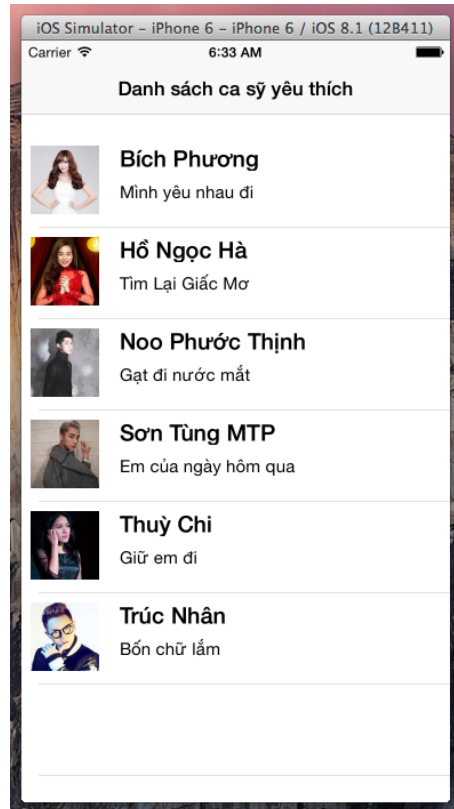


BÀI LAB : TÙY BIẾN TABLEVIEWCELL

Yêu cầu :

- Trong bài lab này giúp chúng ta hiểu được cách tùy biến, chỉnh sửa 1 cell trong UITableViewCell. Căn chỉnh hình , tiêu đề, nội dung, chiều cao của 1 cell...
- Tùy biến ô trong table view, thay vì sử dụng giao diện mặc định của table view cell, lần này chúng ta sẽ xây dựng giao diện cho riêng mình.

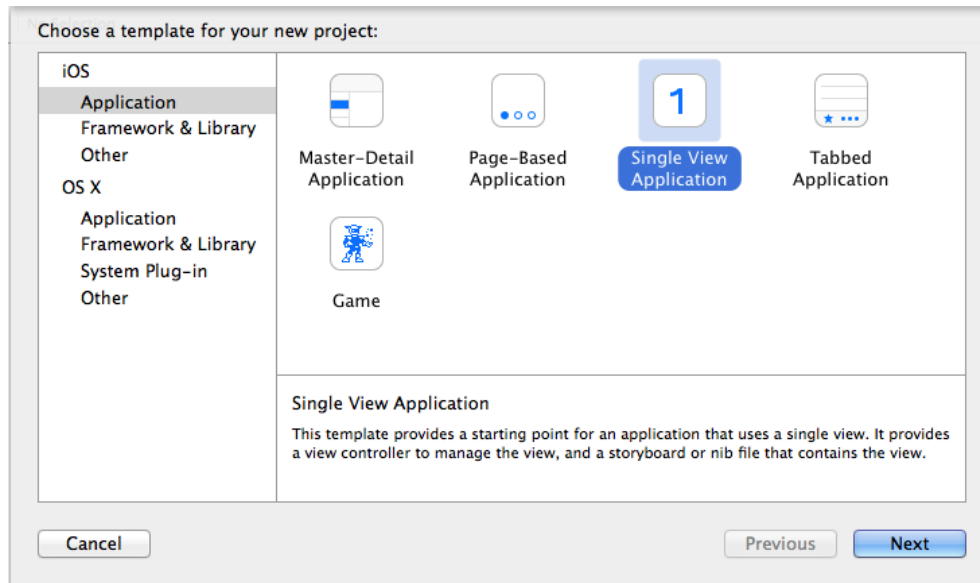
Kết quả đạt được :



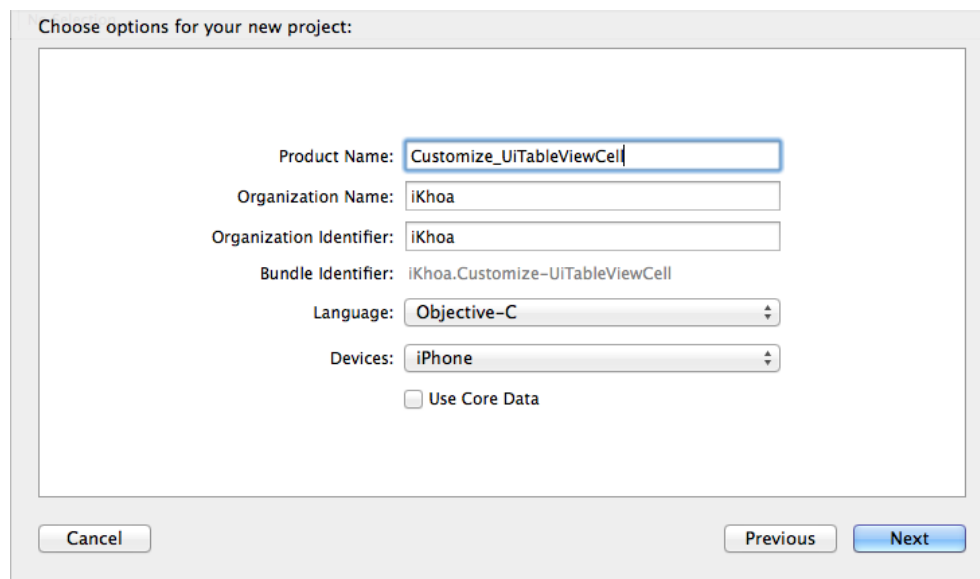
TẠO TABLEVIEW TRONG VIEWCONTROLLER

1. Tiến hành tạo project mới

- Tạo project Single View Application

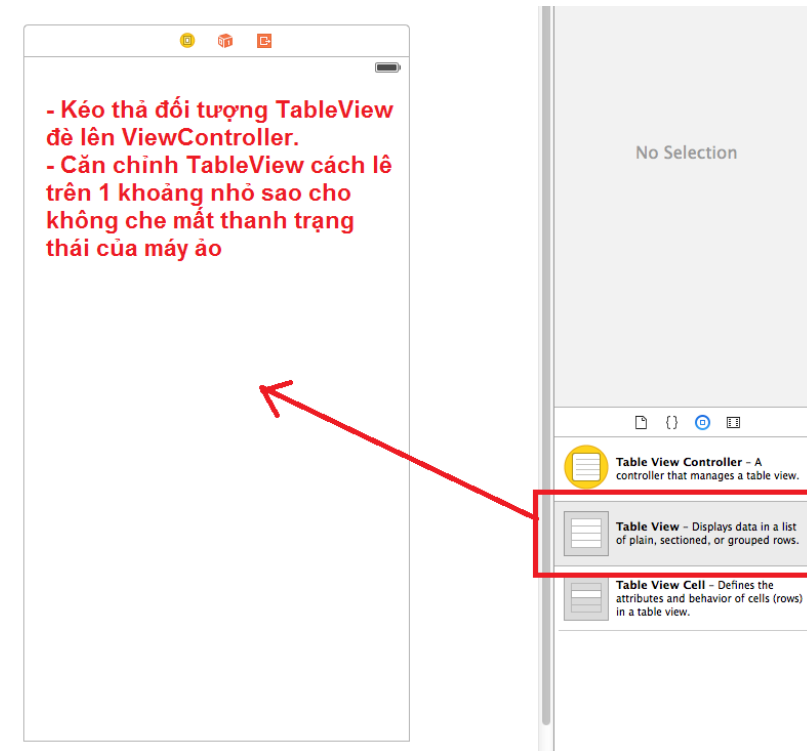


- Đặt tên Customize_UITableViewCell

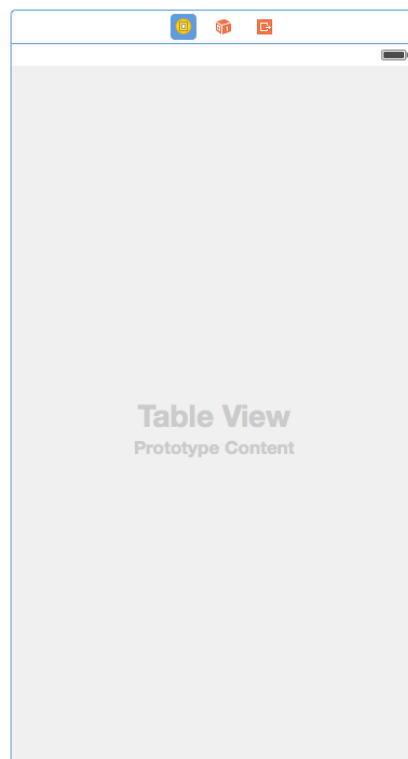


2. Thiết kế giao diện TableView

- Vào Main.storyboard , kéo đối tượng UITableView lên ViewController như hình:



- Kết quả như hình :



- Sau đó : Kết nối DataSource và Delegate:



- Vào ViewController.h , khai báo thêm 2 protocol : UITableViewDataSource, UITableViewDelegate

```
#import <UIKit/UIKit.h>

@interface ViewController : UIViewController <UITableViewDataSource, UITableViewDelegate>

@end
```

- Để truyền dữ liệu cho UITableView danh sách dữ liệu để hiển thị. Chúng ta phải dùng giao thức UITableViewDataSource. UITableViewDataSource là kết nối giữa dữ liệu của bạn với Table View.
- UITableViewDelegate là đối tượng quản lý giao diện của UITableView. Các phương thức tùy chọn của các giao thức cho phép bạn quản lý chiều cao của hàng trong bảng, tái trật tự các ô (cell) trong bảng...

3. Viết code xử lý hiển thị thông tin trên TableView

TableView của chúng ta hiển thị danh sách thông tin về Ca Sĩ, chúng ta tạo bộ dữ liệu :

- Vào ViewController.m , tạo 3 mảng dữ liệu dsCaSy, dsHìnhCaSy, dsAlbum

```

@implementation ViewController
{
    NSArray *dsCaSy;
    NSArray *dsHinhCaSy;
    NSArray *dsAbum;
}

```

- Trong hàm ViewDidLoad , gán giá trị cho mảng

```

- (void)viewDidLoad {
    [super viewDidLoad];
    dsCaSy = [NSArray arrayWithObjects:@"Bích Phương",@"Hỗ Ngọc Hà",@"Noo Phước Thịnh",@"Sơn Tùng MTP", @"Thủy Chi", @"Trúc Nhân", nil];
    dsHinhCaSy = [NSArray arrayWithObjects:@"bichphuong.png",@"hongocha.png",@"noophuochthinh.png",@"sontung.png",@"thuychi.png",@"trucnhan.png",
    nil];
    dsAbum = [NSArray arrayWithObjects:@"Mình yêu nhau đi", @"Tìm Lại Giấc Mơ", @"Gạt đi nước mắt", @"Em của ngày hôm qua", @"Giữ em đi", @"Bốn
    chữ lắm", nil];

    // Do any additional setup after loading the view, typically from a nib.
}

```

- Cuối cùng chúng ta phải thêm 2 phương thức datasource là:
tableView:numberOfRowsInSection và tableView:cellForRowAtIndexPath.

```

- (NSInteger)tableView:(UITableView *)tableView numberOfRowsInSection:(NSInteger)section
{
    return [dsCaSy count];
}

- (UITableViewCell *)tableView:(UITableView *)tableView cellForRowAtIndexPath:(NSIndexPath *)indexPath
{
    static NSString *simpleTableIdentifier = @"SimpleTableCell";

    UITableViewCell *cell = [tableView dequeueReusableCellWithIdentifier:simpleTableIdentifier];

    if (cell == nil) {
        cell = [[UITableViewCell alloc] initWithStyle:UITableViewCellStyleDefault reuseIdentifier:simpleTableIdentifier];
    }

    cell.textLabel.text = [dsCaSy objectAtIndex:indexPath.row];
    NSString *tenHinh = [dsHinhCaSy objectAtIndex:indexPath.row];
    cell.imageView.image = [UIImage imageNamed:tenHinh];
    return cell;
}

```

- Đến đây các bạn tiến hành chạy thử , kết quả đã hiện ra : Với dữ liệu được đổ vào từ 3 mảng và các trật tự bố trí dữ liệu **mặc định** của TableView.

