

Коллекции и структуры данных в Swift

Основные коллекции

Swift предоставляет три основных типа коллекций:

- 1) **Array**: упорядоченный список элементов.
- 2) **Dictionary**: ключ-значение пары.
- 3) **Set**: неупорядоченное множество уникальных элементов.

Все коллекции типизированы и безопасны.

Пример:

```
let array = [1, 2, 3]           // Array
let dict = [1: "one"]           // Dictionary
let set = Set([1, 2, 2, 3])     // Set: [1, 2, 3]
```

Работа с Array

Неизменяемый массив:

```
let numbers = [1, 2, 3, 4] // [1, 2, 3, 4]
```

Изменяемый массив:

```
var mutableNumbers = [1, 2, 3]  
mutableNumbers.append(4) // [1, 2, 3, 4]  
mutableNumbers[0] = 10   // [10, 2, 3, 4]
```

Доступ по индексу:

```
print(numbers[1]) // Вывод: 2
```

Работа с Dictionary

Неизменяемый словарь:

```
let words = [1: "one", 2: "two"] // [1: "one", 2: "two"]
```

Изменяемый словарь:

```
var mutableWords = [1: "one"]  
mutableWords[2] = "two" // [1: "one", 2: "two"]  
mutableWords[1] = nil   // [2: "two"]
```

Доступ к значению:

```
print(words[1] ?? "none") // Вывод: one
```

Работа с Set

Неизменяемое множество:

```
let unique = Set([1, 2, 2, 3]) // [1, 2, 3]
```

Изменяемое множество:

```
var mutableSet = Set([1, 2])  
mutableSet.insert(3) // [1, 2, 3]  
mutableSet.remove(1) // [2, 3]
```

Проверка наличия:

```
print(mutableSet.contains(2)) // Вывод: true
```

Итерация по коллекциям

Array:

```
let colors = ["red", "green"]
for color in colors {
    print(color) // Вывод: red, green
}
```

Dictionary:

```
let dict = [1: "one", 2: "two"]
for (key, value) in dict {
    print("\(key): \(value)") // Вывод: 1: one, 2: two
}
```

Set:

```
let set = Set([1, 2, 3])
for num in set {
    print(num) // Вывод: 1, 2, 3 (порядок не гарантирован)
}
```

Полезные методы

Array:

```
let nums = [1, 2, 3]
print(nums.count)    // Вывод: 3
print(nums.isEmpty) // Вывод: false
nums.map { $0 * 2 } // [2, 4, 6]
```

Dictionary:

```
let dict = [1: "a", 2: "b"]
print(dict.keys) // Вывод: [1, 2]
```

Set:

```
let set1 = Set([1, 2])
let set2 = Set([2, 3])
print(set1.union(set2)) // Вывод: [1, 2, 3]
```

Практическое применение

Пример фильтрации массива:

```
let numbers = [1, 2, 3, 4, 5]
let evenNumbers = numbers.filter { $0 % 2 == 0 }
print(evenNumbers) // Вывод: [2, 4]
```

Использование в **iOS**:

- **Array**: список элементов в **UITableView**.
- **Dictionary**: хранение настроек **UI**.
- **Set**: уникальные идентификаторы объектов.