

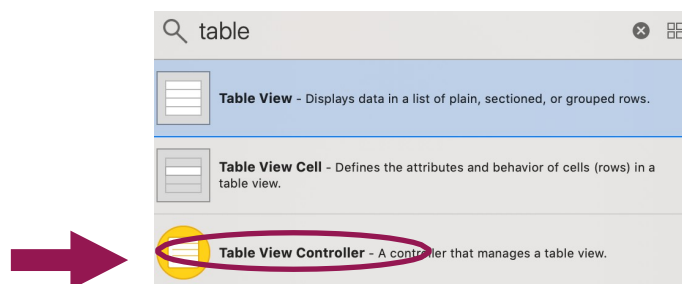
## Nivel Bellota

### 4. Tema: Table View Controller y Search Bar

Dentro de nuestra librería de objetos podremos encontrar diferentes opciones, una de ellas es un:

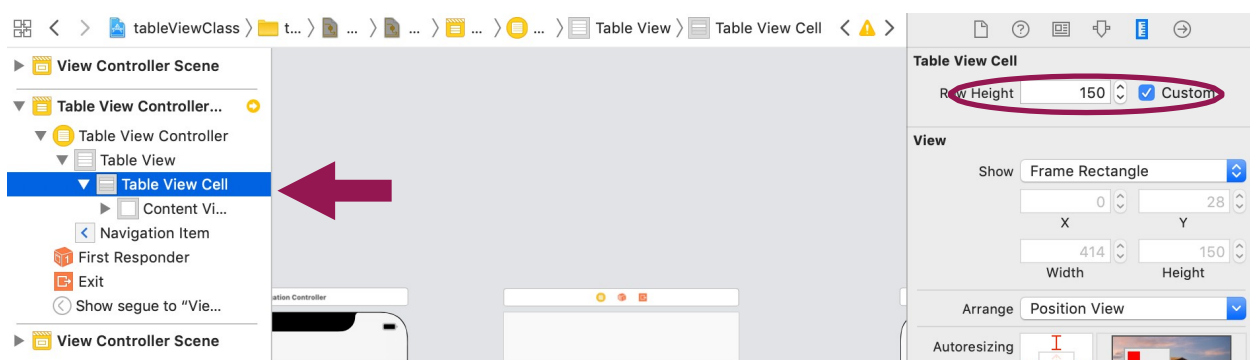
# Table View Controller

- 1.- Creamos un nuevo proyecto en *XCODE*, *Single View App*.
- 2.- Nos dirigimos a el main story board.
- 3.- En la biblioteca de objetos, agregamos un *Table View Controller*

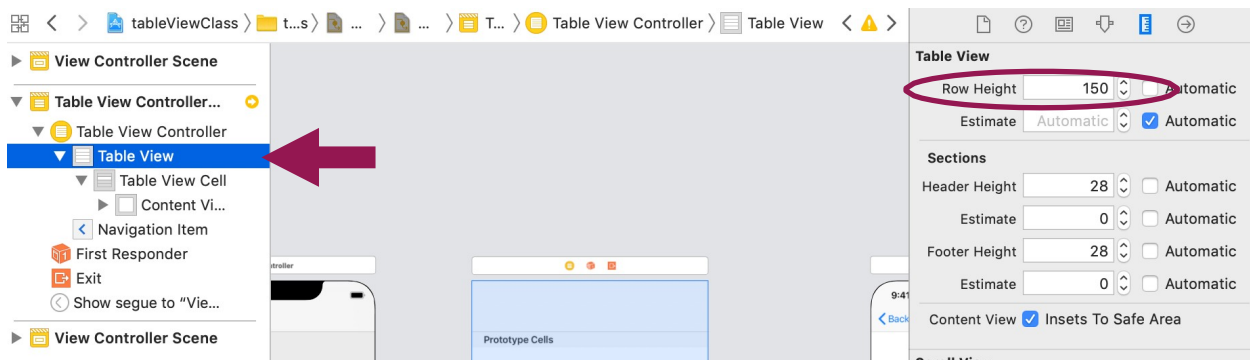


- 4.- En *Table View Controller*, modificar el tamaño de las celdas para una mejor visualización del usuario.

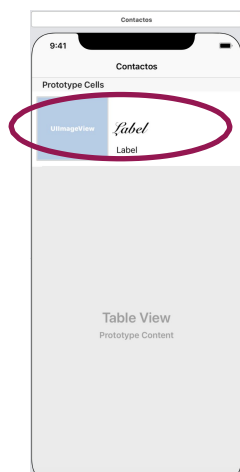
Seleccionamos *Table View Cell*, después en el área de las propiedades nos colocamos en el cuarto elemento  modificamos la parte de: *Row Height* con 150.



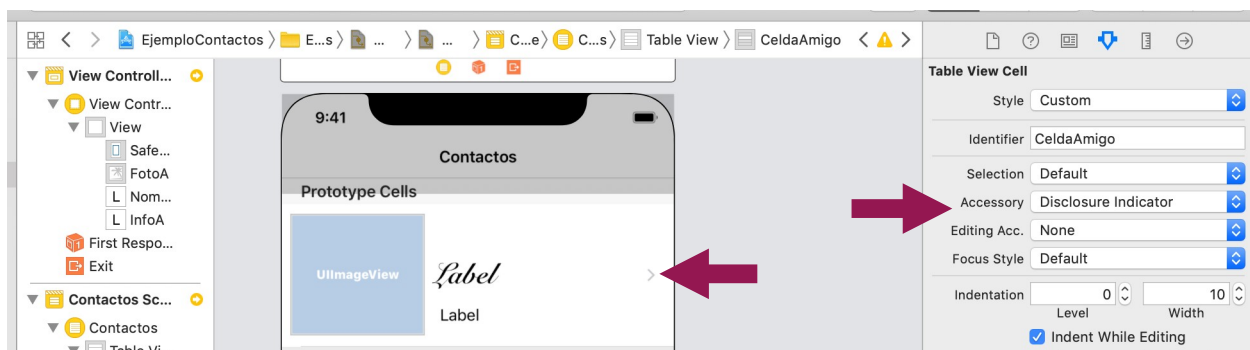
Posteriormente seleccionamos Table View, igualmente en la parte de las propiedades del objeto del lado derecho en: Row Height colocamos 150.



5.- En la vista del Table View Controller agregar en la celda: 1 Image y 2 Label. (Cambiar las propiedades de los objetos a su gusto).



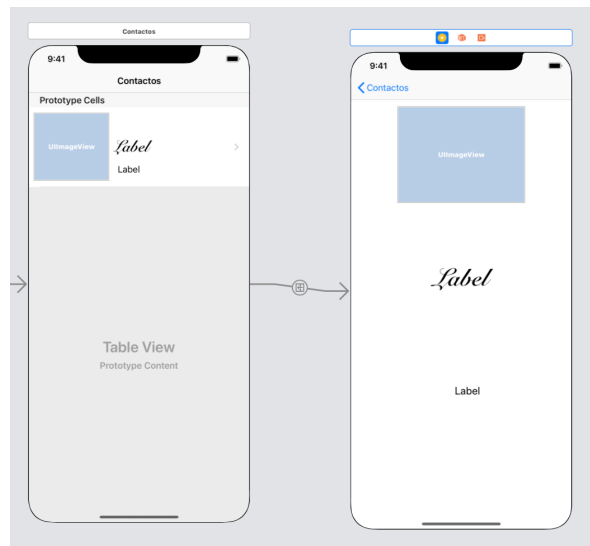
6.- En las propiedades de la celda, podemos encontrar en la cuarta pestaña la opción Accessory, donde pondremos la opción: **Disclosure Indicator**.



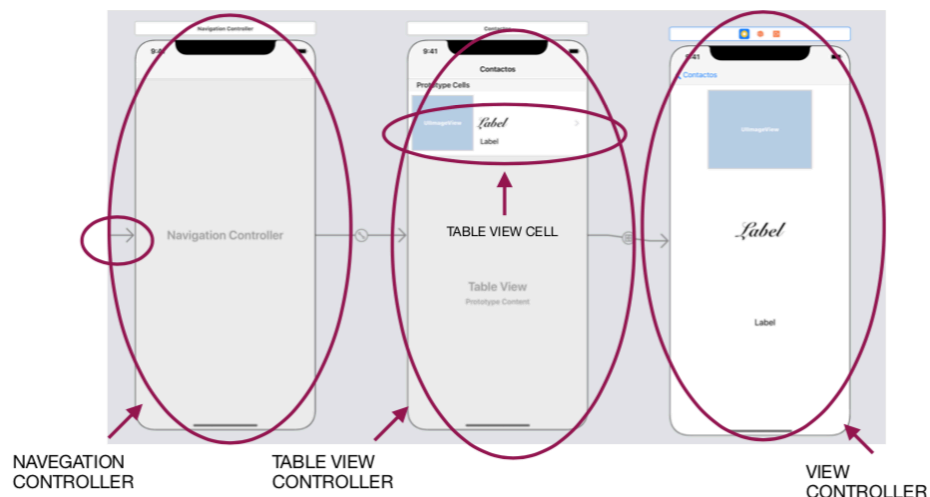
7.- Agregar al View Controller un objeto tipo Image y 2 label.



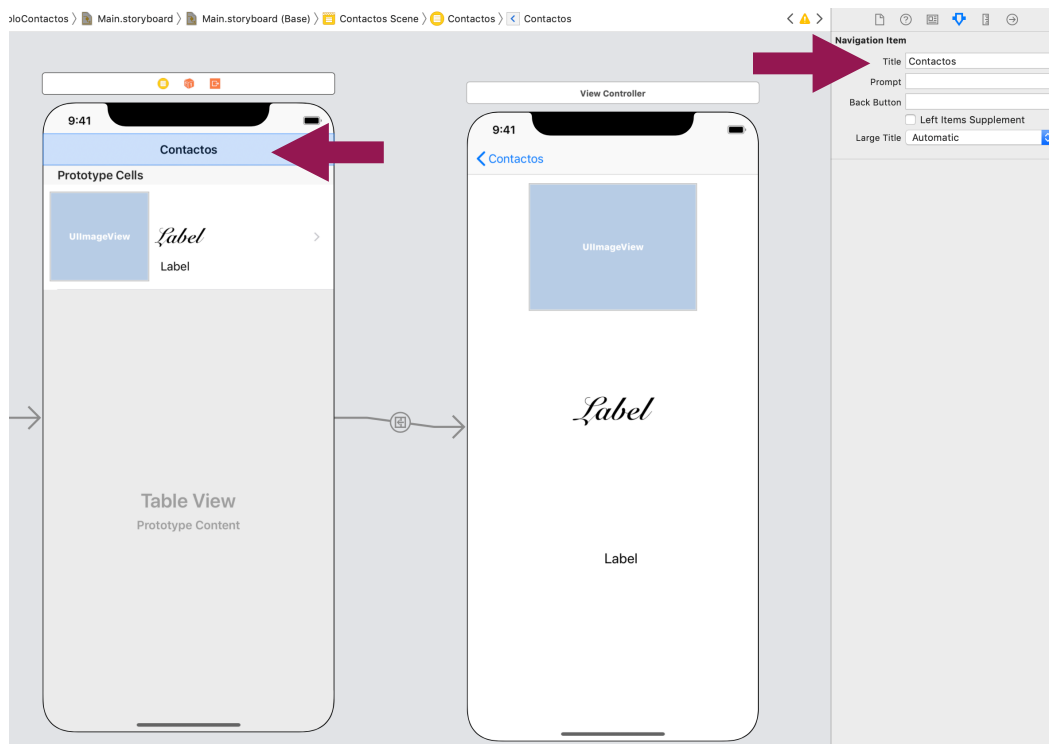
8.- Conectar la celda del Table View Controller (control+touch pad) que se conecte con el View Controller.



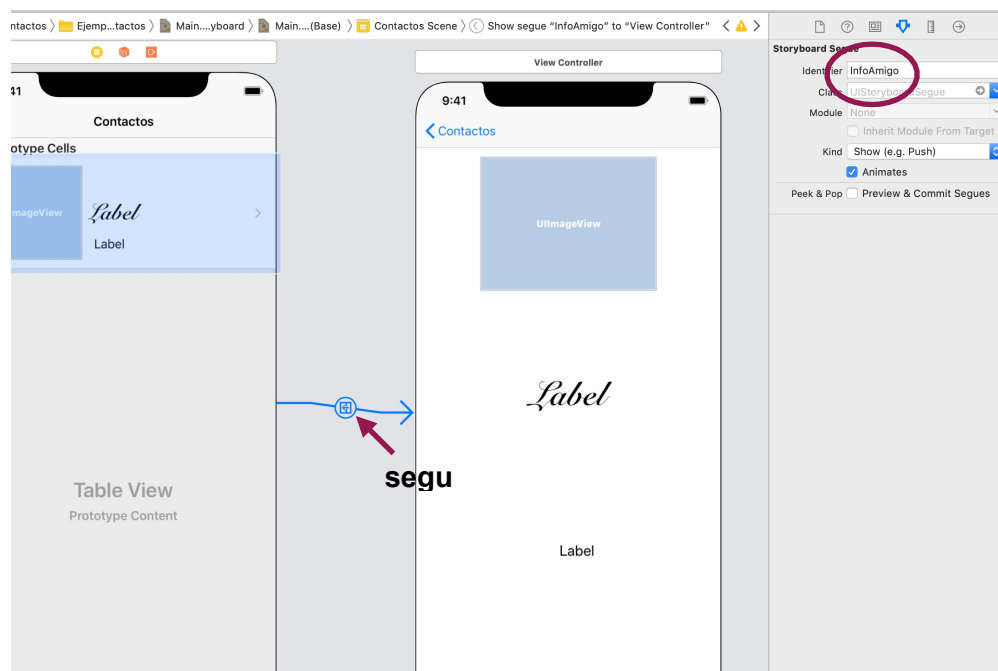
9.- Agregamos un Navigation Controller para poder regresar a la lista y colocamos la flechita al Navigation Controller, es la que indica dónde iniciará la compilación.



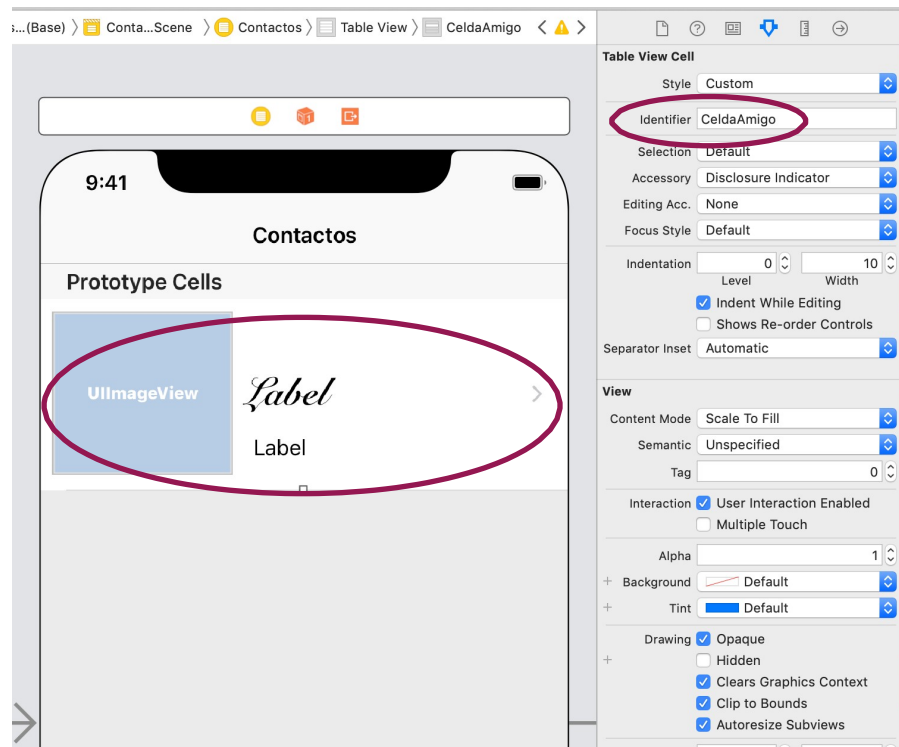
10.- Podemos cambiar el nombre de la lista en las propiedades del Navigation Controller.



11.- Ahora colocaremos dos identificadores, el primero en el segue (flechita) que conecta el Table View con el View Controller, mi identificador se llama: "infoAmigo".



El segundo será poner el identificador a la celda dentro del Table View, la seleccionamos y en las propiedades en identificar: "CeldaAmigo".



12.- Crear 2 nuevas clases, la primera será de tipo `UITableViewController`, `UIViewController` y la otra será `UITableViewCell`, la primera la conectaremos con el `Table View Controller`, la segunda a la celda que se encuentra dentro de este.

### Creación:

Class:

Subclass of:

☐ Also create XIB file

Language:  Whether to create a XIB file with the same name

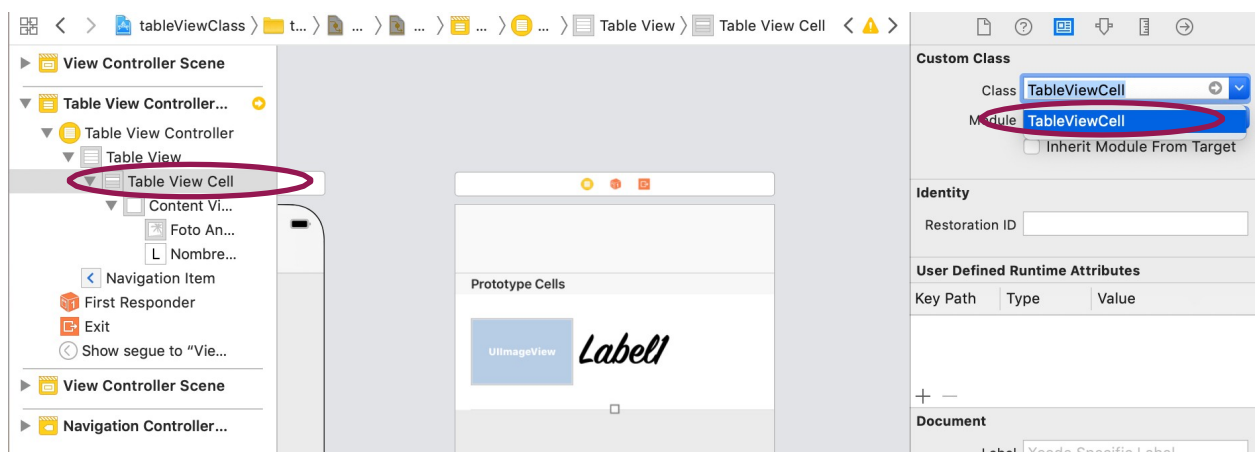
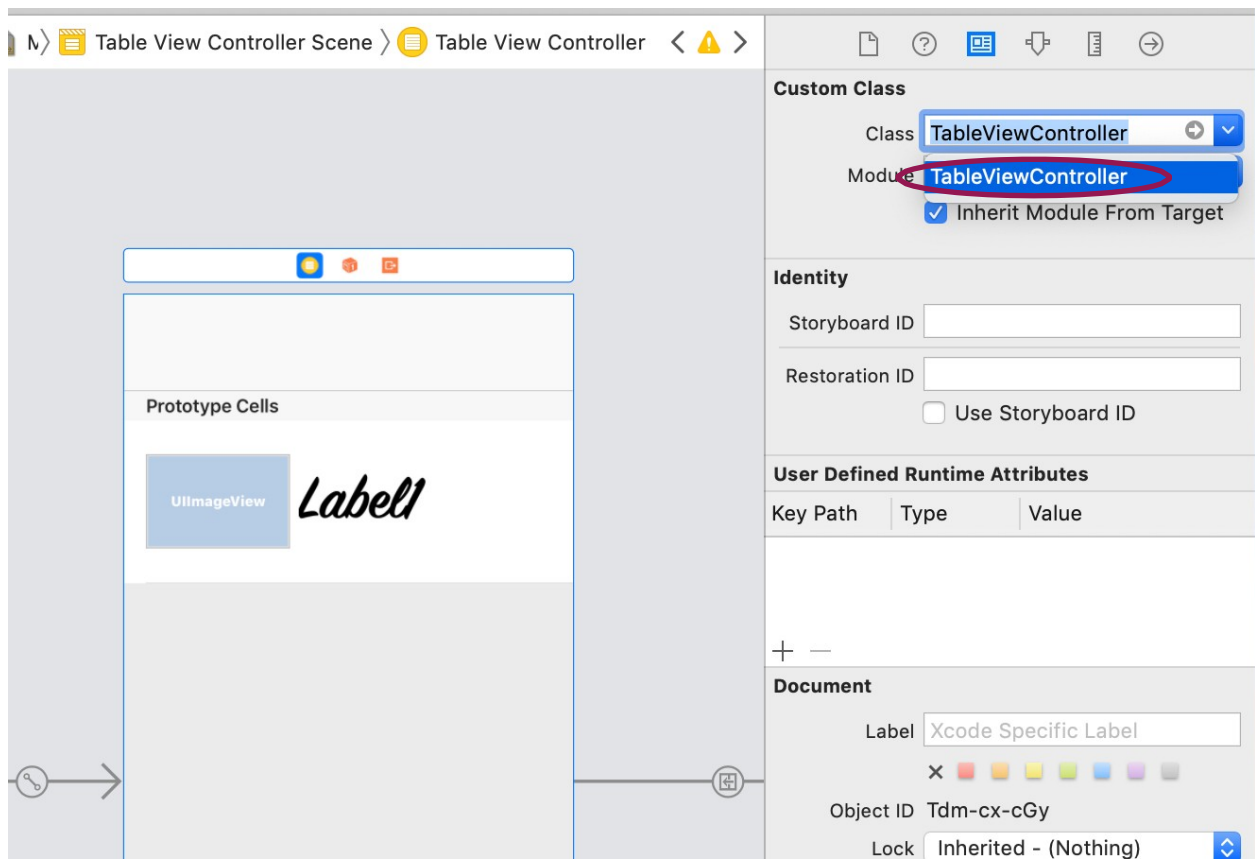
Class:

Subclass of:

☐ Also create XIB file

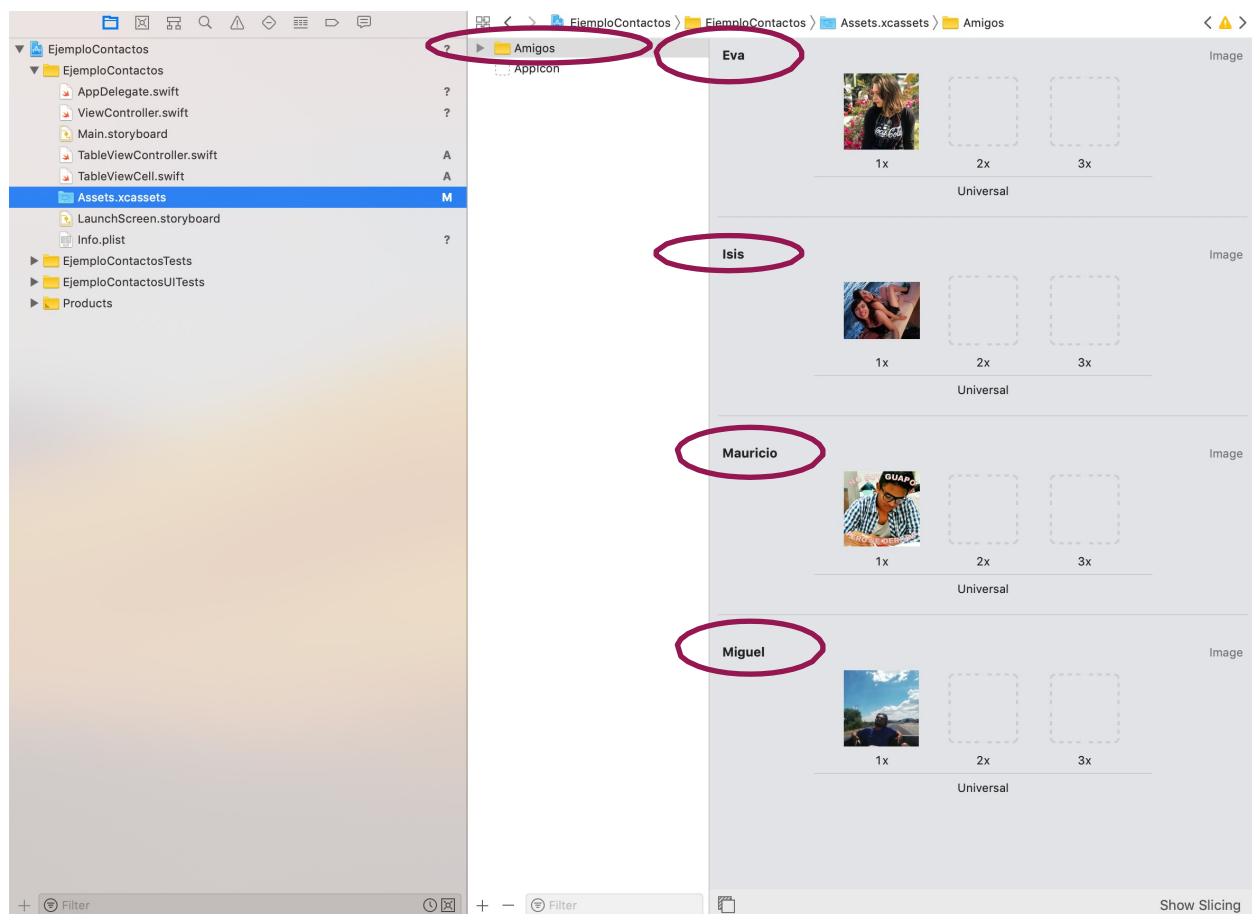
Language:

## Conexión:



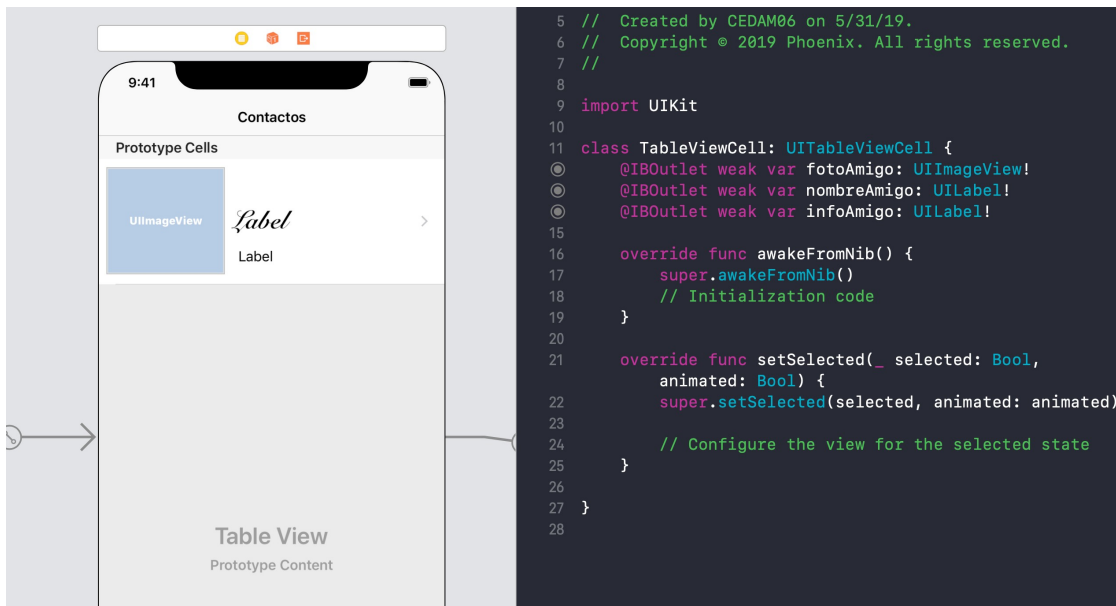
13.- Agregamos una carpeta con las imágenes de cada persona, únicamente tenemos que crear la carpeta en el Desktop y la jalamos al proyecto en la parte de los archivos: **Assets.xcassets**.

Es importante poner los nombres de las imágenes de manera correcta ya que posteriormente haremos referencia a ellos por medio de su nombre.



14.- Dentro de la clase *Table View Cell*, agregaremos 3 objetos:

- A. *fotoAmigo*, dónde se encontrará la imagen de la persona.
- B. *nombreAmigo*, el label dónde se pintará el nombre de cada persona.
- C. *infoAmigo*, el label dónde se colocará su información de contacto.

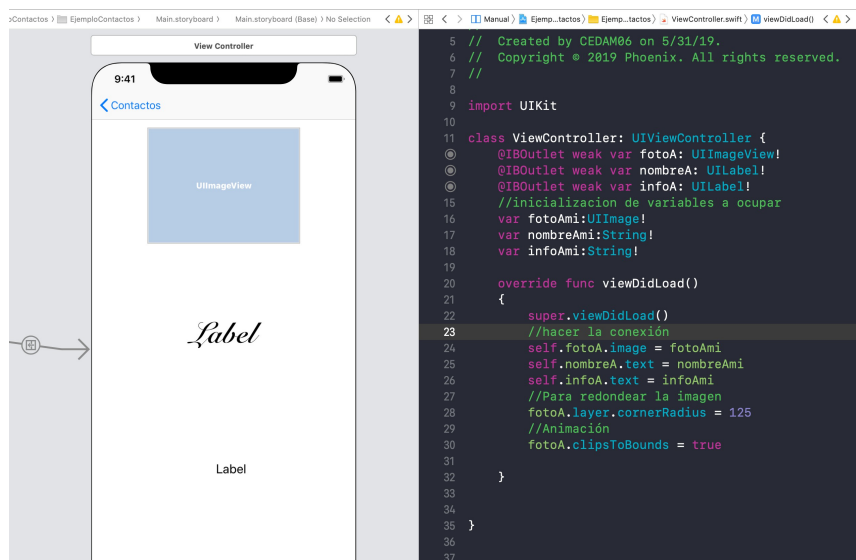


15.- Nos dirigimos al *View Controller* y conectamos los objetos anteriores para la presentación de cada persona.

- I. *fotoA*, dónde se encontrará la imagen de la persona.
- II. *nombreA*, el label dónde se pintará el nombre de cada persona.
- III. *infoA*, el label dónde se colocará su información de contacto.

16.- Una vez conectados los objetos (outlet) hay que realizar la inicialización de otras variables donde se guardarán los objetos de las vistas.

17.- Una vez terminado esto, en la función *viewDidLoad()* haremos la conexión entre los objetos de la vista con las variables creadas en el código.





18.- Nos dirigimos a la clase Table View Controller, entre la librería import UIKit y la clase TableViewController colocaremos una nueva clase, llamada Amigo, donde sus propiedades tendrán en consideración: nombreA, imagenA, infoA.

Siempre hay que inicializar las propiedades y hacer la concesión entre las propiedades declaradas y las inicializadas.

```
8
9 import UIKit
10 //Clase del objeto
11 class Amigo
12 {
13     //Declarar las propiedades o atributos
14     let nombreA : String
15     let imagenA : UIImage
16     let infoA : String
17     //inicializacion
18     init(from nA: String, from iA: String, from infA:
        String)
19     {
20         self.nombreA = nA
21         self.imagenA = UIImage(named: iA)!
22         self.infoA = infA
23     }
24
25
26 }
27 //clase de la vista
28 class TableViewController: UITableViewController
```

19.- Una vez terminado esto, dentro de la clase TableViewController crearemos un arreglo de tipo de clase Amigo, colocando todas las propiedades que tiene cada amigo y mostrará cada celda en la vista.

```
27 //clase de la vista
28 class TableViewController: UITableViewController
29 {
30     let amigos : [Amigo] =
31     [
32         Amigo(from: "Isis Franco", from: "Isis",
            from: "Tel: 5539904099\nCorreo:
            isisfranco@gmail.com"),
33         Amigo(from: "Miguel Gallardo", from:
            "Miguel", from: "Tel:
            5539904099\nCorreo:
            mikeygallardo@gmail.com"),
34         Amigo(from: "Mauricio Gómez", from:
            "Mauricio", from: "Tel:
            5539904099\nCorreo: maug@gmail.com"),
35         Amigo(from: "Eva Flores", from: "Eva",
            from: "Tel: 5539904099\nCorreo:
            evaflores@gmail.com")
36     ]
37
38
39     override func viewDidLoad()
40     {
41         super.viewDidLoad()
42     }
43
44     // MARK: - Table view data source
45 }
```

20.- Las siguientes 3 funciones que agregaremos son necesarias para las propiedades de las celdas dentro de la vista.

```
override func viewDidLoad()
{
    super.viewDidLoad()

    // MARK: - Table view data source

    override func numberOfSections(in tableView: UITableView) -> Int
    {
        // #warning Incomplete implementation, return the number of sections
        return 1
    }

    override func tableView(_ tableView: UITableView, numberOfRowsInSectionSection section: Int) -> Int {
        // #warning Incomplete implementation, return the number of rows
        return amigos.count
    }
}
```

21.- Ahora colocaremos la siguiente función, que nos permitirá acceder a los objetos creados en el Table View Cell para hacer una copia de los objetos creados en el Table View Controller.

- La propiedad `dequeueReusableCell`, nos permitirá acceder a la dirección en donde se encuentra el código de la celda.
- Con ayuda del identificador que ya colocamos anteriormente, el código encontrará el camino hasta la otra clase de la celda.
- La propiedad `as!` Realizará una copia
- `[indexPath.row]` será el identificador dentro del arreglo `amigos` para cada una de las personas que colocamos dentro del arreglo.
- Las variables que igualarás serán las que creaste primero en el Table View Cell y al final irán las propiedades creadas dentro de la clase `Amigo` en el Table View Controller.

```
//Pasar los objetos de la clase TableViewController a la clase TableViewCell
override func tableView(_ tableView: UITableView, cellForRowAt indexPath: IndexPath) -> UITableViewCell
{
    //se busca la dirección de la clase TableViewCell
    let celda = tableView.dequeueReusableCell(withIdentifier: "CeldaAmigo") as! TableViewCell
    //Se igualan variables de ambas clases
    celda.nombreAmigo.text = amigos[indexPath.row].nombreA
    celda.fotoAmigo.image = amigos[indexPath.row].imagenA
    celda.infoAmigo.text = amigos[indexPath.row].infoA

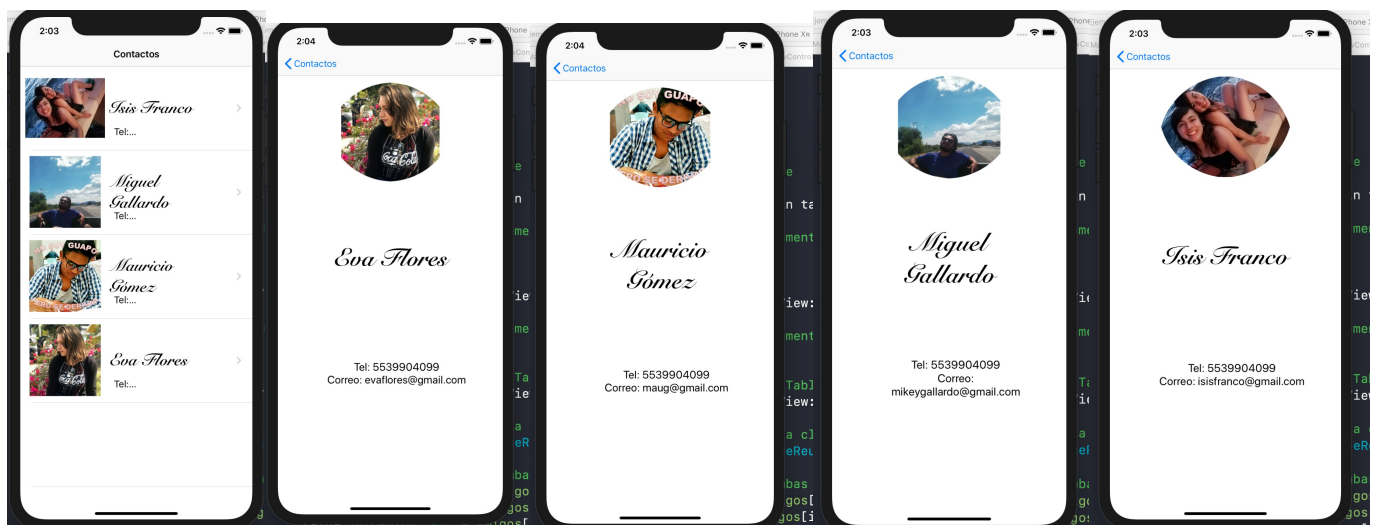
    return celda
}
```

## 22.- Realizaremos algo parecido pero ahora para colocar la información creada en el Table View Controller al View Controller.

- El primer if se refiere al identificador del segue que ya colocamos anteriormente
- El segundo if da la condición de que depende que tipo de celda se eligió para mostrar la información.
- Realizamos después el mismo codeo para la conexión de variables del View Controller y del Table View Controller.

```
69 //Pasar los objetos de la clase TableViewController a la clase ViewController
70
71 override func prepare(for segue: UIStoryboardSegue, sender: Any?)
72 {
73     if segue.identifier == "InfoAmigo"
74     {
75         if let indexPath = self.tableView.indexPathForSelectedRow
76         {
77             let destino = segue.destination as! ViewController
78             destino.fotoAmi = self.amigos[indexPath.row].imagenA
79             destino.nombreAmi = self.amigos[indexPath.row].nombreA
80             destino.infoAmi = self.amigos[indexPath.row].infoA
81         }
82     }
83 }
84
```

## 23.- Finalmente Los Resultados:



**Actividad 4:**

- Con lo aprendido en la sesión.

Agrega un Table View Controller a la app de tu tienda Virtual.

Fecha de entrega: 3 Octubre 2019