

数组

OC数组 只能存储对象类型

NSArray

不可变数组

初始化方法

```
NSArray *names = [[NSArray alloc] initWithObjects:@"frank", @"duck", @"monkey", nil];
```

注: nil作为数组存放元素的结束标志, 多个元素通过逗号间隔。

便利构造器

```
NSArray *names = [NSArray arrayWithObjects:@"frank", @"duck", @"monkey", nil];
```

元素个数

```
@property (readonly) NSUInteger count;
```

获取数组中指定下标对应的元素

```
- (id)objectAtIndex:(NSUInteger)index;
```

字面量方式获取数组元素

数组对象[元素下标]

字面量

使用"@"、"[]"、"[]"中存入对应的对象, 对象之间使用","相隔。比如:

```
NSArray *names = @"frank", @"duck", @"monkey", @"cow";
```

用来判断数组中是否包含某一个给定的对象

```
- (BOOL)containsObject:(id)anObject;
```

获取数组元素对应的索引

```
- (NSUInteger)indexOfObject:(id)anObject;
```

NSString 转换NSArray

按照给定的字符串进行截取, 将截取的多段字符串放入数组中

```
- (NSArray *)componentsSeparatedByString:(NSString *)separator;
```

例: 有一个字符串, 通过.进行分割

```
NSString *string = @"www.lanou3g.com";
NSArray *array = [string componentsSeparatedByString:@"."];
NSLog(@"%@", array);
```

NSArray 转换NSString

将数组中的元素按照给定的字符串格式拼接成一个完整的字符串对象

```
- (NSString *)componentsJoinedByString:(NSString *)separator;
```

例: 有一个数组, 通过&将所有元素拼接成一个字符串

```
NSArray *array = @"北京", @"大连", @"河南", @"上海", @"广州", @"西安";
NSString *string = [array componentsJoinedByString:@"&"];
NSLog(@"%@", string);
```

父类是NSArray

初始化方法

```
NSMutableArray *names = [[NSMutableArray alloc] initWithCapacity:0];
```

便利构造器

```
NSMutableArray *names = [NSMutableArray arrayWithCapacity:0];
```

字面量

```
NSMutableArray *names = [@"frank", @"duck", @"monkey", @"cow"] mutableCopy];
```

数组中添加一个对象

```
- (void)addObject:(id)object;
```

数组中指定位置插入一个对象

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

数组中移除一个对象

```
- (void)removeObject:(id)object;
```

移除数组中最后一个对象

```
- (void)removeLastObject;
```

移除数组中所有的元素

```
- (void)removeAllObjects;
```

数组中移除指定位置的元素

```
- (void)removeObjectAtIndex:(NSUInteger)index;
```

使用指定的对象替换指定位置的元素

```
- (void)replaceObjectAtIndex:(NSUInteger)index withObject:(id)anObject;
```

交换指定的两个下标对应的对象

```
- (void)exchangeObjectAtIndex:(NSUInteger)idx1 withObjectAtIndex:(NSUInteger)idx2;
```

概述

字典是一个无序的集合

字典某个对象都是一对键值 键值是key和value, 两个的值都是对象类型

每一对key-value称为一个条目

字典靠key存取元素

key, Value必须都是对象类型

特点

存储无序

可变和不可变

不可变NSDictionary

初始化方法

```
NSDictionary *name = [[NSDictionary alloc] initWithObjectsAndKeys:@"frank", @"name1", @"duck", @"name2", @"monkey", @"name3", nil];
```

注: 先写value, 再写key, 一对key-value是一个元素, nil作为字典存放元素的结束标志。

便利构造器

```
NSDictionary *name = [NSDictionary dictionaryWithObjectsAndKeys:@"frank", @"name1", @"duck", @"name2", @"monkey", @"name3", nil];
```

字面量

使用"@"、"{} "中存入字典元素, key: value——对应, 元素之间使用","相隔。比如:

```
NSDictionary *dataDic = @{@"key1" : @"frank", @"key2" : @"duck"};
```

获取字典中键值对的个数

```
@property (readonly) NSUInteger count;
```

获取字典中所有的键

```
@property (readonly, copy) NSArray *allKeys;
```

获取字典中所有的值

```
@property (readonly, copy) NSArray *allValues
```

根据键获得对应的值

```
- (id)objectForKey:(NSString *)anAttribute;
```

可变字典NSMutableDictionary

初始化方法

```
NSMutableDictionary *name = [[NSMutableDictionary alloc] initWithCapacity:0];
```

便利构造器

```
NSMutableDictionary *name = [NSMutableDictionary dictionaryWithCapacity:0];
```

字面量

```
NSMutableDictionary *name = [@"key1":@"frank", @"key2":@"duck"] mutableCopy];
```

字典中添加新的键值对以及 修改某一键值对

```
- (void)setObject:(id)anObject forKey:(id <NSCopying>)aKey;
```

移除指定的键对应的键值对

```
- (void)removeObjectForKey:(id)aKey;
```

移除字典中所有的键值对

```
- (void)removeAllObjects;
```

特点

互异性

集合中不能存在两个相同的对象

无序形

集合中的对象没有顺序

经常用来处理重用问题

不可变集合NSSet

初始化方法

```
NSSet *name = [[NSSet alloc] initWithObjects:@"frank", @"duck", @"monkey", nil];
```

便利构造器

```
NSSet *name = [NSSet setWithObjects:@"frank", @"duck", @"monkey", nil];
```

注意: 不可变集合一旦创建, 集合中的对象无法修改, 只能从集合中读取对象, 并且没有快速创建集合对象的字面量。

获取集合中对象的个数

```
@property (readonly) NSUInteger count;
```

获取集合中所有的对象

```
@property (readonly, copy) NSArray *allObjects;
```

从集合中取出一个对象

```
- (id)anyObject;
```

判断集合中是否包含某一个指定的对象

```
- (BOOL)containsObject:(id)anObject;
```

可变集合NSMutableSet

初始化方法

```
NSMutableSet *name = [[NSMutableSet alloc] initWithCapacity:0];
```

便利构造器

```
NSMutableSet *name = [NSMutableSet setWithCapacity:0];
```

添加一个对象

```
- (void)addObject:(id)object;
```

移除一个对象

```
- (void)removeObject:(id)object;
```

移除所有对象

```
- (void)removeAllObjects;
```

OC中为我们提供了三大容器: 数组、字典和集合。三个容器中存储的都是对象类型。

数组: 是一个有序的集合。当我们管理有序的一组对象的时候需要使用数组。

字典: 是一个无序集合。里面面存储的对象是一个一个的键-值对。我们获取数据的时候需要使用对应的key来操作, 并且key在字典中是唯一的。

集合: 集合是一个无序的容器。并且容器中的对象不能够重复。

可变容器类对象是不可变容器类对象的子类, 在拥有父类功能的基础上扩充了对原有对象的增删改操作。