

# Exercise

This is an practical exercise to test your programming and design skills. You should implement the following features using Java or Groovy. There is no need to persist any data to a database or to build a user interface. You should prove your code works by unit testing it to cover the features listed below. Automating your unit testing with Maven or Ant would be preferable.

You have been asked to add a new feature to F1000 which provides the ability to rank journals.

## Feature: Rank journals

In order to highlight the journals which have the biggest impact in a given year

As a user

I want to be able to see journals ordered by their score across a range of years. If two or more journals have the same score they should be assigned the same rank, ordering within a rank should be done alphabetically on the journal's name.

### Scenario 1: Rank journals

Given the following journals have scores for 2010:

Journal A = 5.6, Journal B = 2.4, Journal C = 3.1

When the user views the ranked journals

Then the order and rank should be:

Rank	Journal	Score
1	Journal A	5.6
2	Journal C	3.1
3	Journal B	2.4

## Scenario 2: Rank journals with a shared rank

Given the following journals have scores for 2009:

Journal A = 2.2, Journal B = 6.2, Journal C = 6.2

When the user views the ranked journals

Then the order should be determined by score then journal name:

Rank	Journal	Score
1	Journal B	6.2
1	Journal C	6.2
3	Journal A	2.2

## Scenario 3: Rank journals excluding review journals

Given the following journals have scores for 2008:

Journal A = 5.6, Journal B = 2.4, Journal C = 3.1

And Journal A is a review journal

When the journals are ranked

Then the order should be:

Rank	Journal	Score
1	Journal C	3.1
2	Journal B	2.4