



Listando os alunos cadastrados

Continuando o projeto, agora precisamos recuperar os alunos que cadastramos para exibi-los na primeira tela do nosso app.

Vamos voltar para o arquivo `HomeTableViewController` e importar o `Core Data` novamente:

```
import UIKit
import CoreData
```

Para recuperar os alunos, faremos uso da classe ***NSFetchRequest***. Vamos criar o método '`recuperaAlunos()`' e em seguida vamos criar o objeto `NSFetchRequest`:

```
func recuperaAluno() {
    let pesquisaAluno:NSFetchRequest<Aluno> = Aluno.fetchRequest()
}
```

Agora que criamos a constante `pesquisaAlunos` do tipo `NSFetchRequest`, podemos aproveitar e pedir para ela trazer os alunos ordenados por ordem alfabética. Para isso, vamos fazer uso da classe `NSSortDescriptor`:

```
func recuperaAluno() {
    let pesquisaAluno:NSFetchRequest<Aluno> = Aluno.fetchRequest()
    let ordenaPorNome = NSSortDescriptor(key: "nome", ascending: true)
    pesquisaAluno.sortDescriptors = [ordenaPorNome]
}
```

Agora que já criamos uma variável onde contem qual entidade o `Core Data` irá buscar e outra onde passamos a ordenação, o próximo passo será fazer efetivamente a busca. Para conseguirmos fazer a precisaremos utilizar a classe ***NSFetchedResultsController***

Essa classe é utilizada para gerenciar os resultados de busca do `Core Data`.

Vamos criar uma variável, do tipo `NSFetchedResultsController` para conseguirmos recuperar os alunos salvo no banco de dados do nosso app:

```
// MARK: - Variáveis

let searchController = UISearchController(searchResultsController: nil)
var gerenciadorDeResultados:NSFetchedResultsController<Aluno>?
```

Quando criamos a variável precisamos especificar o tipo da entidade que criamos no `Core Data`.

Agora podemos utilizar essa variável que criamos para fazer as buscas:

```
gerenciadorDeResultados = NSFetchedResultsController(fetchRequest: NSFetchRequest<Aluno>, managedObjectContext:
NSManagedObjectContext, sectionNameKeyPath: String?, cacheName: String?)
```

Repare que no método construtor precisamos passar um `NSFetchRequest` que já temos (`pesquisaAlunos`)

Em seguida precisamos passar o **contexto**, vamos copiar a implementação que fizemos no formulário:

```
//MARK: - Variáveis

var contexto:NSManagedObjectContext {
    let appDelegate = UIApplication.shared.delegate as! AppDelegate
    return appDelegate.persistentContainer.viewContext
}
let searchController = UISearchController(searchResultsController: nil)
var gerenciadorDeResultados:NSFetchedResultsController<Aluno>?
```

Agora que temos o contexto, vamos usá-lo no construtor:

```
func recuperaAluno() {
    let pesquisaAluno:NSFetchRequest<Aluno> = Aluno.fetchRequest()
    let ordenaPorNome = NSSortDescriptor(key: "nome", ascending: true)
    pesquisaAluno.sortDescriptors = [ordenaPorNome]

    gerenciadorDeResultados = NSFetchedResultsController(fetchRequest: pesquisaAluno, managedObjectContext: contexto, sectionNameKeyPath: nil, cacheName: nil)
    gerenciadorDeResultados?.delegate = self

    do {
        try gerenciadorDeResultados?.performFetch()
    } catch {
        print(error.localizedDescription)
    }
}
```

Vamos chamar esse método no `ViewDidLoad()`

```
// MARK: - View Lifecycle

override func viewDidLoad() {
    super.viewDidLoad()
    self.configuraSearch()
    self.recuperaAluno()
}
```

Agora é a sua vez:

- Implemente a listagem de alunos na `TableView`.

