//  
// ViewController.swift  
// customFoodTable  
//  
// Created by MacStudent on 2019-10-30.  
// Copyright © 2019 MacStudent. All rights reserved.  
//  
  
import UIKit  
  
class ViewController: UIViewController, UITableViewDataSource, UITableViewDelegate {  
   
   
 @IBOutlet weak var tableView: UITableView!  
 var foods: [String]?  
 var calories: [Int]?  
 //using tuple, can get the item name and calories -array  
 var foodData: [(name: String, calory:Int)]?  
 override func viewDidLoad() {  
 super.viewDidLoad()  
 // Do any additional setup after loading the view.  
 //one of the ways to add delegate and datasource is written below   
 //tableView.delegate = self  
 //tableView.datasource = self  
 foods = ["Apple", "Banana", "Burger", "Fries", "Orange", "Pizza"]  
 calories = [50, 60, 900, 600, 30, 700]  
 //initializer- if we are using tuple  
 foodData = [("Apple", 50), ("Banana", 60), ("Burger", 900), ("Fries", 30), ("Orange", 30), ("Pizza", 700)]  
 //using register to reuse the cell that is created - foodTableViewCell  
 tableView.register(FoodTableViewCell.self, forCellReuseIdentifier: "food cell")  
 }  
 func tableView(\_ tableView: UITableView, numberOfRowsInSection section: Int) -> Int {  
 return foods?.count ?? 0  
 }  
   
 func tableView(\_ tableView: UITableView, cellForRowAt indexPath: IndexPath) -> UITableViewCell {  
 guard foods != nil else {return UITableViewCell()}  
// let cell = UITableViewCell(style: .value1, reuseIdentifier: "")  
// let foodName = foods![indexPath.row]  
// cell.textLabel?.text = foodName  
// cell.imageView?.image = UIImage(named: foodName)  
// cell.detailTextLabel?.text = "calories: \(calories![indexPath.row])"  
// return cell  
 // displaying cell after using register  
 let foodName = foodData![indexPath.row].name  
 let foodCalory = foodData![indexPath.row].calory  
 let cell = tableView.dequeueReusableCell(withIdentifier: "food cell") as! FoodTableViewCell  
 cell.setName(name: foodName, calories: foodCalory, image: foodName)  
 return cell  
 }  
 //by clicking on the items - alerts  
 func tableView(\_ tableView: UITableView, didSelectRowAt indexPath: IndexPath) {  
 let foodName = foodData![indexPath.row].name  
 let alertController = UIAlertController(title: "Food Selected", message: "You have selected \(foodName)", preferredStyle: .alert)  
 //constant and instance from  
 let okAction = UIAlertAction(title: "OK", style: .cancel, handler: nil)  
 // handler converted to clouser by using enter key automatically  
 let printAction = UIAlertAction(title: "Print", style: .default) { (action) in  
 print("Selected food is \(foodName)")  
 }  
 alertController.addAction(okAction)  
 alertController.addAction(printAction)  
   
 self.present(alertController, animated: true, completion: nil  
 )  
   
 }  
}