

Aprendendo a programar para iOS



UIAlertController

An object that displays an alert message to the user.

Utilizando UIAlertController

- Classes utilizadas pra exibir mensagens de alerta para o usuário.
- Você cria o alerta escolhendo um estilo, cria ações (botões) escolhendo o estilo e métodos de callback, adiciona as ações ao alerta e exibe o mesmo.

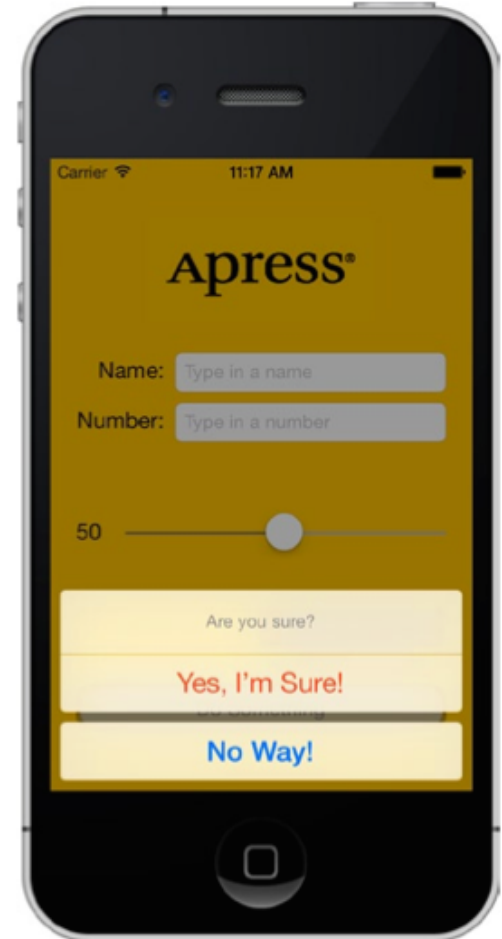
Utilizando UIAlertController

- Estilo do UIAlertController são:
 - UIAlertControllerStyleAlert
 - Alerta no centro da tela
 - UIAlertControllerStyleActionSheet
 - Alerta na parte inferior da tela

Utilizando UIAlertController

- Estilo do UIAlertAction são:
 - `UIAlertActionStyleDefault`
 - Botão padrão.
 - `UIAlertActionStyleCancel`
 - Estilo que indica cancelamento da operação sem mudar nada.
 - `UIAlertActionStyleDestructive`
 - Estilo que indica que esta ação pode perder ou deletar dados.

Utilizando UIAlertController



Utilizando UIAlertController

- Criando uma UIAlertController com um botão

```
UIAlertController *alertController = [UIAlertController
    alertControllerWithTitle:@"Opa"
                        message:@"Você deseja realmente fazer um alerta? "
    preferredStyle:UIAlertControllerStyleAlert];

UIAlertAction* okButton = [UIAlertAction actionWithTitle:@"Não"
                        style:UIAlertActionStyleDefault
                        handler:nil];

[alertController addAction:okButton];

[self presentViewController:alertController animated:YES completion:nil];
```

Utilizando UIAlertController

- Adicionando ações no clique dos botões.

```
UIAlertAction* cancelButton2 = [UIAlertAction  
    initWithTitle:@"Sim"  
    style:UIAlertActionStyleDefault  
    handler:^(UIAlertAction *action){  
        [SVProgressHUD show];  
    }];
```


UITableView

A view that presents data using rows arranged in a single column.

Utilizando UITableView

- Classe utilizada para exibir uma lista de itens em uma única coluna.
- É uma subclasse de UIScrollView, o que permite que o usuário faça “scroll” na lista verticalmente.
- Toda célula da lista é exibida utilizando UITableViewCell, que possui por padrão um título, imagem e uma view adicional no canto direito. Utilizando UITableView

Utilizando UITableView

- As UITableViewCell podem ser customizadas no storyboard ou utilizando Subclasses e arquivos XIB para a interface.
- UITableViews podem entrar em modo de edição, permitindo que usuários removam/ordenem/adicionem itens.
- Uma tabela é feita com uma ou mais “Sections”, cada “Section” contendo uma ou mais “Células”.

Utilizando UITableView

- Métodos avançados para sobrecarga:

```
(NSInteger)numberOfSectionsInTableView:(UITableView *)tableView;
```

```
(NSString *)tableView:(UITableView *)tableView  
titleForHeaderInSection:(NSInteger)section;
```

```
(UIView *)tableView:(UITableView *)tableView  
viewForHeaderInSection:(NSInteger)section;
```

```
(CGFloat)tableView:(UITableView *)tableView  
heightForHeaderInSection:(NSInteger)section;
```

```
(void)tableView:(UITableView *)tableView didSelectRowAtIndexPath:  
(NSIndexPath *)indexPath;
```

Utilizando UITableView

- Métodos utilitários:

```
[self.tableView setEditing:YES animated:YES];
```

```
[self.tableView reloadData];
```

```
[self.tableView reloadSectionIndexTitles];
```

```
[self.tableView deselectRowAtIndexPath:indexPath animated:YES];
```

Links

- Material desta aula:

Exercícios

- Crie uma aplicação com uma table view com Sections para regiões do Recife e células para os bairros.
- Valores a serem exibidos devem ser armazenados em uma propriedade.
- Ao selecionar uma célula, um UIAlertView deve ser exibido, perguntando se aquela célula deve ser deletada, ou se a tela de detalhes deve ser exibida.
- Se o usuário escolher deletar, a tableView deve ser recarregada sem a célula deletada.
- Se o usuário escolher exibir detalhes, uma nova viewController deve ser exibida utilizando UINavigationController