Using super() - Implementing the toString() method

class Book {
    private int pages; // number of pages in the book

    Book(int pages) {
        this.pages = pages;
    }

    public String toString() {
        return ""+pages; // convert pages to String before returning
    }
}

class Dictionary extends Book {
    private int words; // number of words in the dictionary

    Dictionary(int words, int pages) {
        super(pages);
        this.words = words;
    }

    public String toString() {
        String s = "Number of pages: " + super.toString();
        s = s + ", Number of words: " + words;

        return s;
    }
}

public class BookTest {
    public static void main(String[] args) {
        Dictionary d1 = new Dictionary(100000, 500);
        System.out.println(d1);
    }
}
Final Classes and Methods

1. A final class cannot be extended. Therefore \( X_2 \) cannot extend \( X_1 \). Either \( X_1 \) should not become final, or \( X_2 \) should not attempt to inherit from it.

2. final method cannot be overridden. Therefore, \( \text{foo} \) cannot be overridden in \( X_3 \). The final keyword should be removed from the \( \text{foo} \) version of \( X_2 \).

Construction of Objects

1. \( \text{Cell2} \) constructor called
   TinyCell2 constructor called
   MicroscopicCell constructor called

2. The \( \text{TinyCell2} \) constructor will call the default constructor of its parent class \( \text{Cell2} \). As \( \text{Cell2} \) defines a constructor with arguments, the compiler will not synthesise a no-arguments constructor for \( \text{Cell2} \) and the compiler will report an error.

   Adding the line \( \text{super}(5) \) at the indicated point will force the constructor of \( \text{TinyCell2} \) to call the \( \text{Cell2}(\text{int}) \) constructor as opposed to the default constructor. The program will then compile and run.

Enumerated Types

enum Colour { red, green, blue, yellow, magenta }

class Flower {
    Colour colour;

    Flower(Colour c) {
        colour = c;
    }
}

class ColourTest {
    public static void main(String[] args) {
        Flower f1 = new Flower(Colour.yellow);
        Flower f2 = new Flower(Colour.red);
    }
}