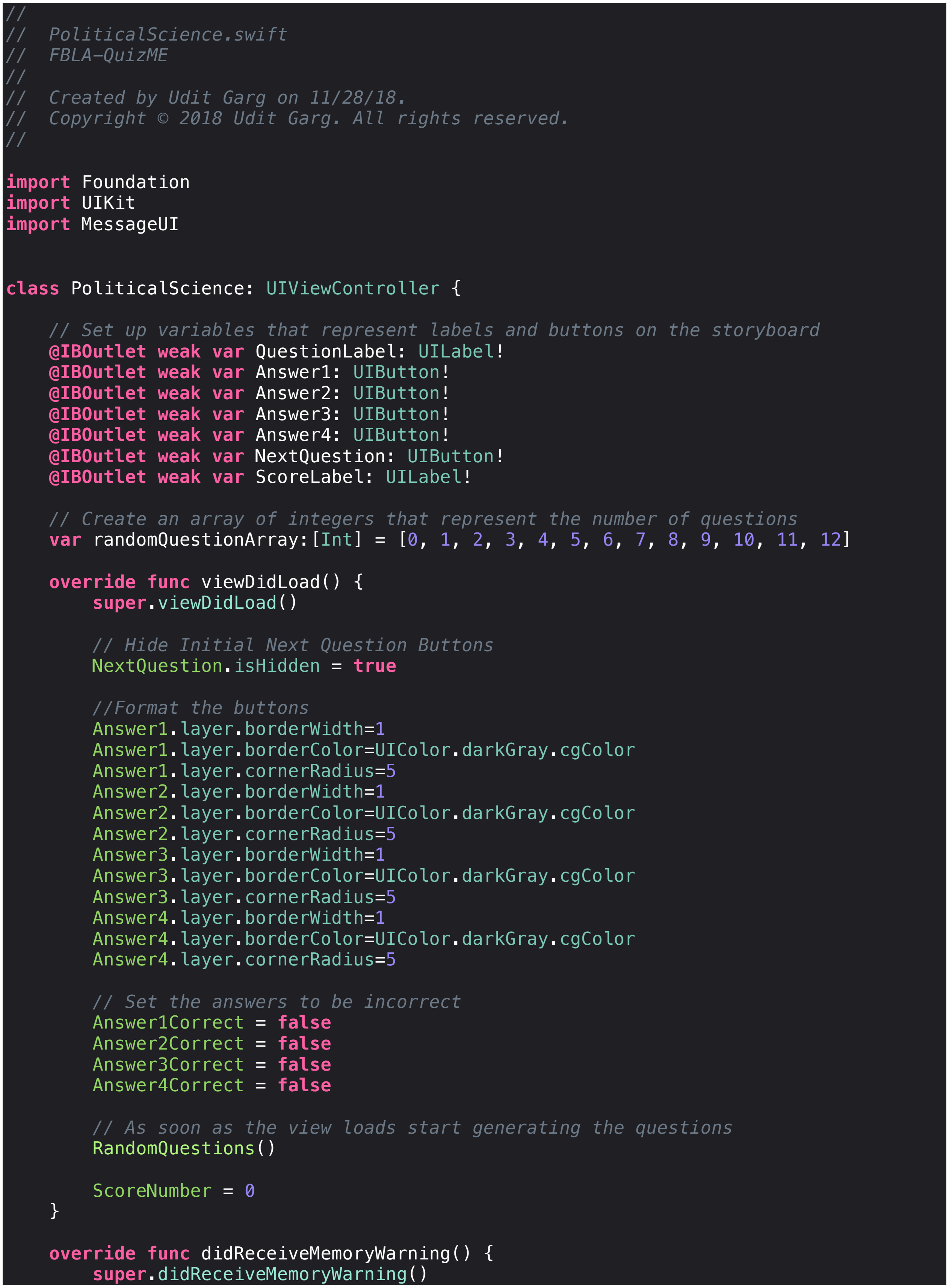
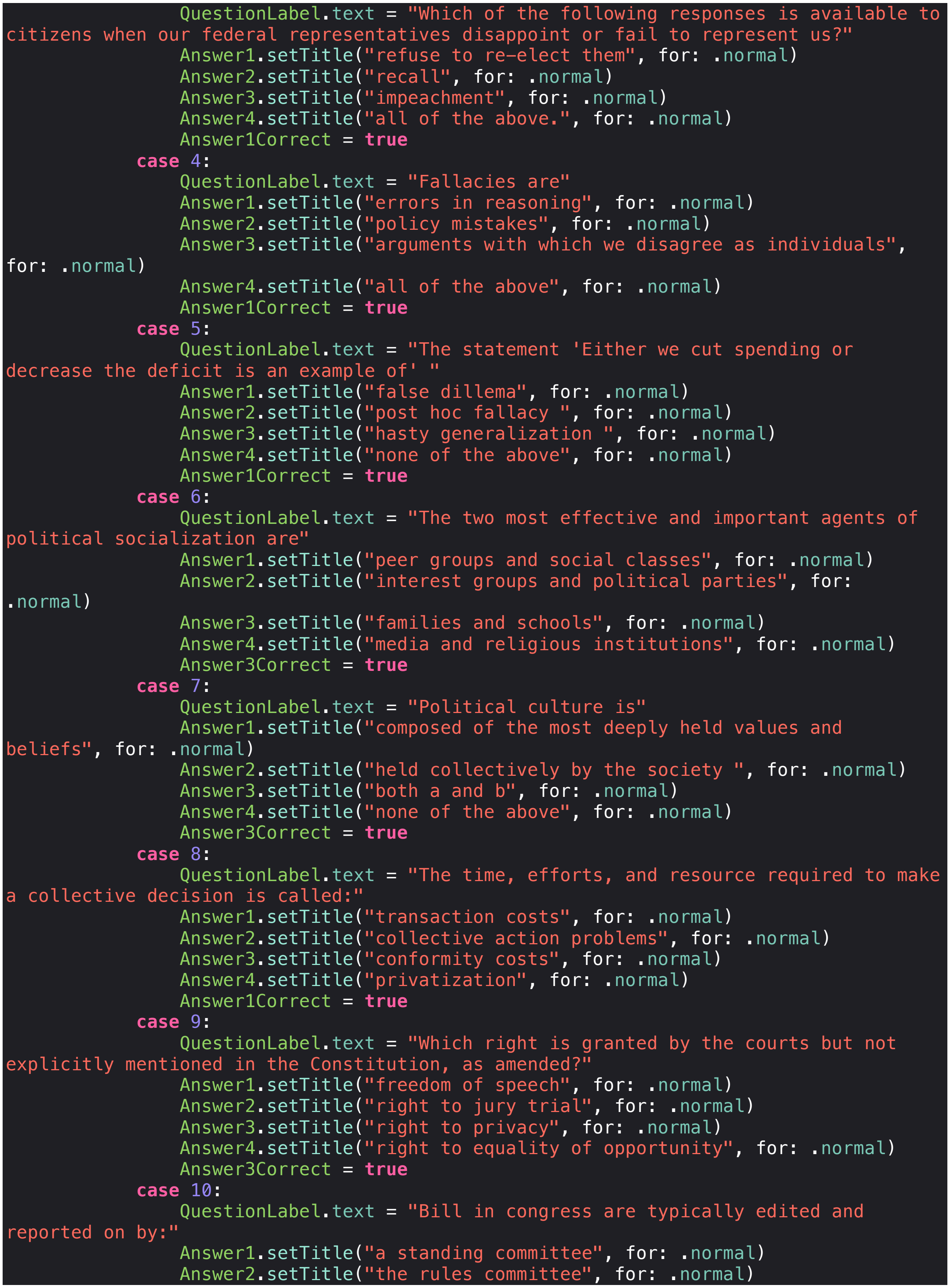
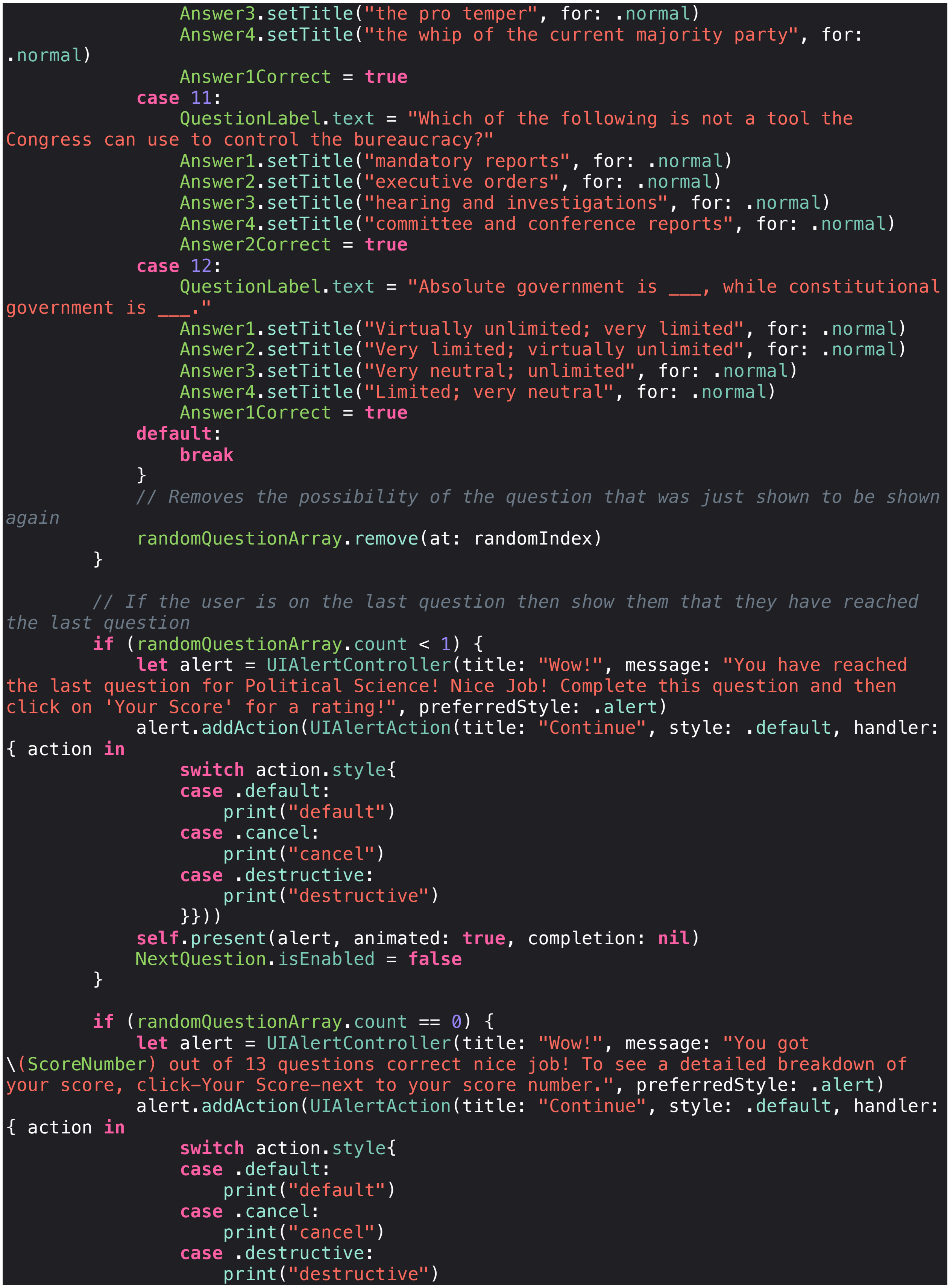


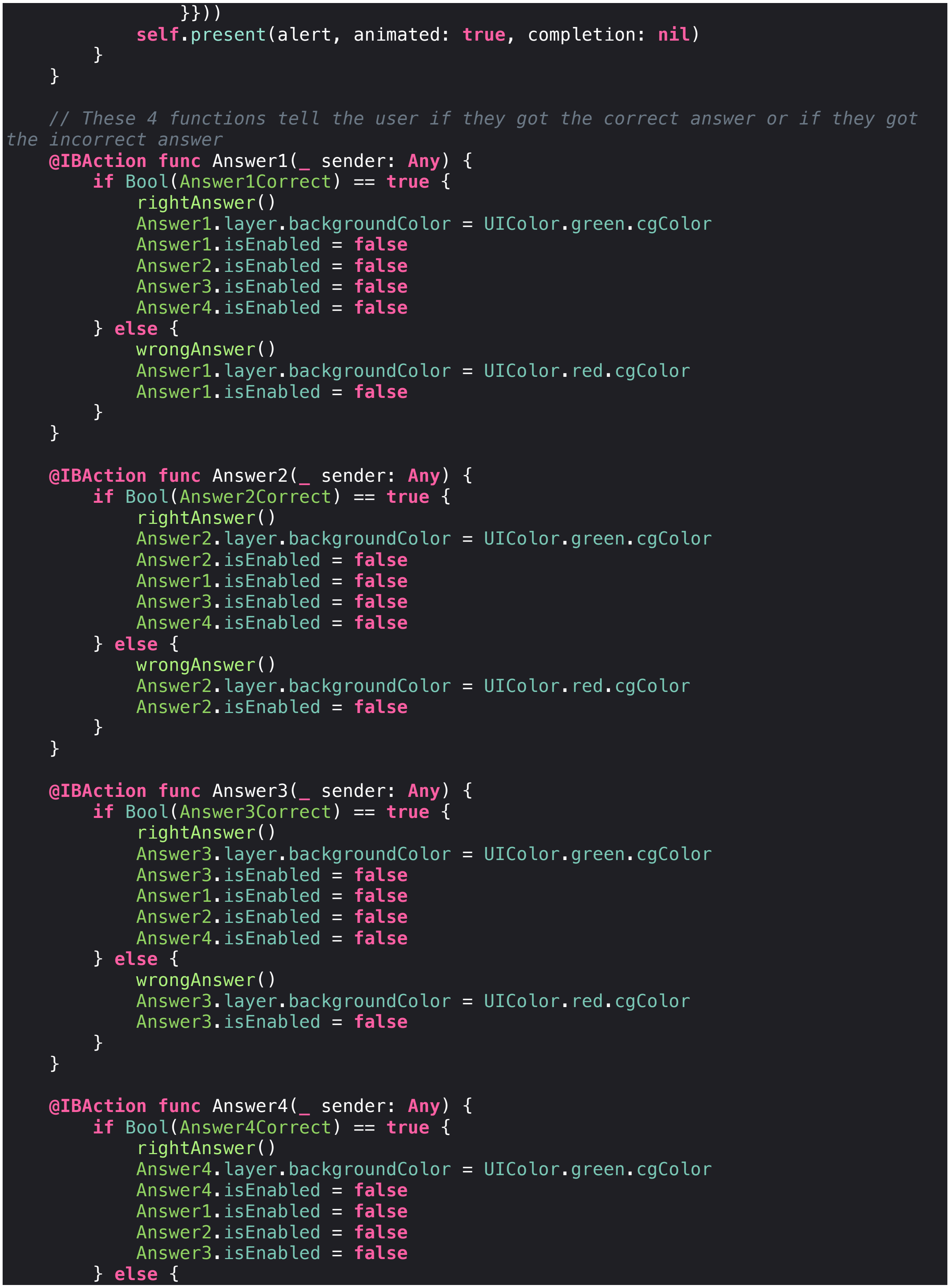
```swift
//
//  PoliticalScience.swift
//  FBLA-QuizME
//
//  Created by Udit Garg on 11/28/18.
//  Copyright © 2018 Udit Garg. All rights reserved.
//

import Foundation
import UIKit
import MessageUI


class PoliticalScience: 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 concepts is not included
in the definition of politics?"
                Answer1.setTitle("power", for: .normal)
                Answer2.setTitle("popular Sovereignty", for: .normal)
                Answer3.setTitle("allocation/distribution", for: .normal)
                Answer4.setTitle("society/community/public", for: .normal)
                Answer2Correct = true
            case 1:
                QuestionLabel.text = "The principle that citizens in a democracy are
protected from the government interference in the exercise of several basic freedoms
is known as "
                Answer1.setTitle("popular sovereignty", for: .normal)
                Answer2.setTitle("political equality", for: .normal)
                Answer3.setTitle("political liberty", for: .normal)
                Answer4.setTitle("democracy", for: .normal)
                Answer3Correct = true
            case 2:
                QuestionLabel.text = "What is it called when we expect our
representatives to follow the policy wishes of the voters back home, regardless of
their own personal policy preferences? "
                Answer1.setTitle("direct democracy", for: .normal)
                Answer2.setTitle("trustee model of representation", for: .normal)
                Answer3.setTitle("delegate model of representation", for: .normal)
                Answer4.setTitle("none of the above", for: .normal)
                Answer3Correct = true
            case 3:
```



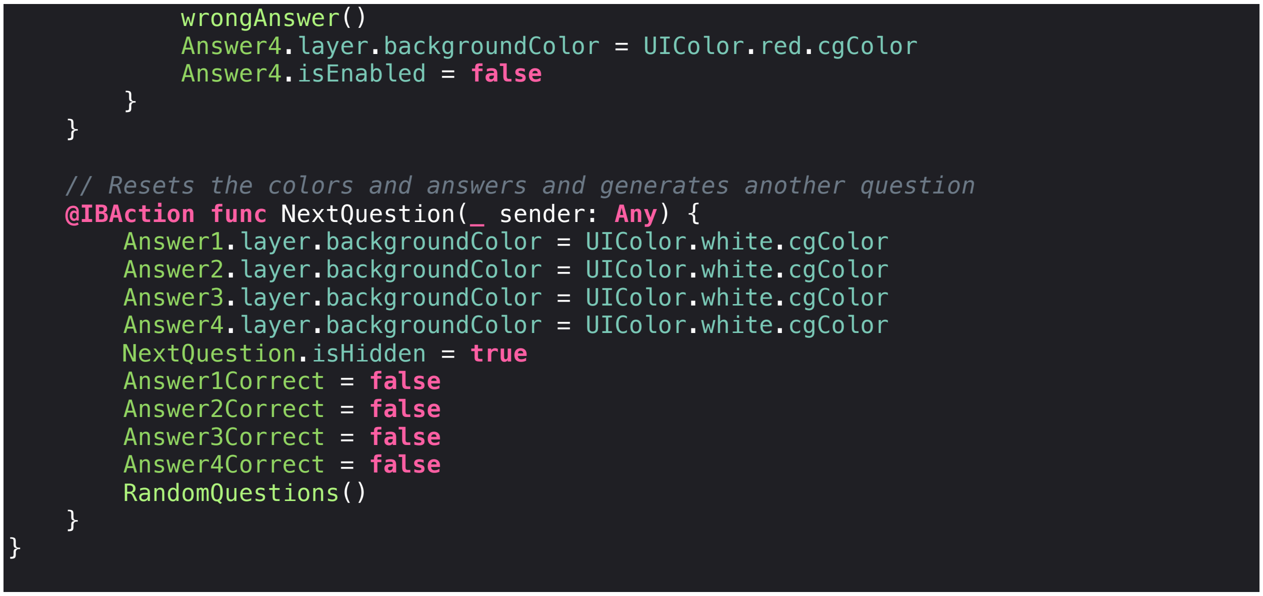
```
                QuestionLabel.text = "Which of the following responses is available to
citizens when our federal representatives disappoint or fail to represent us?"
                Answer1.setTitle("refuse to re-elect them", for: .normal)
                Answer2.setTitle("recall", for: .normal)
                Answer3.setTitle("impeachment", for: .normal)
                Answer4.setTitle("all of the above.", for: .normal)
                Answer1Correct = true
            case 4:
                QuestionLabel.text = "Fallacies are"
                Answer1.setTitle("errors in reasoning", for: .normal)
                Answer2.setTitle("policy mistakes", for: .normal)
                Answer3.setTitle("arguments with which we disagree as individuals",
for: .normal)
                Answer4.setTitle("all of the above", for: .normal)
                Answer1Correct = true
            case 5:
                QuestionLabel.text = "The statement 'Either we cut spending or
decrease the deficit is an example of' "
                Answer1.setTitle("false dillema", for: .normal)
                Answer2.setTitle("post hoc fallacy ", for: .normal)
                Answer3.setTitle("hasty generalization ", for: .normal)
                Answer4.setTitle("none of the above", for: .normal)
                Answer1Correct = true
            case 6:
                QuestionLabel.text = "The two most effective and important agents of
political socialization are"
                Answer1.setTitle("peer groups and social classes", for: .normal)
                Answer2.setTitle("interest groups and political parties", for:
.normal)
                Answer3.setTitle("families and schools", for: .normal)
                Answer4.setTitle("media and religious institutions", for: .normal)
                Answer3Correct = true
            case 7:
                QuestionLabel.text = "Political culture is"
                Answer1.setTitle("composed of the most deeply held values and
beliefs", for: .normal)
                Answer2.setTitle("held collectively by the society ", for: .normal)
                Answer3.setTitle("both a and b", for: .normal)
                Answer4.setTitle("none of the above", for: .normal)
                Answer3Correct = true
            case 8:
                QuestionLabel.text = "The time, efforts, and resource required to make
a collective decision is called:"
                Answer1.setTitle("transaction costs", for: .normal)
                Answer2.setTitle("collective action problems", for: .normal)
                Answer3.setTitle("conformity costs", for: .normal)
                Answer4.setTitle("privatization", for: .normal)
                Answer1Correct = true
            case 9:
                QuestionLabel.text = "Which right is granted by the courts but not
explicitly mentioned in the Constitution, as amended?"
                Answer1.setTitle("freedom of speech", for: .normal)
                Answer2.setTitle("right to jury trial", for: .normal)
                Answer3.setTitle("right to privacy", for: .normal)
                Answer4.setTitle("right to equality of opportunity", for: .normal)
                Answer3Correct = true
            case 10:
                QuestionLabel.text = "Bill in congress are typically edited and
reported on by:"
                Answer1.setTitle("a standing committee", for: .normal)
                Answer2.setTitle("the rules committee", for: .normal)
```



```
                Answer3.setTitle("the pro temper", for: .normal)
                Answer4.setTitle("the whip of the current majority party", for:
.normal)
                Answer1Correct = true
            case 11:
                QuestionLabel.text = "Which of the following is not a tool the
Congress can use to control the bureaucracy?"
                Answer1.setTitle("mandatory reports", for: .normal)
                Answer2.setTitle("executive orders", for: .normal)
                Answer3.setTitle("hearing and investigations", for: .normal)
                Answer4.setTitle("committee and conference reports", for: .normal)
                Answer2Correct = true
            case 12:
                QuestionLabel.text = "Absolute government is ___, while constitutional
government is ___."
                Answer1.setTitle("Virtually unlimited; very limited", for: .normal)
                Answer2.setTitle("Very limited; virtually unlimited", for: .normal)
                Answer3.setTitle("Very neutral; unlimited", for: .normal)
                Answer4.setTitle("Limited; very neutral", 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 Political Science! 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")
```



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