Ruby Practice Exercise #2

*Note that these practice exercises are NOT optional. They are not add-ons to this module – they are part of the **core content** of your course. Do not skip these practice exercises!*

Exercise 2.1 (review Learn Ruby 1)

A small web-based program generates the following HTML snippet:

```
<p>
Thank you, <em>John</em>, for placing your order with us.<br />
Please download our <a href="terms.html">terms and conditions</a>.<hr />
</p>
```

Write a Ruby program that generates this code and allows for the name to be *parametrised*. That is, when the program is executed, arrange for the name in the generated HTML ("John" in the example HTML code) to take its value from a variable object.

Exercise 2.2 (review Learn Ruby 2 and 3)

The Institute maintains a list of lecturer names and areas of interest, for example:

```
Paul Barry -> Programming, Networking, Security, Open Source, Frameworks
Chris Meudec -> Testing, Safety Systems, Formal Systems, Programming Languages
Nigel Whyte -> Graphics, Imaging, Programming, Sign Language, Trees
Austin Kinsella -> Networks, WANs, Programming, Macintosh, Digital Photography
Gerry Moloney -> Placement, Employment, Emerging Systems, Web Development
```

1. Write a Ruby data structure to hold this data, then use the data structure to display the 2nd and 4th area of interest for each lecturer.

Once you have program (1) written, use the data structure in these programs:

2. Write a program to display each lecturer's final area of interest.

3. Write a program to display each lecturer's name together with the number of areas of interest that they have.

4. Write a program to replace the area of interest "Programming" in each lecturer's list with the words "Software Engineering". Print out the entire list once the change has occurred. [Make sure your code changes the data within the data structure].

5. Write a program to sort and display each lecturer's list in alphabetical order. [Make sure your code changes the data within the data structure].

6. Extend the program in (4) to swap the first and last areas of interest on each lecturer's sorted list. [Make sure your code changes the data within the data structure].