

NSPredicates

For Fun and Profit

An NSPredicate is a Query

What's it for?

- string stuff
- filtering arrays
- Core Data
- Spotlight

It's in Foundation

What's it for?

- string stuff
- filtering
- Core Data
- Spotlight

Creating NSPredicates

```
[NSPredicate predicateWithFormat:@"FORMAT_STRING",args]
```

[NSPredicate predicateWithFormat:@"SELF contains %@", @"hello"]

[NSPredicate predicateWithFormat:@”SELF contains %@”,@”hello”]

```
NSPredicate *greeting =  
[NSPredicate predicateWithFormat:@"self contains %@", @"hello"];  
  
STAssertTrue([greeting evaluateWithObject:@"hello everyone"], @"");
```

String Matching

BEGINSWITH

CONTAINS

ENDSWITH

LIKE

MATCHES

String Matching

BEGINSWITH

CONTAINS

ENDSWITH

LIKE

MATCHES

BEGINSWITH

String Matching

BEGINSWITH

CONTAINS

ENDSWITH

LIKE

MATCHES

CONTAINS

String Matching

BEGINSWITH

CONTAINS

ENDSWITH

LIKE

MATCHES

ENDSWITH

String Matching

BEGINSWITH

CONTAINS

ENDSWITH

LIKE

MATCHES

LIKE *star?

destared	de st are d	NO
is staring	is star ing	NO
stars	star s	YES
deathstart	de ath star t	YES
deathstar	de ath star	NO

String Matching

BEGINSWITH

CONTAINS

ENDSWITH

LIKE

MATCHES

Matches `\\\\d+[a-z]`

0A	0A	NO
0a	0a	YES
000000ab	000000ab	NO
000000c	000000c	YES

Strings

- String begins with 'CH'
- longer than 3 chars
- does NOT contain 'broken'
- shorter than 20 chars
- contains at least one digit
- contains NO spaces

But then things change

Strings

- String begins with 'CH'
- longer than 3 chars
- does NOT contain 'broken'
- shorter than 20 chars
- contains at least one digit
- contains NO spaces

Strings

- String begins with 'CH' or 'HC'
- longer than 3 chars
- does NOT contain 'broken'
- contains 'broken' if string starts with 'HC'
- shorter than 20 chars
- contains at least one digit
- contains NO spaces

MATCHES validation

Filtering Arrays

- mutable and immutable
- works great for table views

Simple Filtering

```
NSArray *_start = @[ @1,@3,@5,@7,@8 ];  
  
NSPredicate *_filter =  
    [NSPredicate predicateWithFormat:@"self in {1,5,7}"];  
  
return [_start filteredArrayUsingPredicate:_filter];
```

Mutable Filter

```
NSPredicate *_filter =  
    [NSPredicate predicateWithFormat:@"self in {1,5,7}"];  
  
[_mutable filterUsingPredicate:_filter];
```

Key Paths

Some Keys Nest

Other ways to define

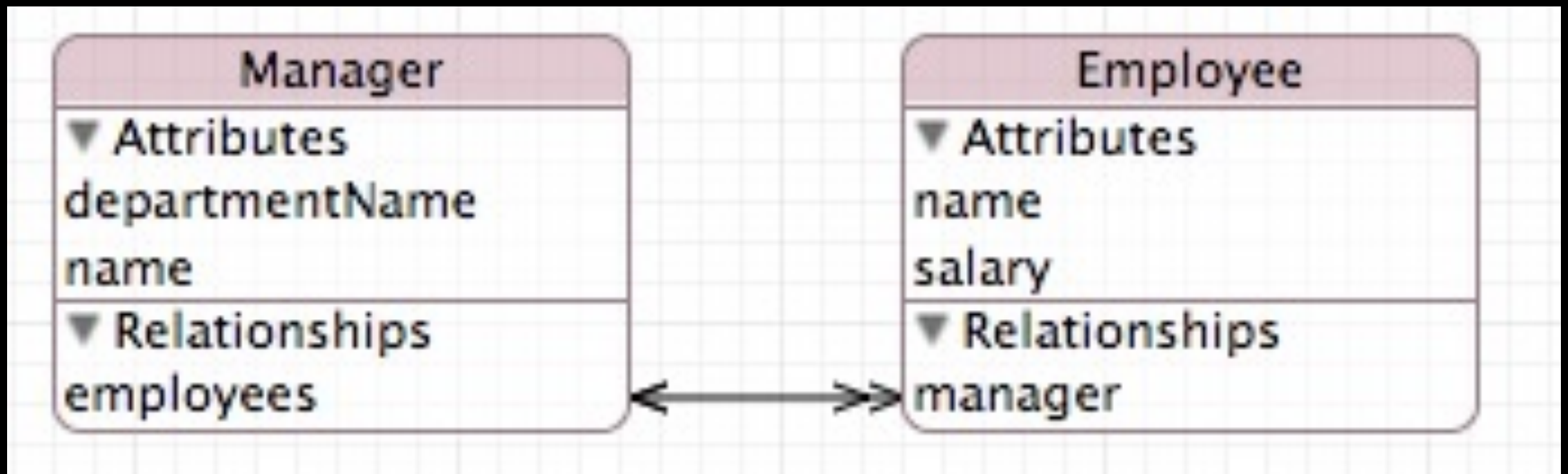
Core Data
IS NOT SQL

You Can't Do This

```
select SOMETABLE.* from SOMETABLE,OTHERTABLE  
    where SOMETABLE.column > 3  
    and OTHERTABLE.othercolumn =  
        SOMETABLE.othercolumn;
```

Object Graph

Model



Subquery

Don't

Core Data Validation

Performance

- It's very fast
- But it's not magic

Do

- minimize the amount of work done
- put 'match' last
- avoid joins
- denormalize performance sensitive data
- use the right store for the right job

Don't

- use joins, use subqueries (sparingly)
- confuse it with SQL databases
- use the wrong store

Remember



The Compiler Can Do It Better

Resources

- Pro Core Data for iOS
- Core Data (Zarra)
- Apple's Docs
- Practice