

样本

索引	0			1			2			3			
firstName	Alice			Bob			Charlie			Quentin			
lastName	Smith			Jones			Smith			alberts			
age	24			27			33			31			
nums	99	18	6	71	43	83	18	96	69	70	48	51	55

样本

```
@interface Person : NSObject

@property NSString *firstName;
@property NSString *lastName;
@property NSUInteger age;
@property NSArray *nums;

+ (NSMutableArray *)peoples;

@end
```

示例代码

替换

%K	key path 的替换值。（不可用 %@ 作为 key path 的替换值）
\$VARIABLE_NAME	NSPredicate -predicateWithSubstitutionVariables: 的替换值

```
[NSPredicate predicateWithFormat:@"%K = 'Bob'", @"firstName"];
[NSPredicate predicateWithFormat:@"%K >= %d", @"age", 30];

[NSPredicate predicateWithFormat:@"(firstName BEGINSWITH[c] $letter) OR (lastName BEGINSWITH $letter)"];
[array filteredArrayUsingPredicate:[letterPredicate predicateWithSubstitutionVariables:@{@"letter": @"a"}]]
// 表示firstName以a、A开头，并且lastName以a开头
// [c]表示不区分大小写
```

示例代码

比较

<code>=, ==</code>	左边的表达式 <code>=</code> 右边的表达式。
<code>>=, =></code>	左边的表达式 <code>>=</code> 右边的表达式。
<code><=, =<</code>	左边的表达式 <code><=</code> 右边的表达式。
<code>></code>	左边的表达式 <code>></code> 右边的表达式。
<code><</code>	左边的表达式 <code><</code> 右边的表达式。
<code>!=, <></code>	左边的表达式 <code>!=</code> 右边的表达式。
<code>BETWEEN</code>	左边的表达式处在闭区间。 <code>\$INPUT BETWEEN { \$LOWER, \$UPPER }</code> 。

```
[NSPredicate predicateWithFormat:@"age > 30"];

NSEExpression *lhs = [NSEExpression expressionWithFormat:@"age"];
NSEExpression *rhs = [NSEExpression expressionWithFormat:@"30"];
NSComparisonPredicate *comparisonPredicate = [NSComparisonPredicate predicateWithLeftExpression:lhs
rightExpression:rhs modifier:0 type:NSGreaterThanPredicateOperatorType options:0];
```

示例代码

复合

<code>AND, &&</code>	逻辑与。
<code>OR, </code>	逻辑或。
<code>NOT, !</code>	逻辑非。

```
[NSPredicate predicateWithFormat:@"age > 30 && firstName == 'Quentin'"];

NSPredicate *predicateA = [NSPredicate predicateWithFormat:@"age > 30"];
NSPredicate *predicateB = [NSPredicate predicateWithFormat:@"firstName == 'Quentin'"];
NSCompoundPredicate *compoundPredicate = [NSCompoundPredicate
andPredicateWithSubpredicates:[predicateA, predicateB]];
```

示例代码

字符串比较

BEGINSWITH	以 右边的表达式 开头。
CONTAINS	包含 右边的表达式。
ENDSWITH	以 右边的表达式 结束。
LIKE	模糊匹配。? 和 * 为通配符。? 匹配 1 个字符；* 匹配 0 个或者多个字符。
MATCHES	正则表达式匹配。

```
[NSPredicate predicateWithFormat:@"%K LIKE '?li*', @"firstName"];  
  
NSString *regexp = @"^A.+e$";    // 以A开头，e结尾  
[NSPredicate predicateWithFormat:@"%K MATCHES %@", @"firstName", regexp];
```

示例代码

集合操作

ANY, SOME	匹配任一元素。
ALL	匹配所有元素。
NONE	没有元素匹配。
IN	在指定集合中。
array[index]	指定数组中特定索引处的元素。
array[FIRST]	指定数组中第一个元素。
array[LAST]	指定数组中最后一个元素。
array[SIZE]	指定数组的个数

```
NSPredicate *anyPredicate = [NSPredicate predicateWithFormat:@"%ANY nums > 100"];  
// 表示Person.num中的所有元素中，只要有一个元素>100，那么这个Person就满足条件  
  
[NSPredicate predicateWithFormat:@"%age IN {24, 27}"];  
[NSPredicate predicateWithFormat:@"%firstName IN {'Alice', 'Quentin'}"];  
  
[NSPredicate predicateWithFormat:@"%nums[FIRST] < 55"];  
[NSPredicate predicateWithFormat:@"%nums[0] < 55"];  
// 表示Person.num中的第一个元素<55，那么这个Person就满足条件  
[NSPredicate predicateWithFormat:@"%nums[SIZE] == 4"];  
// 表示Person.num中的元素个数为4，那么这个Person就满足条件
```

示例代码

Block 谓词

```
NSPredicate *predicate = [NSPredicate predicateWithBlock:^(BOOL(id _Nullable evaluatedObject,  
NSDictionary<NSString *,id> * _Nullable bindings) {  
    Person *p = (Person *)evaluatedObject;  
    return p.age > 30;  
}];
```

示例代码