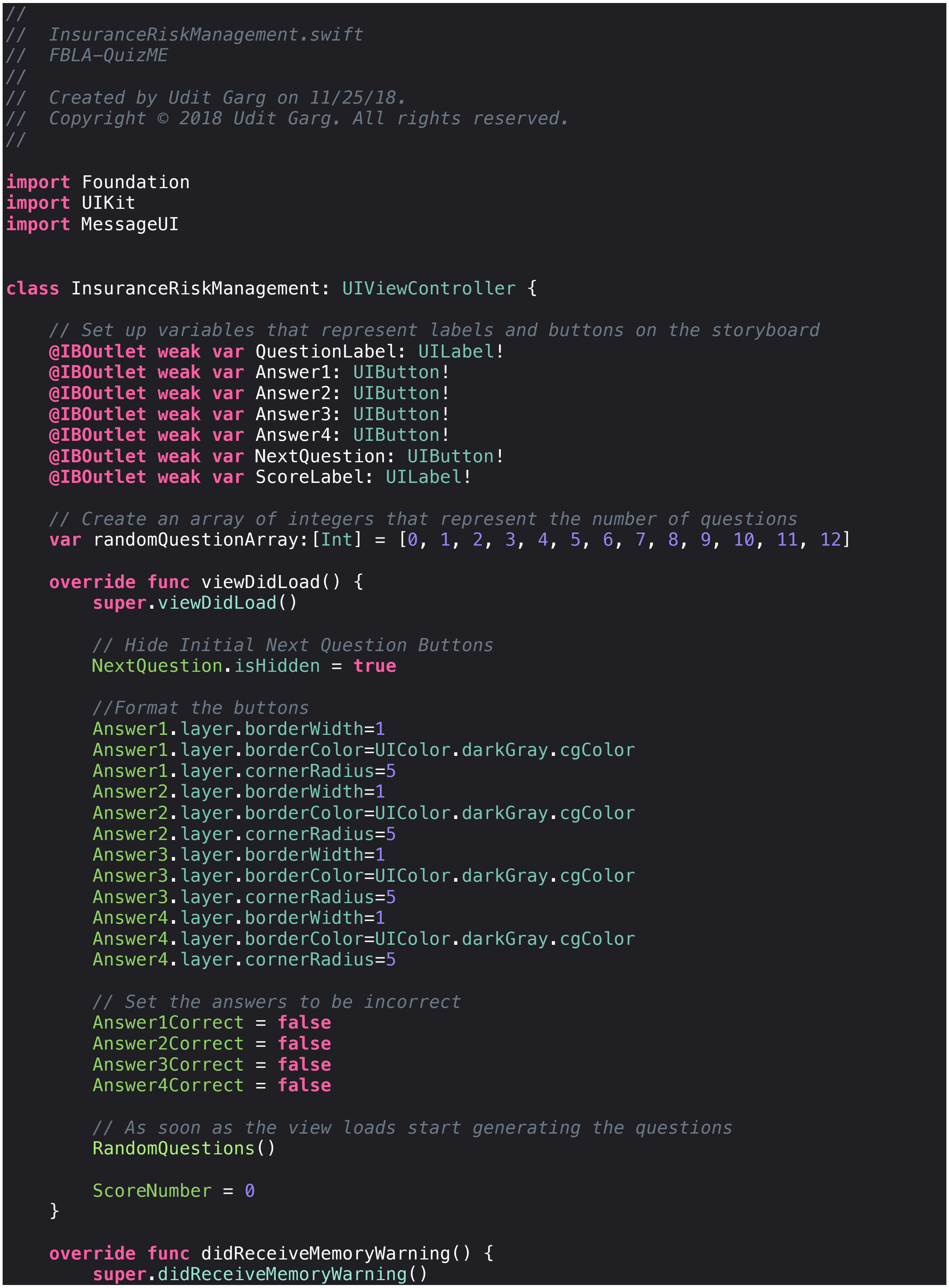
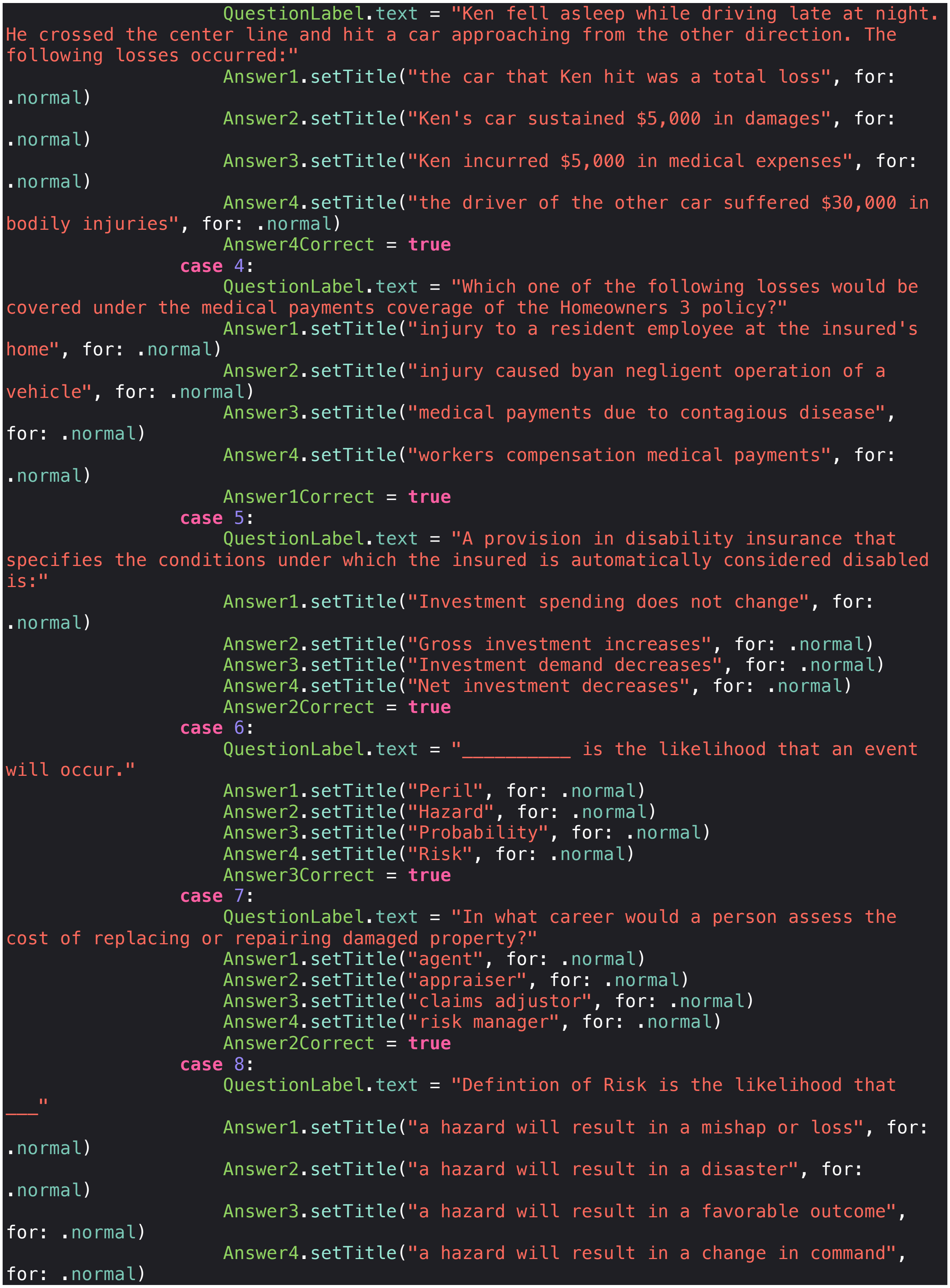
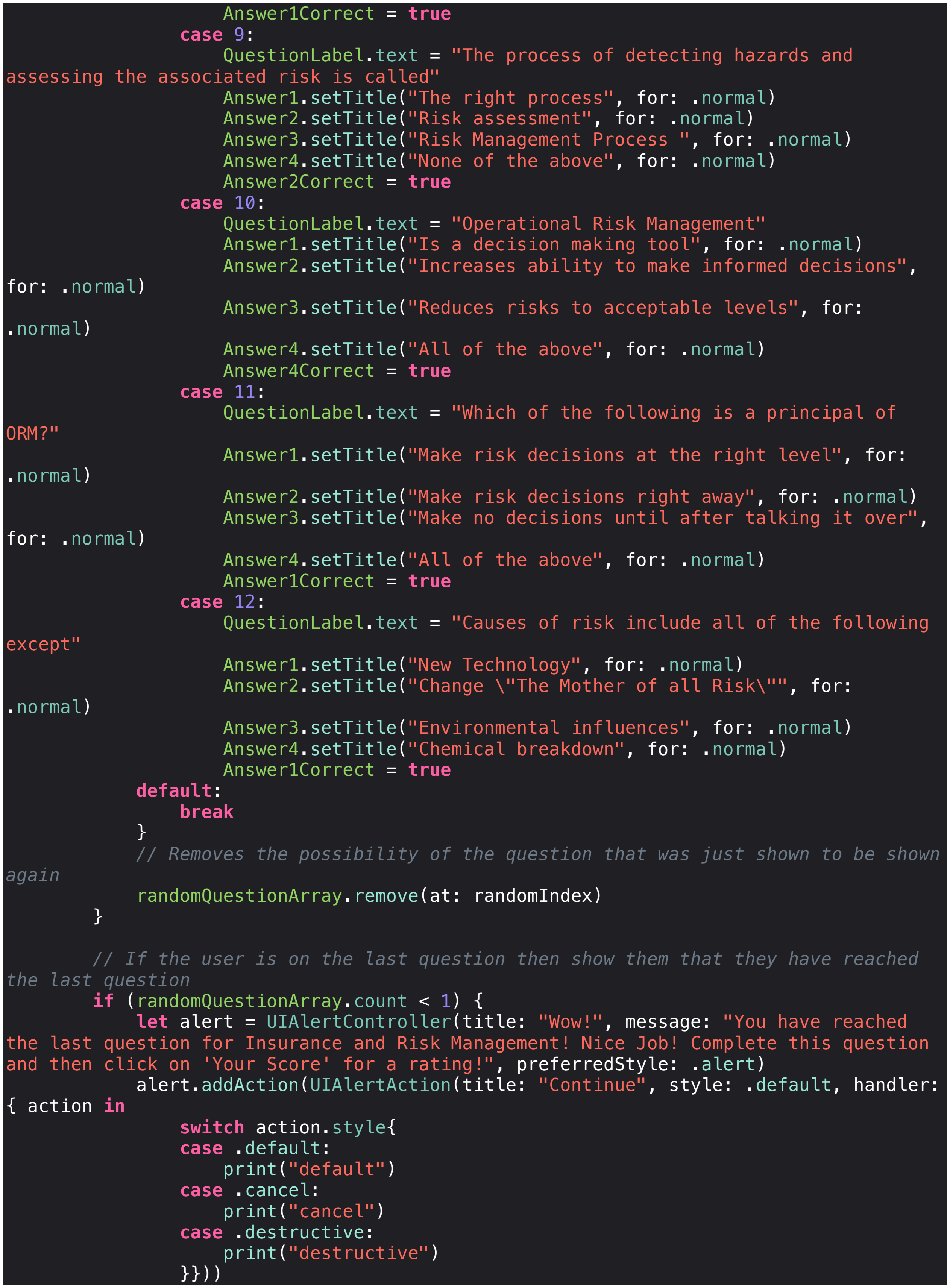


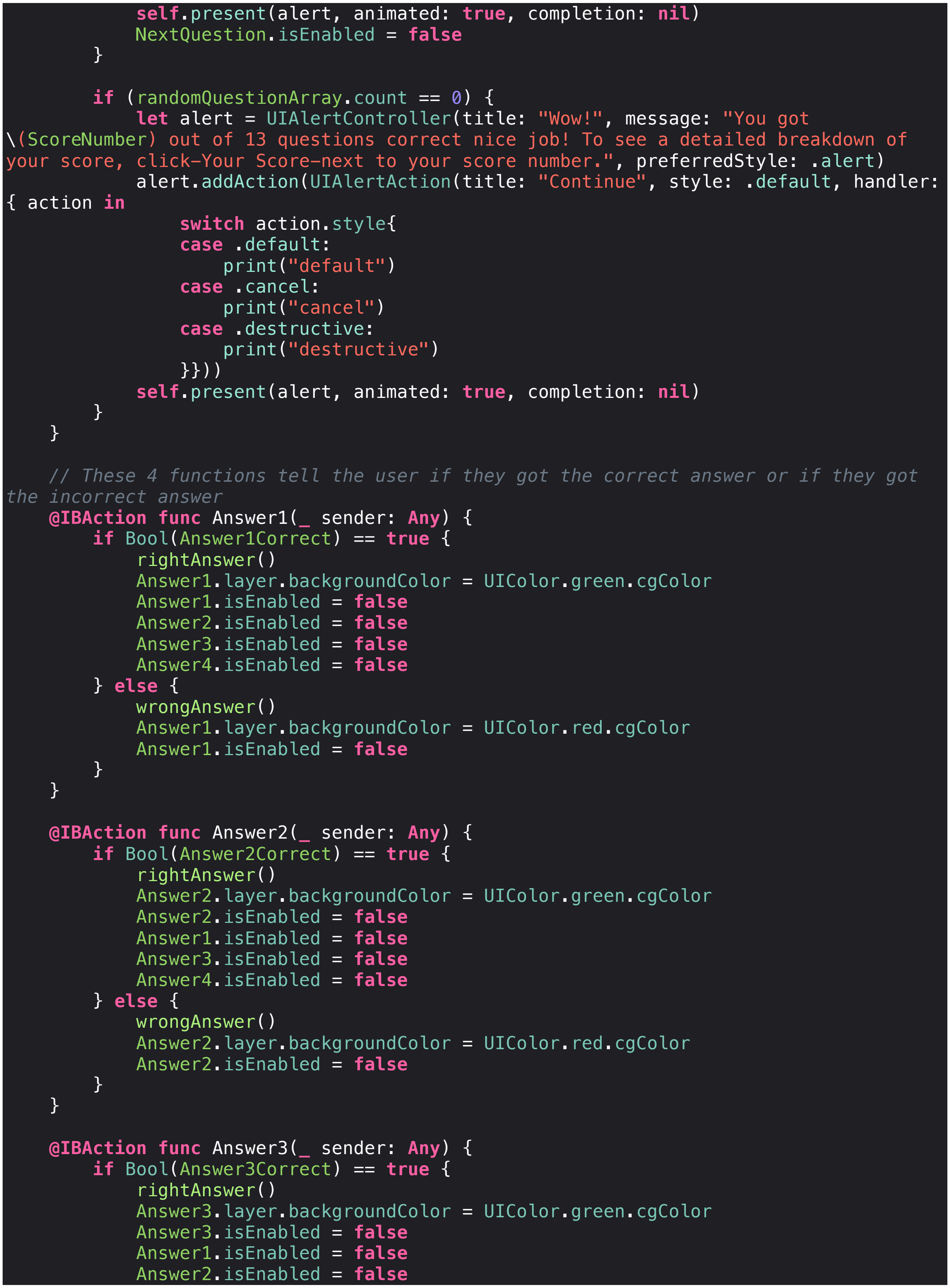
```swift
//
//  InsuranceRiskManagement.swift
//  FBLA-QuizME
//
//  Created by Udit Garg on 11/25/18.
//  Copyright © 2018 Udit Garg. All rights reserved.
//

import Foundation
import UIKit
import MessageUI


class InsuranceRiskManagement: 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 = "Pure risk may be said to create an economic burden in all except which one of the following ways?"
                    Answer1.setTitle("pure risk results in gambling", for: .normal)
                    Answer2.setTitle("pure risk may deprive society of certain resources", for: .normal)
                    Answer3.setTitle("pure risk can result in feelings of mental unrest", for: .normal)
                    Answer4.setTitle("all of the above", for: .normal)
                    Answer1Correct = true
                case 1:
                    QuestionLabel.text = "The risk of potential losses to others as a result of injury or damage you may have caused is"
                    Answer1.setTitle("property risk", for: .normal)
                    Answer2.setTitle("insurance", for: .normal)
                    Answer3.setTitle("liability risk", for: .normal)
                    Answer4.setTitle("personal risk", for: .normal)
                    Answer3Correct = true
                case 2:
                    QuestionLabel.text = "Examples of physical hazards include:"
                    Answer1.setTitle("dishonest employee", for: .normal)
                    Answer2.setTitle("building fire, oily rags, and a dishonest employee", for: .normal)
                    Answer3.setTitle("oily rags and a gas leak", for: .normal)
                    Answer4.setTitle("a building fire", for: .normal)
                    Answer4Correct = true
                case 3:
```



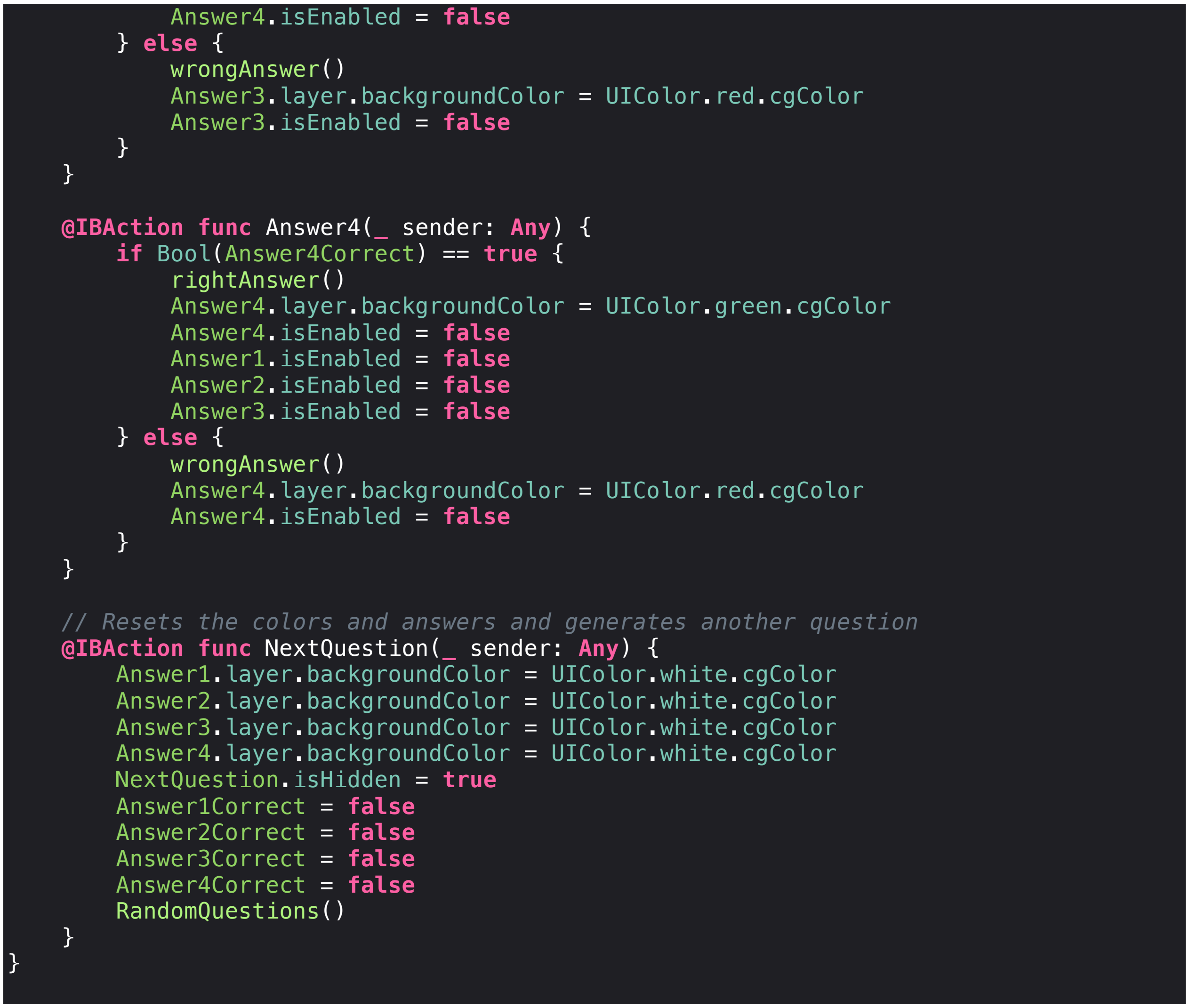
```
                    QuestionLabel.text = "Ken fell asleep while driving late at night.
He crossed the center line and hit a car approaching from the other direction. The
following losses occurred:"
                    Answer1.setTitle("the car that Ken hit was a total loss", for:
.normal)
                    Answer2.setTitle("Ken's car sustained $5,000 in damages", for:
.normal)
                    Answer3.setTitle("Ken incurred $5,000 in medical expenses", for:
.normal)
                    Answer4.setTitle("the driver of the other car suffered $30,000 in
bodily injuries", for: .normal)
                    Answer4Correct = true
                case 4:
                    QuestionLabel.text = "Which one of the following losses would be
covered under the medical payments coverage of the Homeowners 3 policy?"
                    Answer1.setTitle("injury to a resident employee at the insured's
home", for: .normal)
                    Answer2.setTitle("injury caused byan negligent operation of a
vehicle", for: .normal)
                    Answer3.setTitle("medical payments due to contagious disease",
for: .normal)
                    Answer4.setTitle("workers compensation medical payments", for:
.normal)
                    Answer1Correct = true
                case 5:
                    QuestionLabel.text = "A provision in disability insurance that
specifies the conditions under which the insured is automatically considered disabled
is:"
                    Answer1.setTitle("Investment spending does not change", for:
.normal)
                    Answer2.setTitle("Gross investment increases", for: .normal)
                    Answer3.setTitle("Investment demand decreases", for: .normal)
                    Answer4.setTitle("Net investment decreases", for: .normal)
                    Answer2Correct = true
                case 6:
                    QuestionLabel.text = "__________ is the likelihood that an event
will occur."
                    Answer1.setTitle("Peril", for: .normal)
                    Answer2.setTitle("Hazard", for: .normal)
                    Answer3.setTitle("Probability", for: .normal)
                    Answer4.setTitle("Risk", for: .normal)
                    Answer3Correct = true
                case 7:
                    QuestionLabel.text = "In what career would a person assess the
cost of replacing or repairing damaged property?"
                    Answer1.setTitle("agent", for: .normal)
                    Answer2.setTitle("appraiser", for: .normal)
                    Answer3.setTitle("claims adjustor", for: .normal)
                    Answer4.setTitle("risk manager", for: .normal)
                    Answer2Correct = true
                case 8:
                    QuestionLabel.text = "Defintion of Risk is the likelihood that
___"
                    Answer1.setTitle("a hazard will result in a mishap or loss", for:
.normal)
                    Answer2.setTitle("a hazard will result in a disaster", for:
.normal)
                    Answer3.setTitle("a hazard will result in a favorable outcome",
for: .normal)
                    Answer4.setTitle("a hazard will result in a change in command",
for: .normal)
```



```
                    Answer1Correct = true
                case 9:
                    QuestionLabel.text = "The process of detecting hazards and 
assessing the associated risk is called"
                    Answer1.setTitle("The right process", for: .normal)
                    Answer2.setTitle("Risk assessment", for: .normal)
                    Answer3.setTitle("Risk Management Process ", for: .normal)
                    Answer4.setTitle("None of the above", for: .normal)
                    Answer2Correct = true
                case 10:
                    QuestionLabel.text = "Operational Risk Management"
                    Answer1.setTitle("Is a decision making tool", for: .normal)
                    Answer2.setTitle("Increases ability to make informed decisions", 
for: .normal)
                    Answer3.setTitle("Reduces risks to acceptable levels", for: 
.normal)
                    Answer4.setTitle("All of the above", for: .normal)
                    Answer4Correct = true
                case 11:
                    QuestionLabel.text = "Which of the following is a principal of 
ORM?"
                    Answer1.setTitle("Make risk decisions at the right level", for: 
.normal)
                    Answer2.setTitle("Make risk decisions right away", for: .normal)
                    Answer3.setTitle("Make no decisions until after talking it over", 
for: .normal)
                    Answer4.setTitle("All of the above", for: .normal)
                    Answer1Correct = true
                case 12:
                    QuestionLabel.text = "Causes of risk include all of the following 
except"
                    Answer1.setTitle("New Technology", for: .normal)
                    Answer2.setTitle("Change \"The Mother of all Risk\"", for: 
.normal)
                    Answer3.setTitle("Environmental influences", for: .normal)
                    Answer4.setTitle("Chemical breakdown", for: .normal)
                    Answer1Correct = 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 Insurance and Risk Management! 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()
    }
}
```