

14

Para saber mais - Salvando imagens

Quando precisamos persistir imagens pesadas no device, ao invés de salvar a imagem direto no Core Data, podemos salvá-la no **sistema de arquivos** (File System) do iOS, e salvar apenas o seu caminho (path) no Core Data. Dessa forma conseguimos otimizar as consultas e o consumo de memória.

Para entender como funciona essa forma de persistência de imagem, daremos um exemplo utilizando o próprio app da agenda.

Salvando o path da imagem

O primeiro passo para salvar uma imagem no sistema de arquivos do iOS é obter o caminho que a imagem será salva. Para obter esse **caminho** (path), faremos uso da classe **NSHomeDirectory**.

```

    @IBAction func buttonSalvar(_ sender: UIButton) {
        if aluno == nil {
            aluno = Aluno(context: contexto)
        }
        let caminhoDoSistemaDeArquivos = NSHomeDirectory()
        print(caminhoDoSistemaDeArquivos)

        aluno?.nome = textFieldNome.text
        aluno?.endereco = textFieldEndereco.text
        aluno?.telefone = textFieldTelefone.text
        aluno?.site = textFieldSite.text
        aluno?.nota = (textFieldNota.text! as NSString).doubleValue
        aluno?.foto = imageAluno.image

        do {
            try contexto.save()
            navigationController?.popViewController(animated: true)
        } catch {
        }
    }

```

Para não deixar todas as imagens soltas dentro desse diretório, uma boa prática é criar pastas para organizar os arquivos. Vamos começar criando uma pasta chamada **Documents** e dentro dela criaremos uma pasta chamada **Images** onde realmente ficará as imagens persistidas no device.

```

    @IBAction func buttonSalvar(_ sender: UIButton) {
        if aluno == nil {
            aluno = Aluno(context: contexto)
        }
        let caminhoDoSistemaDeArquivos = NSHomeDirectory() as NSString
        let diretorioDeImagens = "Documents/Images"
        let caminhoCompleto = caminhoDoSistemaDeArquivos.appendingPathComponent(diretorioDeImagens)

        print(caminhoCompleto)

        aluno?.nome = textFieldNome.text
        aluno?.endereco = textFieldEndereco.text
        aluno?.telefone = textFieldTelefone.text
        aluno?.site = textFieldSite.text
        aluno?.nota = (textFieldNota.text! as NSString).doubleValue
        aluno?.foto = imageAluno.image

        do {
    
```

Primeiro fizemos um casting do **NSHomeDirectory** para **NSString** assim conseguimos utilizar o método **appendPathComponent** onde concatenamos o caminho do sistema com o caminho que criamos para organizar as imagens (Documents/Images).

Agora que já temos o caminho onde a imagem será salva, precisamos de fato criar esse diretório.

Para criar esse diretório no gerenciador de arquivos do iOS, precisamos utilizar a classe **FileManager**.

```
gerenciador de
arquivos

102    @IBAction func buttonSalvar(_ sender: UIButton) {
103        if aluno == nil {
104            aluno = Aluno(context: contexto)
105        }
106
107        let caminhoDoSistemaDeArquivos = NSHomeDirectory() as NSString
108        let diretorioDeImagens = "Documents/Images"
109        let caminhoCompleto = caminhoDoSistemaDeArquivos.appendingPathComponent(diretorioDeImagens) ⚠️ Initialization of immutable value 'gerenciadorDeArquivos' was never used
110        let gerenciadorDeArquivos = FileManager.default
111
112        aluno?.nome = textFieldNome.text
113        aluno?.endereco = textFieldEndereco.text
114        aluno?.telefone = textFieldTelefone.text
115        aluno?.site = textFieldSite.text
116        aluno?.nota = (textFieldNota.text! as NSString).doubleValue
117        aluno?.foto = imageAluno.image
118
119        do {
120            try contexto.save()
121            navigationController?.popViewController(animated: true)
122        } catch {
123            print(error.localizedDescription)
124        }
125    }
126}
127
```

Agora que temos o gerenciador de arquivos, podemos utilizar o método **createDirectory** passando o caminho que criamos:

criando o diretório

```
utilizando o caminho criado

@IBAction func buttonSalvar(_ sender: UIButton) {
    if aluno == nil {
        aluno = Aluno(context: contexto)
    }

    let caminhoDoSistemaDeArquivos = NSHomeDirectory() as NSString
    let diretorioDeImagens = "Documents/Images"
    let caminhoCompleto = caminhoDoSistemaDeArquivos.appendingPathComponent(diretorioDeImagens)

    let gerenciadorDeArquivos = FileManager.default

    do {
        try gerenciadorDeArquivos.createDirectory(atPath: caminhoCompleto, withIntermediateDirectories: false, attributes: nil)
    } catch {
        print(error.localizedDescription)
    }

    aluno?.nome = textFieldNome.text
    aluno?.endereco = textFieldEndereco.text
    aluno?.telefone = textFieldTelefone.text
    aluno?.site = textFieldSite.text
    aluno?.nota = (textFieldNota.text! as NSString).doubleValue
    aluno?.foto = imageAluno.image

    do {
        try contexto.save()
        navigationController?.popViewController(animated: true)
    } catch {
        print(error.localizedDescription)
    }
}
```

Ao disparar a ação do botão **Salvar** o diretório será criado. Porém não é toda vez que clicarmos em Salvar que queremos que o diretório seja criado.

Por isso precisamos fazer uma **verificação** para checar se o **diretório já existe**. Se já existir não precisamos criar novamente, caso contrário criamos.

```
if !gerenciadorDeArquivos.fileExists(atPath: caminhoCompleto) {
    do {
        try gerenciadorDeArquivos.createDirectory(atPath: caminhoCompleto, withIntermediateDirectories: false, attributes: nil)
    } catch {
        print(error.localizedDescription)
    }
}
```

Agora que já temos o diretório criado, o próximo passo é criar um nome para a foto que vamos armazenar.

O nome deve ser **único**, para não ficar sobrescrevendo as imagens que já existem dentro do diretório.

utilizando o objectId

```
120 let nomeDaImagem = String(format: "%@.jpeg", aluno!.objectId.uriRepresentation().lastPathComponent) // Initialization of immutable value 'nomeDaImagem' was never used; consider removing it
121
122 aluno?.nome = textFieldNome.text
123 aluno?.endereco = textFieldEndereco.text
124 aluno?.telefone = textFieldTelefone.text
125 aluno?.site = textFieldSite.text
126 aluno?.nota = (textFieldNota.text! as NSString).doubleValue
127 aluno?.foto = imageAluno.image
128
129 do {
130     try contexto.save()
131 }
```

Thread 1: breakpoint 1.1

nome da imagem

Já temos o nome do arquivo. Agora vamos criar uma **url** onde será concatenado o **caminho + o nome da imagem**:

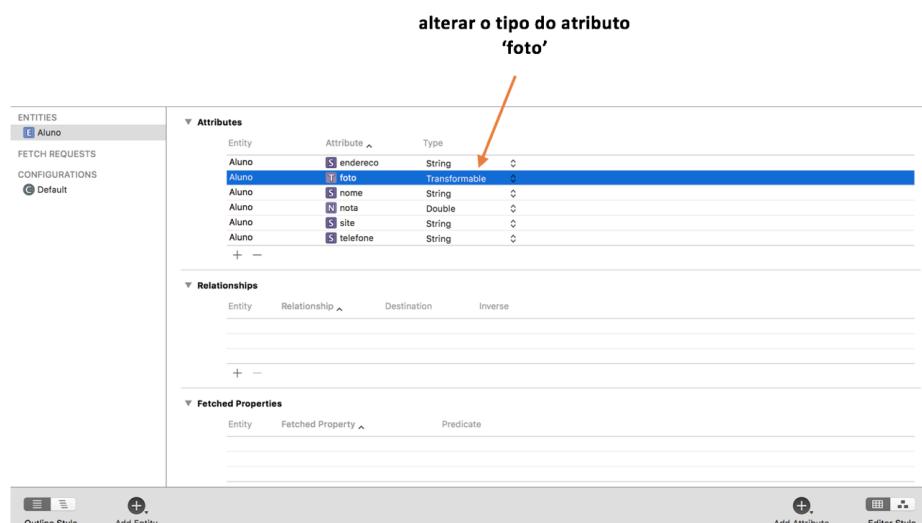
```
120
121     let nomeDaImagem = String(format: "%@.jpeg", aluno!.objectId.uriRepresentation().lastPathComponent)
122
123     let url = URL(fileURLWithPath: String(format: "%@%@", caminhoCompleto, nomeDaImagem)) ⚠️ Initialization of immutable value 'url' was never used; consider replacing with assignment
124
125     aluno?.nome = textFieldNome.text
126     aluno?.endereco = textFieldEndereco.text
127     aluno?.telefone = textFieldTelefone.text
128     aluno?.site = textFieldSite.text
129     aluno?.nota = (textFieldNota.text! as NSString).doubleValue
130     aluno?.foto = imageAluno.image
131
132     sender.buttonSalvar_.title = "Salvar"
133
134     self.agendaAlunoViewController?.dismiss(animated: true, completion: nil)
135
136     let nomeDalgemas = String("Documents/Images")
137     let caminhoCompleto = String("/Users/andriucoleho/Library/Developer/CoreSimulator/Devices/D964B8E4-5BF4-4DC5-A9BD-CCEA90B3F3012.jpeg")
138     let nomeDalgemas = String("CA9B3EDC-5BF4-4DC5-A9BD-CCEA90B3F3012.jpg")
139
140     let caminhoDoSistemaDeArquivos = NSString(format: "/Users/%@/Library/Developer/CoreSimulator/Device...")
```

Agora que temos a url, precisamos **converter a imagem** (`UIImage`) em **Data** para conseguir salvá-la nesse diretório.

```
124     guard let imagem = imageAluno.image else { return }
125     guard let data = UIImagePNGRepresentation(imagem) else { return }
126
127     do {
128         try data.write(to: url)
129     } catch {
130         print(error.localizedDescription)
131     }
132
```

Com essa implementação já estamos salvando a imagem no diretório que criamos.

Porém ainda falta refatorar o **Schema do Core Data**, pois não salvaremos mais a imagem pura, agora precisamos **salvar o caminho da imagem**:



Vamos alterar para **String**:

Entity	Attribute	Type
Aluno	endereco	String
Aluno	foto	String
Aluno	nome	String
Aluno	nota	Double
Aluno	site	String
Aluno	telefone	String

Substituindo o atributo foto pelo path da imagem:

setando o caminho (path) da imagem no atributo foto

path que será salvo no Core Data

```

133     aluno?.nome = textFieldNome.text
134     aluno?.endereco = textFieldEndereco.text
135     aluno?.telefone = textFieldTelefone.text
136     aluno?.site = textFieldSite.text
137     aluno?.nota = (textFieldNota.text! as NSString).doubleValue
138     aluno?.foto = String(format: "%@%@", diretorioDeImagens, nomeDaImagem)
139
140     do {
141         try contexto.save()
142         navigationController?.popViewControllerAnimated(true)
143     } catch {
144         print(error.localizedDescription)
145     }
146 }
```

Thread 1: breakpoint 1.1

(lldb) po aluno!.foto!
"Documents/Images/t87182D05-AB7-4A43-93AC-E826B4C79CB12.jpeg"

Recuperando a imagem salva

Como foi alterada a forma de salvar imagem, precisamos refatorar 2 lugares onde setamos a imagem no UIImage.

Primeiro vamos abrir o arquivo **HomeTableViewCell**:

```

19     func configuraCelula(_ aluno:Aluno) {
20         labelNomeDoAluno.text = aluno.nome
21         viewImagem.layer.cornerRadius = imageAluno.frame.width / 2
22
23         let gerenciadorDeArquivos = FileManager.default
24
25         let caminho = NSSearchPathDirectory(.documentDirectory, inDomain: .userDomainMask, appropriateFor: nil, create: false) as NSString
26         let caminhoDaImagem = caminho.appendingPathComponent(aluno.foto!)
27
28         if gerenciadorDeArquivos.fileExists(atPath: caminhoDaImagem) {
29             imageAluno.image = UIImage(contentsOfFile: caminhoDaImagem)
30         }
31     }

```

Faremos o mesmo no arquivo **AlunoViewController**:

```

45 // MARK: - Métodos
46
47 func setup() {
48     imagePicker.delegate = self
49     guard let alunoSelecionado = aluno else { return }
50     textFieldNome.text = alunoSelecionado.nome
51     textFieldEndereco.text = alunoSelecionado.endereco
52     textFieldTelefone.text = alunoSelecionado.telefone
53     textFieldSite.text = alunoSelecionado.site
54     textFieldNota.text = "\(alunoSelecionado.nota)"
55
56     let gerenciadorDeArquivos = FileManager.default
57
58     let caminho = NSHomeDirectory() as NSString
59     let caminhoDaImagem = caminho.appendingPathComponent(alunoSelecionado.foto!)
60
61     if gerenciadorDeArquivos.fileExists(atPath: caminhoDaImagem) {
62         imageAluno.image = UIImage(contentsOfFile: caminhoDaImagem)
63     }
64 }
```



recuperando a imagem e setando no UIImageView

Rodando o app temos o mesmo resultado, porém com as imagens salvas no gerenciador de arquivos do iOS, e apenas o caminho no Core Data.

Documentação:

- File System:

<https://developer.apple.com/library/content/documentation/FileManagement/Conceptual/FileSystemProgrammingGuide/FileSystemArchitecture.html>
https://developer.apple.com/library/content/documentation/FileManagement/Conceptual/FileSystemProgrammingGuide/FileSystemArchitecture.html#//apple_ref/doc/uid/TP40011888-CH1-SW1