ACCESSING CLASS VARIABLES

Now that we have some default values, we can add a method that sets some different values for them:

```
StudentResults () {
   Exam_Name= “Unknown”;
   Exam_Score= “No score”;
}

String ExamName (String eName) {
   Exam_Name= eName;
   return Exam_Name;
}
```

This new method is called ExamName and has a String variable called eName between its round brackets. The method doesn’t do a great deal, and is here for simplicity’s sake. It simply sets a value for the Exam_Name field, and returns this field as a value. When we call this method, it will overwrite the default value for Exam_Name and insert a new value. Whatever is in the variable eName will be the new Exam_Name value. Let’s see it in action. Let’s put it to work in the main

class of the publishing class, i.e. the class that contains the main method.

```
package exams;
public class ExamDetails {
  public static void main (String [] args) {
    StudentResults exam= new StudentResults ();
    String examName= exam.ExamName (“Biology”);
    System.out.println(examName);
  }
}
```

What we’ve just done is calling the ExamName method of our exam object. We’re handing over the value of “Biology.” This will then be the value for the Exam_Name field. The Exam_Name value is then returned and stored in the variable examName.