



# Queries with For

Principles of Functional Programming

## Queries with for

The for notation is essentially equivalent to the common operations of query languages for databases.

**Example:** Suppose that we have a database books, represented as a list of books.

```
case class Book(title: String, authors: List[String])
```

## A Mini-Database

```
val books: List[Book] = List(  
  Book(title = "Structure and Interpretation of Computer Programs",  
        authors = List("Abelson, Harald", "Sussman, Gerald J.")),  
  Book(title = "Introduction to Functional Programming",  
        authors = List("Bird, Richard", "Wadler, Phil")),  
  Book(title = "Effective Java",  
        authors = List("Bloch, Joshua")),  
  Book(title = "Java Puzzlers",  
        authors = List("Bloch, Joshua", "Gafter, Neal")),  
  Book(title = "Programming in Scala",  
        authors = List("Odersky, Martin", "Spoon, Lex", "Venners, Bill")))
```

## Some Queries

To find the titles of books whose author's name is "Bird":

```
for
  b <- books
  a <- b.authors
  if a.startsWith("Bird,")
yield b.title
```

To find all the books which have the word "Program" in the title:

```
for b <- books if b.title.indexOf("Program") >= 0
yield b.title
```

## Another Query

To find the names of all authors who have written at least two books present in the database.

```
for
  b1 <- books
  b2 <- books
  if b1 != b2
    a1 <- b1.authors
    a2 <- b2.authors
    if a1 == a2
yield a1
```

## Another Query

To find the names of all authors who have written at least two books present in the database.

```
for
  b1 <- books
  b2 <- books
  if b1 != b2
    a1 <- b1.authors
    a2 <- b2.authors
    if a1 == a2
  yield a1
```

Why do solutions show up twice?

How can we avoid this?

## Modified Query

To find the names of all authors who have written at least two books present in the database.

```
for
  b1 <- books
  b2 <- books
  if b1.title < b2.title
    a1 <- b1.authors
    a2 <- b2.authors
    if a1 == a2
yield a1
```

## Problem

What happens if an author has published three books?

- ☐ The author is printed once
- ☐ The author is printed twice
- ☐ The author is printed three times
- ☐ The author is not printed at all



## Problem

What happens if an author has published three books?

- ☐ The author is printed once
- ☐ The author is printed twice
- ☒ The author is printed three times
- ☐ The author is not printed at all

## Modified Query (2)

*Solution:* We must remove duplicate authors who are in the results list twice.

This is achieved using the `distinct` method on sequences:

```
val repeated =  
  for  
    b1 <- books  
    b2 <- books  
    if b1.title < b2.title  
    a1 <- b1.authors  
    a2 <- b2.authors  
    if a1 == a2  
  yield a1  
repeated.distinct
```

## Modified Query

*Better alternative:* Compute with sets instead of sequences:

```
val bookSet = books.toSet
for
  b1 <- bookSet
  b2 <- bookSet
  if b1 != b2
  a1 <- b1.authors
  a2 <- b2.authors
  if a1 == a2
yield a1
```