```
//Professor Ziegler
//HW5
//Jinglin Tan

import java.util.Scanner;
import java.io.*;

//program 5 outputs bowling score statistic of each group of scores
class pgm5{
    public static void main(String[] args) throws IOException{
        int score1;
        int score2 = 0;
        int score3 = 0;
        int group = 0;      //track number of groups
        int valid = 0;      //track number of valid groups
        int invalid = 0;    //track number of invalid groups

        File myfile = new File("c:/myfile.txt");    //creates File object
        //Scanner inputFile = new Scanner(System.in);
        Scanner inputFile = new Scanner(myfile);    //read from file

        //PrintWriter outputFile = new PrintWriter(System.out);
        PrintWriter outputFile = new PrintWriter("c:/myoutput.txt"); //write to file

        System.out.print("Please enter the 1st score(enter 999 to stop): ");
        score1 = inputFile.nextInt();   //read score1
        while(score1 != 999){           //set 999 as sentinel
            group++;            //count number of groups
            System.out.print("Please enter the 2nd score: ");
            score2 = inputFile.nextInt();       //read score2
            System.out.print("Please enter the 3rd score: ");
            score3 = inputFile.nextInt();       //read score3
            outputFile.println("score1\tscore2\tscore3");   //headings for scores
            outputFile.printf("%6d %7d %7d", score1, score2, score3);  //print scores
            outputFile.println();
            //call method validGroup() in if() to check if group is valid
            //it's valid when it returns 1 and invalid when it returns 0
            if(validGroup(score1, score2, score3, outputFile) == 0)
                invalid++;      //if invalid, count number of invalid group
            else{
                valid++;        //if valid, count number of valid group
                //call method oneGameScore() to print status of scores
                oneGameScore(score1, outputFile);
                oneGameScore(score2, outputFile);
                oneGameScore(score3, outputFile);
                //call method avg3Scores() in println to print average score
                outputFile.println("The average score is " +
                        avg3Scores(score1, score2, score3));
                //call oneGameScore() again to check average score status
                oneGameScore(avg3Scores(score1, score2, score3), outputFile);
            }
            outputFile.println();
            outputFile.println();
            outputFile.println();
            System.out.print("\nPlease enter the 1st score(enter 999 to stop): ");
            score1 = inputFile.nextInt();
        }
        outputFile.println("The total number of groups processed is " + group);
        outputFile.println("The number of valid groups is " + valid);
        outputFile.println("The number of invalid groups is " + invalid);
        outputFile.flush();     //flush the output buffer
        System.out.println();
        System.out.println("The program has completed");
        outputFile.close();     //close output file
        inputFile.close();      //close input file
    }

    /*method validGroup()
     * Input:   3 scores and a PrintWriter object
     * Process: Use boolean expression in if() to decide validity of the group
     *          Assign 1 or 0 to a variable basing on whether all scores are in 0 to 300
     *          Prints statement of validity for group
     *          Prints error status for a score if it is out of range of 0 to 300
     *          Prints "score is negative" if score is less than 0
     *          Prints "score is over 300" if score is over 300
     * Output:  Prints a statement of validity for group
     *          Prints error status for each score that is out of range
     *          Return an integer 1 or 0
     */
    public static int validGroup(int s1, int s2, int s3, PrintWriter out1){
        int validity = 0;   //set it to 0, invalid by default

        if(s1 >= 0 && s1 <= 300 && s2 >= 0 && s2 <= 300 && s3 >= 0 && s3 <= 300){
            validity = 1;   //when s1,s2 and s3 are within 0 to 300, set it to 1
            out1.println("The group is valid");
        }
        else{               //else, validity is still 0
            out1.println("The group is invalid");
            //based on the type of error, prints an error message for each error score
            if(s1 < 0)
                out1.println("Error: score1 " + s1 + " is negative");
            if(s1 > 300)
                out1.println("Error: score1 " + s1 + " is over 300");
            if(s2 < 0)
                out1.println("Error: score2 " + s2 + " is negative");
            if(s2 > 300)
                out1.println("Error: score2 " + s2 + " is over 300");
            if(s3 < 0)
                out1.println("Error: score3 " + s3 + " is negative");
            if(s3 > 300)
                out1.println("Error: score3 " + s3 + " is over 300");
        }
            return validity;    //return 0 or 1 basing on validity
    }

    /* method oneGameScore()
     * Input:   A score and a PrintWriter object
     * Process: Use boolean expression in nested if() to decide what range the score is in
     *          Based on the range of score, prints a message with the original score and
     *          the bowler's rating
     * Output:  prints a message with the original score and the bowler's rating
     */
    public static void oneGameScore(int s, PrintWriter out2){
        //prints score status for each score basing on score range
        if(s >= 250 && s <= 300)
            out2.println(s + " is a professional game");
        else if(s >= 200 && s <= 249)
            out2.println(s + " is an excellent game");
        else if(s >= 140 && s <= 199)
            out2.println(s + " is a very good game");
        else if(s >= 100 && s <= 139)
            out2.println(s + " is a good game");
```

```java
        else if(s >= 50 && s <= 99)
            out2.println(s + " is a poor game");
        else if(s < 50)
            out2.println(s + " is a horrible game");
    }

    /* method avg3Scores()
     * Input:   3 scores
     * Process: Evaluate the average of 3 scores
     * Output:  Return the average of 3 scores
     */
    public static int avg3Scores(int s1, int s2, int s3){
        return (s1 + s2 + s3) / 3;  //evaluate average score and return it
    }
}
```