

Soft Blue Soft Blue Soft Blue Soft Blue Soft Blue

Soft Blue Soft Blue Soft Blue Soft Blue Soft Blue

Soft Blue Soft Blue Soft Blue Soft Blue Soft Blue

iOS

Criando Aplicações para iPhone e iPad

Persistência de Dados

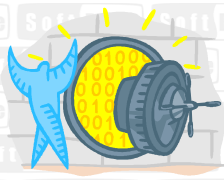
Soft Blue

www.softblue.com.br

Todos os direitos de cópia reservados. Não é permitida a distribuição física ou eletrônica desta matéria sem a permissão expressa e por escrito do autor.

Persistência de dados

- Principais meios de persistência:
 - Lista de propriedades / Serialização de Objetos
 - Gravação e leitura de arquivos (linguagem C)
 - Core Data
 - SQLite3
- Sandbox



Sandbox

- Área reservada pelo iOS destinada para o seu aplicativo criar e gerenciar arquivos e dados
- Cada aplicação possui o seu próprio **sandbox**
- Possível visualizar no iOS Simulator em:
 - Home → Library → Application Support → iPhone Simulator → <iOS Version> → Applications
 - Documents:** Dados gerais, com exceção das configurações
 - Library:** Configurações da aplicação
 - tmp:** Arquivos temporários (sem backup do iTunes)

Lista de propriedades

- Serialização de objetos

- NSArray
- NSMutableArray
- NSDictionary
- NSMutableDictionary
- NSData
- NSMutableData
- NSString
- NSMutableString
- NSNumber
- NSDate



- Fácil manutenção (edição via Xcode)

- Métodos

- **writeToFile**
- **initWithContentsOfFile**

Serializando objetos próprios

- Protocolo **NSCoding**

- **encodeWithCoder** Serializa o objeto
- **initWithCoder** Deserializa o objeto

- Protocolo **NSCopying**

- **copyWithZone** Cria uma cópia do objeto

Exemplo (UserSample.h)

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

@interface UserSample : NSObject <NSCoding, NSCopying> {
    NSString *fieldFirstName;
    NSString *fieldLastName;
    NSString *fieldEmail;
}

@property (nonatomic, retain) NSString *fieldFirstName;
@property (nonatomic, retain) NSString *fieldLastName;
@property (nonatomic, retain) NSString *fieldEmail;

@end
```

Exemplo (UserSample.m)

```
#pragma mark NSCoder
- (void) encodeWithCoder:(NSCoder *)aCoder {
    [aCoder encodeObject:fieldFirstName forKey:@"fieldFirstNameKey"];
    [aCoder encodeObject:fieldLastName forKey:@"fieldLastNameKey"];
    [aCoder encodeObject:fieldEmail forKey:@"fieldEmailKey"];
}

- (id) initWithCoder:(NSCoder *)aDecoder {
    if(self = [super init]) {
        fieldFirstName = [[aDecoder decodeObjectForKey:@"fieldFirstNameKey"] retain];
        fieldLastName = [[aDecoder decodeObjectForKey:@"fieldLastNameKey"] retain];
        fieldEmail = [[aDecoder decodeObjectForKey:@"fieldEmailKey"] retain];
    }
    return self;
}

#pragma mark NSCopying
- (id) copyWithZone:(NSZone *)zone {
    UserSample *myUserSample = [[self class] allocWithZone:zone] init;
    myUserSample.fieldFirstName = [[self.fieldFirstName copyWithZone:zone] autorelease];
    myUserSample.fieldLastName = [[self.fieldLastName copyWithZone:zone] autorelease];
    myUserSample.fieldEmail = [[self.fieldEmail copyWithZone:zone] autorelease];
    return myUserSample;
}
```

SQLite3

- Banco de dados baseado no SQL
- Métodos
 - *sqlite3_open*
 - *sqlite3_prepare_v2*
 - *sqlite3_exec*
 - *sqlite3_column_int*
 - *sqlite3_bind_int* / *sqlite3_bind_text*
 - *sqlite3_step*
 - *sqlite3_close*
 - *sqlite3_finalize*


Notification Center

- Permite disparar determinado método em eventos específicos
- Classe **NSNotificationCenter**
 - Método *addObserver*
- Evento invocado quando a aplicação fecha
 - *UIApplicationWillResignActiveNotification*
- Exemplo

```
UIApplication *app = [UIApplication sharedApplication];

[[NSNotificationCenter defaultCenter] addObserver:self
    selector:@selector(<minhaFuncao>)
    name:UIApplicationWillResignActiveNotification
    object:nil];
```

Aulas práticas e manuais on-line



Assista agora às aulas práticas, que apresentam o uso dos comandos abordados nesta aula teórica.

[Clique aqui](#) para visualizar as aulas práticas disponíveis
