Cocoa : Les bases

Let’s talk about Foundation...

Cocoa : Les bases

- Mutable et immutable
- Objet racine
- Les chaînes de caractères
- Les collections
- Les nombres
- Aller plus loin avec la documentation

Mutable et immutable

Mutable et immutable

- NSGizmo & NSMutableGizmo
- Est-ce que le contenu peut changer ?
  - La valeur de la chaine de caractères
  - Les objets contenus dans une collection
  - ...

- NSGizmo
  - Un bon choix en général
- NSMutableGizmo
  - Si le contenu évolue souvent et incrémentalement

Objet racine

Objet racine

- NSObject
  - Classe racine
  - Fourni les mécanismes de création/destruction d’objet
  - Permet la gestion de la mémoire
  - Permet l’introspection

Objet racine

- Création
  - `+(id) alloc;`
  - `-(id) init;`
- En pratique
  - `NSObject *newObject = [[NSObject alloc] init];`

Les chaînes de caractères

- NSString
  - Gestion de chaînes de caractères Unicode
  - Immutable
- NSMutableString
  - Sous-classe mutable de NSString

- Création simplifiée

```
NSString *aString = @"This is a string"
```

- Les formats

```
@"Une NSString %@, un int %d"
```

- Affichage console

```
NSLog(@"Error: %d", errorCode)
```

- NSString

```
-(id)initWithFormat:(NSString *)format...

-(NSUInteger)length

-(NSArray *)pathComponents

-(BOOL)isEqualToString:(NSString *)aString

...
```

- NSMutableString

```
-(void)setString:(NSString *)aString

-(void)replaceCharactersInRange:(NSRange)aRange withString:
(NSString *)aString

-(void)appendFormat:(NSString *)format ...

...
```

Les collections

- Les tableaux : NSArray
  - Collection d’objets ordonnés et indexés
- Les dictionnaires : NSDictionary
  - Collection d’objets associés à des clés
- Les ensembles : NSSet
  - Collection d’objets non ordonnés
- Et leurs sous-classes mutables !

Tableaux

- NSArray

```
- (id)initWithObjects:(id)firstObj,…,nil

- (NSUInteger)count

- (BOOL)containsObject:(id)anObject

- (NSUInteger)indexOfObject:(id)anObject

...
```

Tableaux

- NSMutableArray

```
- (void)addObject:(id)anObject

- (void)removeLastObject

- (void)insertObject:(id)anObject atIndex:(NSUInteger)index

...
```

Tableaux

```objc
NSArray *array = [NSArray alloc] initWithObjects:obj1, @"toto", obj2, nil];
//Syntaxe "originale"

NSArray *array = @[obj1, @"toto", obj2];
//Syntaxe moderne

NSArray <NSString *> *stringArray = @[@"String 1", @"String 2", @"String 3"];
//Syntaxe avec annotation de type (Xcode 7)
```

Dictionnaires

- NSDictionary

```
-(id)initWithObjectsAndKeys:(id)firstObject,…,nil

- (id)valueForKey:(NSString *)key

- (NSUInteger)count

- (NSArray *)allKeys

...
```

Dictionnaires

- NSMutableDictionary

```
- (void)setValue:(id)value forKey:(NSString *)key

- (void)removeObjectForKey:(id)aKey

- (void)removeAllObjects

...
```

Dictionnaires

```objc
NSDictionary *dict = [[NSDictionary alloc] initWithObjectsAndKeys:array, @"key1", stringArray, @"key2", nil];
//Syntaxe "originale"

NSDictionary *dict = @{@"key1":obj1, @"key2":obj2};
//Syntaxe moderne

NSDictionary <NSString *, Vehicule *> *typedDict = @{@"key1" : vehicule1, @"key2" : vehicule2};
//Avec annotation de types
```

Ensembles

- NSSet

```
- (id)initWithObjects:(id)firstObj, ...
- (NSUInteger)count
- (BOOL)containsObject:(id)anObject
- (id)anyObject
```

Ensembles

- NSMutableSet

```
- (void)addObject:(id)anObject

- (void)addObjectFromArray:(NSArray *)anArray

- (void)intersectSet:(NSSet *)aSet

...
```

Les nombres

- NSNumber
  - “Emballage objet” pour des types primitif
  - Permet de mettre une valeur dans une collection

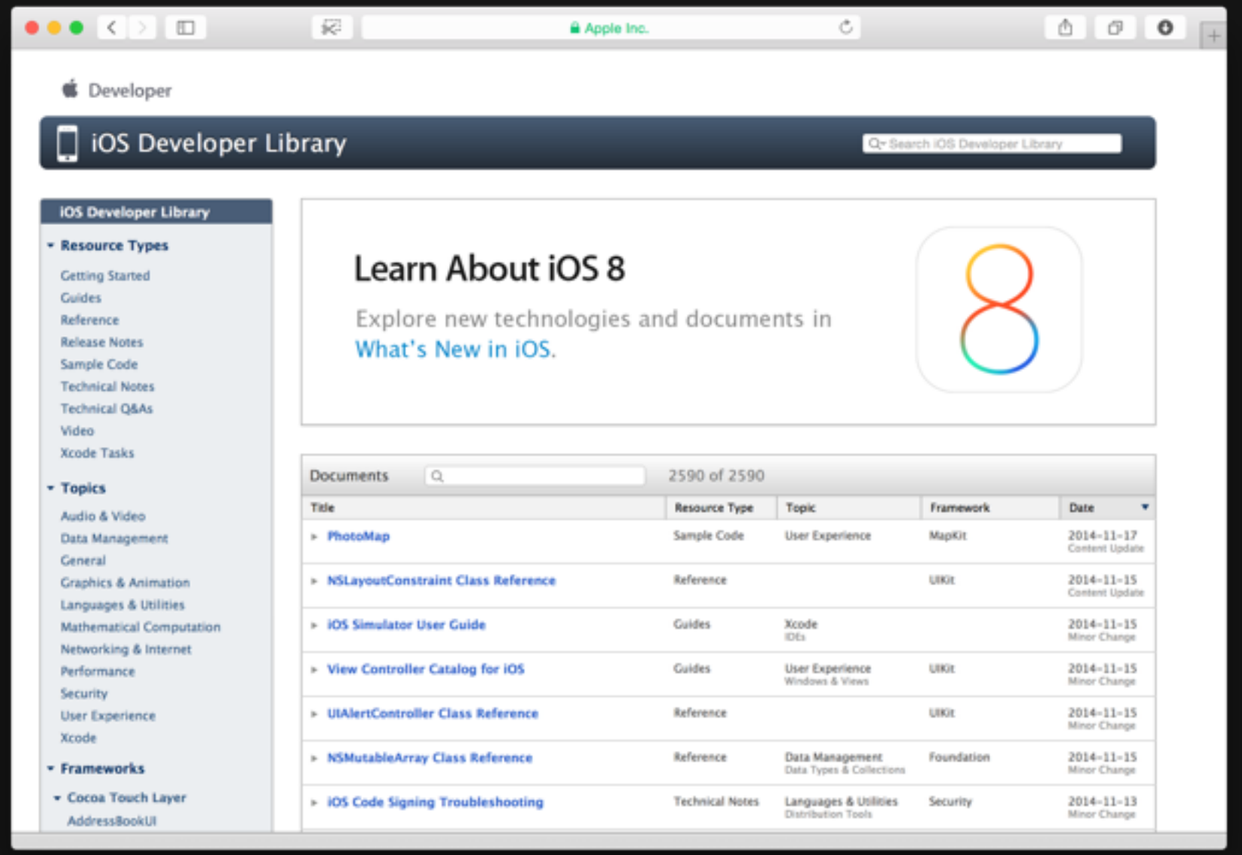
- Créer un NSNumber
  - `+ (NSNumber *)numberWithInt:(int)value`
  - `+ (NSNumber *)numberWithFloat:(float)value`
  - …
- Récupérer un type primitif
  - `- (int)intValue`

Les nombres

```objc
NSNumber *one = [NSNumber numberWithInt:1];

NSNumber *two = @2;

NSNumber *twoAndHalf = @2.5;

float f = [twoAndHalf floatValue];
```

La documentation

La documentation



developer.apple.com/library/ios

XCODE

⌘⇧0

⌘⇧0

developer.apple.com/library/ios

XCODE

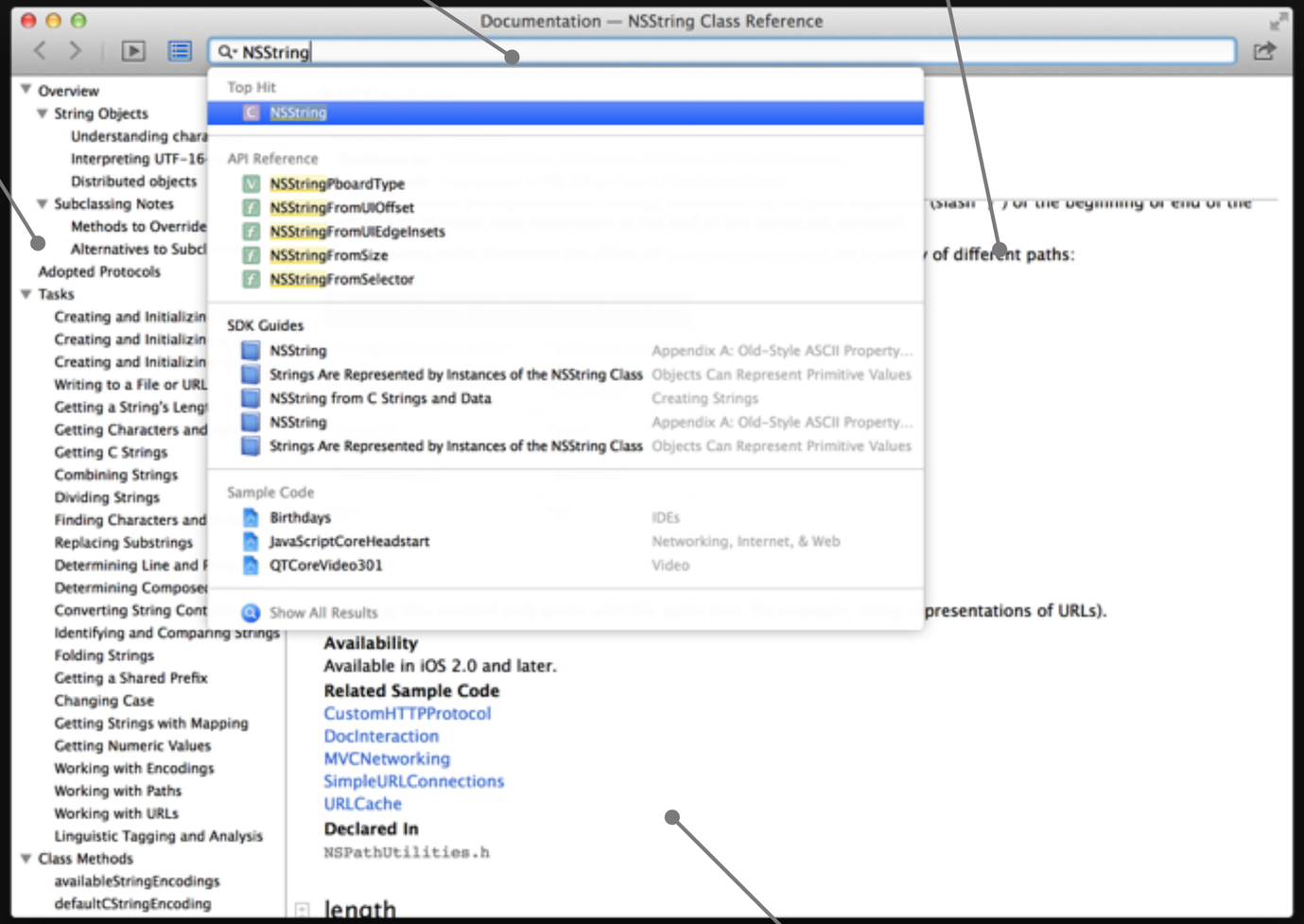
⌘⇧0

La documentation

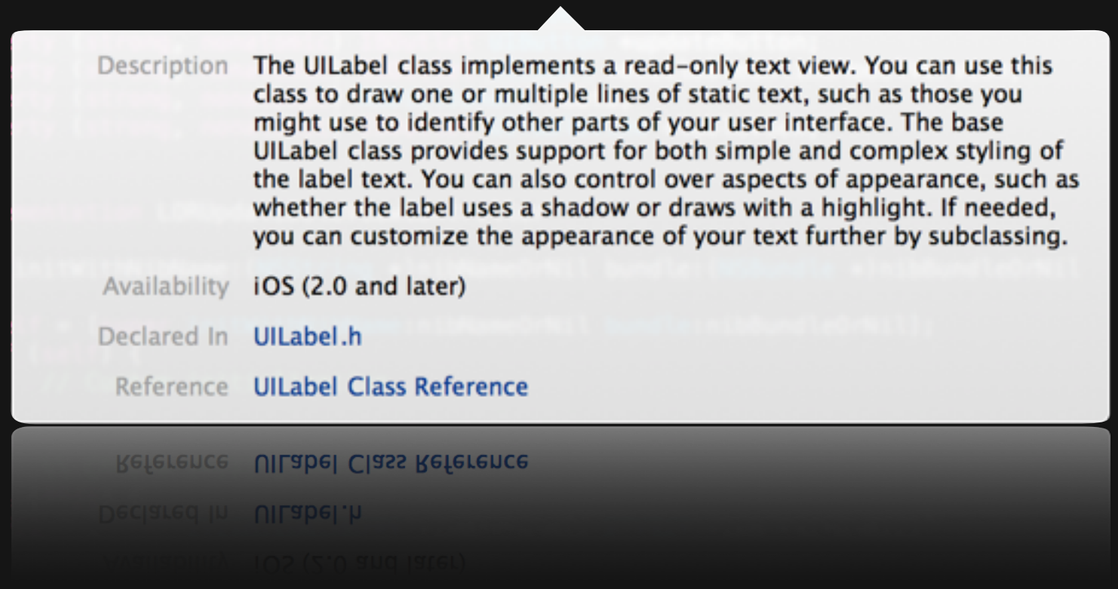
Recherche

Résultats

Table des matières



Documentation



XCODE QUICK HELP

⌥ clic

Pour aller plus loin…

- http://developer.apple.com
- iOS App Programming Guide