Exercise 11.6 The version of **display** shown in Code 11.2 produces the output shown in Figure 11.9. Reorder the statements in the method in your version of the *network* project so that it prints the details as shown in Figure 11.10.

Figure 11.9
Possible output from **display**: superclass call at the beginning of **display** (shaded areas printed by superclass method).

```
Leonardo da Vinci
40 seconds ago - 2 people like this.
  No comments.
Had a great idea this morning.
But now I forgot what it was. Something to do with flying...
```

Figure 11.10
Alternative output from **display** (shaded areas printed by superclass method)

```
Had a great idea this morning.
But now I forgot what it was. Something to do with flying...
Leonardo da Vinci
40 seconds ago - 2 people like this.
  No comments.
```

Exercise 11.7 Having to use a superclass call in **display** is somewhat restrictive in the ways in which we can format the output, because it is dependent on the way the superclass formats its fields. Make any necessary changes to the **Post** class and to the **display** method of **MessagePost** so that it produces the output shown in Figure 11.11. Any changes you make to the **Post** class should be visible only to its subclasses. *Hint:* You could add protected accessors to do this.

Figure 11.11
Output from **display** mixing subclass and superclass details (shaded areas represent superclass details)

```
Leonardo da Vinci
Had a great idea this morning.
But now I forgot what it was. Something to do with flying...
40 seconds ago - 2 people like this.
  No comments.
```

11.10 The instanceof operator

One of the consequences of the introduction of inheritance into the *network* project has been that the **NewsFeed** class knows only about **Post** objects and cannot distinguish between message posts and photo posts. This has allowed all types of posts to be stored in a single list.