Exercise 9-1   Create and work with interfaces

In this exercise, you'll create and implement the DepartmentConstants interface presented in this chapter. You'll also create and implement an interface named Displayable that's similar to the Printable interface.

Create the interfaces

1. Open the project named ch09_ex1_DisplayableTest in the ex_starts directory.

2. Add an interface named DepartmentConstants that contains the three constants shown in figure 9-4.

3. Add an interface named Displayable. This interface should contain a single method named getDisplayText that returns a String.

Implement the interfaces

4. Edit the Product class so it implements the Displayable interface. The getDisplayText method in this class should format a string that can be used to display the product information.

5. Edit the Employee class so it implements the DepartmentConstants and Displayable interfaces. The getDisplayText method in this class should work like the one in the Product class, and it should use the constants in the DepartmentConstants interface to include the department name in the return value.

Use the classes that implement the interfaces

6. Open the DisplayableTestApp class and add code to it that creates an Employee object, assigns it to a Displayable variable, and displays the information in the Employee object at the console. To get the information for an employee, you'll need to use the getDisplayText method of the Displayable interface.

7. Run the application to make sure that it displays the employee information.

8. Repeat steps 6 and 7 for a Product object.

Use an interface as a parameter

9. Open the DisplayableTestApp class and add a method with this signature:

   `private static String displayMultiple(Displayable d, int count)`

   Write the code for this method so it returns a string that contains the Displayable parameter the number of times specified by the int parameter.

10. Modify the code in the main method so it uses the displayMultiple method to display the employee information once and the product information twice.

11. Run the application to make sure it works correctly.

Use of toString() and equals() methods

12. Add the toString() and equals methods to Product class

13. Add the toString() and equals methods to Employee class

14. Modify the code in the main method to create more Employee objects and compare them by using equals() method.

15. Modify the code in the main method to create more Product objects and compare them by using equals() method.

16. Run the application to make sure it works correctly.

```
public interface DepartmentConstants
{
    int ADMIN = 1;
    int EDITORIAL = 2;
    int MARKETING = 3;
}
```

Figure 9-4   How to code an interface