```
/*
Author: ud17
Project Title: Compute Average Grade
Description: A students enters 10 grades and the application thereafter the application calculates the average grade.
In case the student average grade is less than 70 points then the program prints a message You need extra help
from the Professor" otherwise if the average is more than 70 points but less than 80 points,
then the program prints "The You are doing a good job". If the student's average is more than 80 points
but less than 90, then the program will print "You are doing a great job". For averages more than 90 points,
the program will print "You are donig an excellent job, keep it up!"

*/


import javax.swing.JOptionPane;
public class computerAverageGradesFinal {

  //Begin the main method.
   public static void main(String[] args) {

  //Set the averageGrade_NoMaxMin and the feedbackMsg.
    double averageGrade_NoMaxMin = fillGradesArray();
    String feedbackMsg = "";
    feedbackMsg = printGradeMsg(averageGrade_NoMaxMin);

   }

   //Begin the submethod called, printGradeMs with a parameter of averageGrade_NoMaxMin.
   public static String printGradeMsg(double averageGrade_NoMaxMin)
      {

      //Set the strGrade and the feedbackMsg.
         String strGrade= "";
         String feedbackMsg="";

      //Validate the feedbackMsg based on the averageGrade_NoMaxMin.
         if (averageGrade_NoMaxMin < 70)
            {
               feedbackMsg =  "You needs extra help from your Professor";
            }
         else if (averageGrade_NoMaxMin > 70 || averageGrade_NoMaxMin <80)
            {
               feedbackMsg = "You are  doing a good job.";
            }
         else if (averageGrade_NoMaxMin > 80 || averageGrade_NoMaxMin <90)
            {
               feedbackMsg = "You are doing a great job!";
            }
         else if (averageGrade_NoMaxMin > 90 && averageGrade_NoMaxMin < 100)
            {
               feedbackMsg = "You are  doing an excellent job, keep it up!!!";
```

```
            }

        //Return the feedbackMsg to the main method.
         return feedbackMsg;

    }
    //Begin the second submethod, fillGradesArray.
     public static double fillGradesArray()
       {
       //Set maxArrayLength, index, grade, strGrade, averageGrade_NoMaxMin, and initialize allGradesArray.
          final int maxArrayLength = 10;
          int[] allGradesArray = new int[maxArrayLength];
          int index = 0;
          int grade = -1;
          String strGrade="";
          double averageGrade_NoMaxMin = 0.00;

       //Use try and catch within a do loop and do loop within a for loop to prompt and validate the grade user inputs.
       for (index = 0; index <= (maxArrayLength); index++)
          {
          do
             {
             //Validate user inputs.
             try {
               grade = JOptionPane.showInputDialog("Please enter grade(0-100), #"+(index+1)+": ");

             }
             catch (NumberFormatException e)
                {
             grade = -1;
             JOptionPane.showMessageDialog(null, "ERROR! Enter a grade between 0 and 100");
                }

             }
          while(grade < 0 || grade > 100);
            strGrade = grade+", "+strGrade;

         allGradesArray[index] = grade;

          }
     //Set maxGrade, minGrade, readAllGrades, sumAllGrades, output, strOutput, and feedbackMsg.
          int maxGrade = allGradesArray[0];
          int minGrade = allGradesArray[0];
          int readAllGrades; //*** 1 bug missing initialization for variable declaration
          int sumAllGrades = 0;
          String feedbackMsg ="";


          int output = 0;
         String strOutput ="";
```

```
101
102       //Begin a for loop to find the maximum and the minimum values.
103          for (index = 1; index <= (maxArrayLength); index++)
104             {
105                grade=allGradesArray[index];
106                if (allGradesArray[index] > maxGrade)
107                   {
108                      maxGrade = allGradesArray[index];
109
110                   }
111                else if (allGradesArray[index] < minGrade)
112                   {
113                      minGrade = allGradesArray[index]
114                   }
115
116              //Calculate the sum of all grades.
117               sumAllGrades += grade;
118
119             //Calculate the average grade of all grades with no maximum or minimum values.
120               averageGrade_NoMaxMin = (float) (sumAllGrades - maxGrade - minGrade)/(maxArrayLength - 2);
121
122             }
123                feedbackMsg = printGradeMsg(averageGrade_NoMaxMin);
124
125             //Print the report: strGrade, averageGrade_NoMaxMin, and the feedbackMsg.
126            JOptionPane.showMessageDialog(null,"\nThe grades you entered are: " + strGrade + "\nThe student's average grade wit
hout the maximum and the minimum grade is: " + String.format("%.2f", averageGrade_NoMaxMin)+ "\nThe feedback is: "+feedbackMsg);
127
128           //Return the averageGrade_NoMaxMin to the main method.
129           return averageGrade_NoMaxMin
130
131       }
132    }
133
```