7.11.3 Cohesion for readability

There are several ways in which high cohesion benefits a design. The two most important ones are *readability* and *reuse*.

The example discussed in Section 7.11.1, cohesion of the `printWelcome` method, is clearly an example in which increasing cohesion makes a class more readable and thus easier to understand and maintain.

The class cohesion example in Section 7.11.2 also has an element of readability. If a separate `Item` class exists, a maintenance programmer will easily recognize where to start reading code if a change to characteristics of an item is needed. Cohesion of classes also increases readability of a program.

7.11.4 Cohesion for reuse

The second great advantage of cohesion is a higher potential for reuse.

The class cohesion example in Section 7.11.2 also shows an example of this: by creating a separate `Item` class, we can create multiple items, and thus use the same code for more than a single item.

Reuse is also an important aspect of method cohesion. Consider a method in the `Room` class with the following signature:

```
public Room leaveRoom(String direction)
```

This method could return the room in the given direction (so that it can be used as the new `currentRoom`), and also print out the description of the new room that we just entered.

This seems like a possible design, and it can indeed be made to work. In our version, however, we have separated this task into two methods:

```
public Room getExit(String direction)
public String getLongDescription()
```

The first one is responsible for returning the next room, whereas the second one produces the room's description.

The advantage of this design is that the separate tasks can be reused more easily. The `getLongDescription` method, for example, is now used not only in the `goRoom` method, but also in `printWelcome` and the implementation of the *look* command. This is only possible because it displays a high degree of cohesion. Reusing it would not be possible in the version with the `leaveRoom` method.