ReadMe

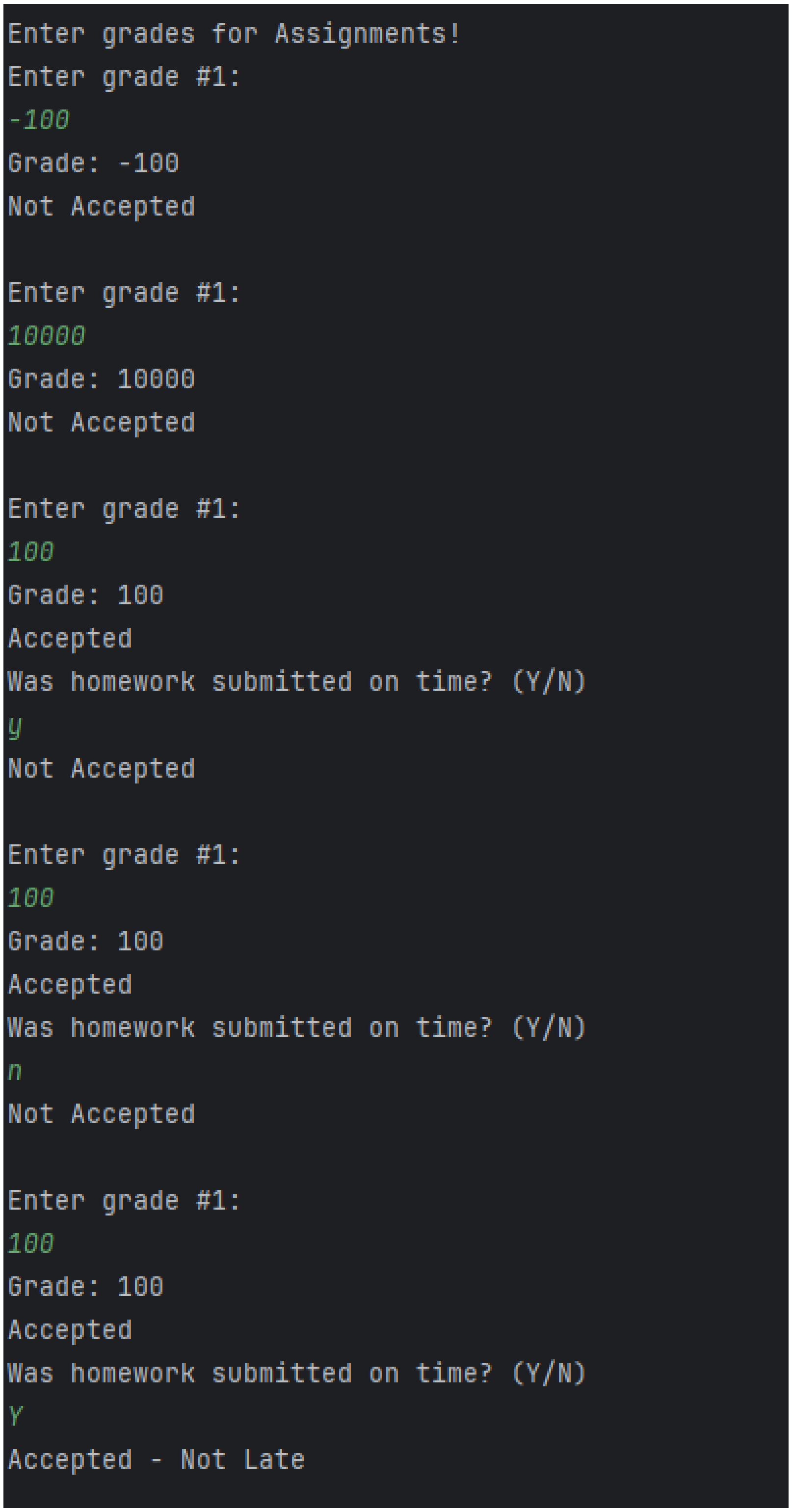
Enter grades between 0 and 100. There is error handling. For late assignments and project, type "Y" for yes or "N" for no. You will lose 20% if assignment is late – this can easily be modified since there were no given values for deductions but the logic is there. For quizzes, it cannot be late. Therefore, no late option. After each section, I list the grades. At the end of all the grade entering, I give the overall grade and corresponding letter. I did everything equal weight value. You would need to change the logic if it is otherwise different.

Some info about my code and approach:

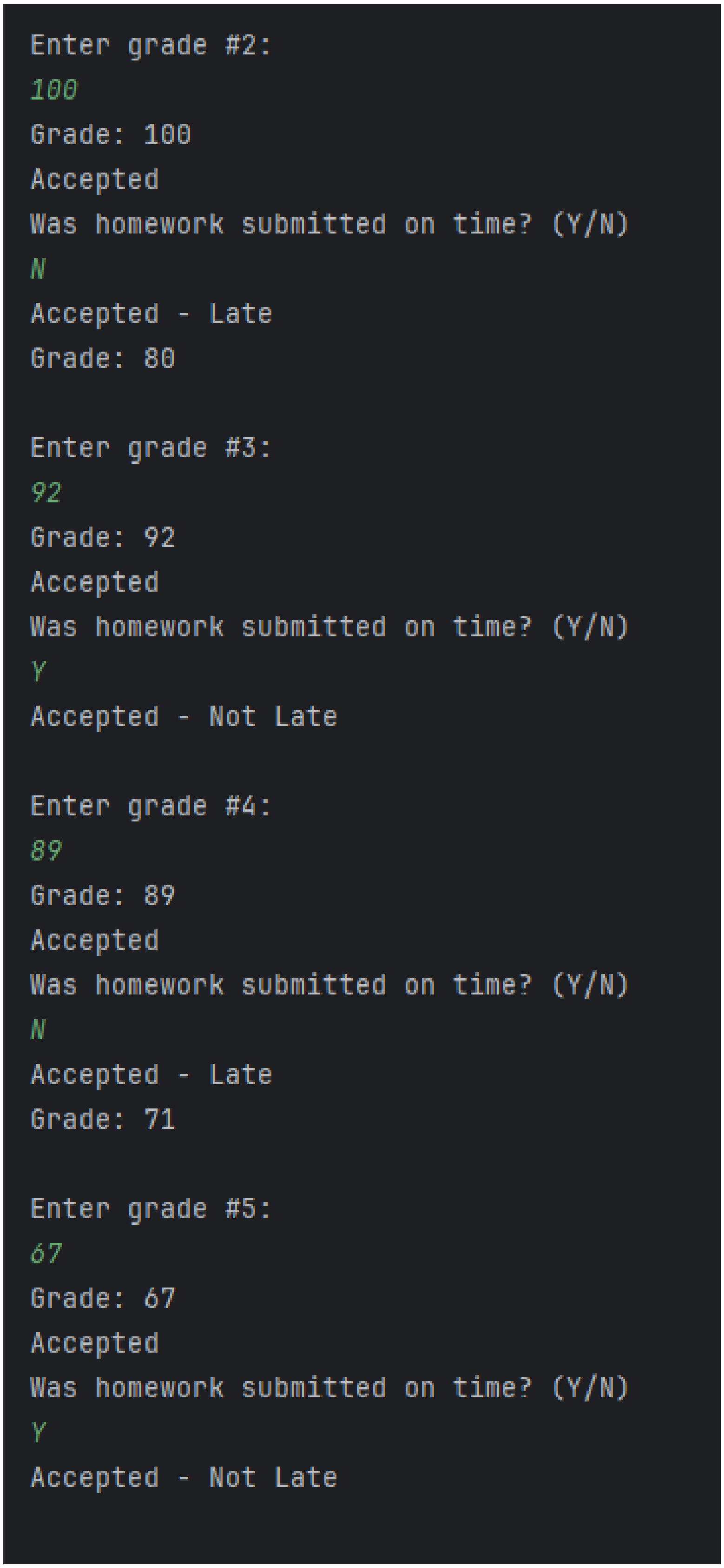
1. I have some error handling for number, letters and wrong inputs.
2. The lowest grade in quizzes will be dropped.
3. I assumed letter grades as it was not given and I did not add too many options but there concept is there. Things can easily be modified.
4. I have inheritance and case statements. Combining inheritance and arrays was seemingly impossible to do – I went to Pear Tutoring and had help without any luck.
5. I was experimenting with encapsulation but I did not do what I was hoping to do with encapsulation.
6. I have multiple java files.
7. I have a code selection

Pictures

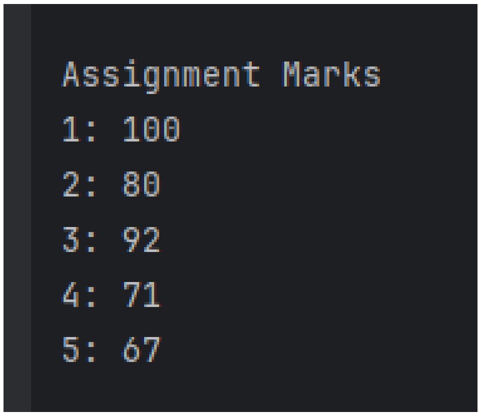
I am showing error handling for all situations. Note that I need capital Y or N



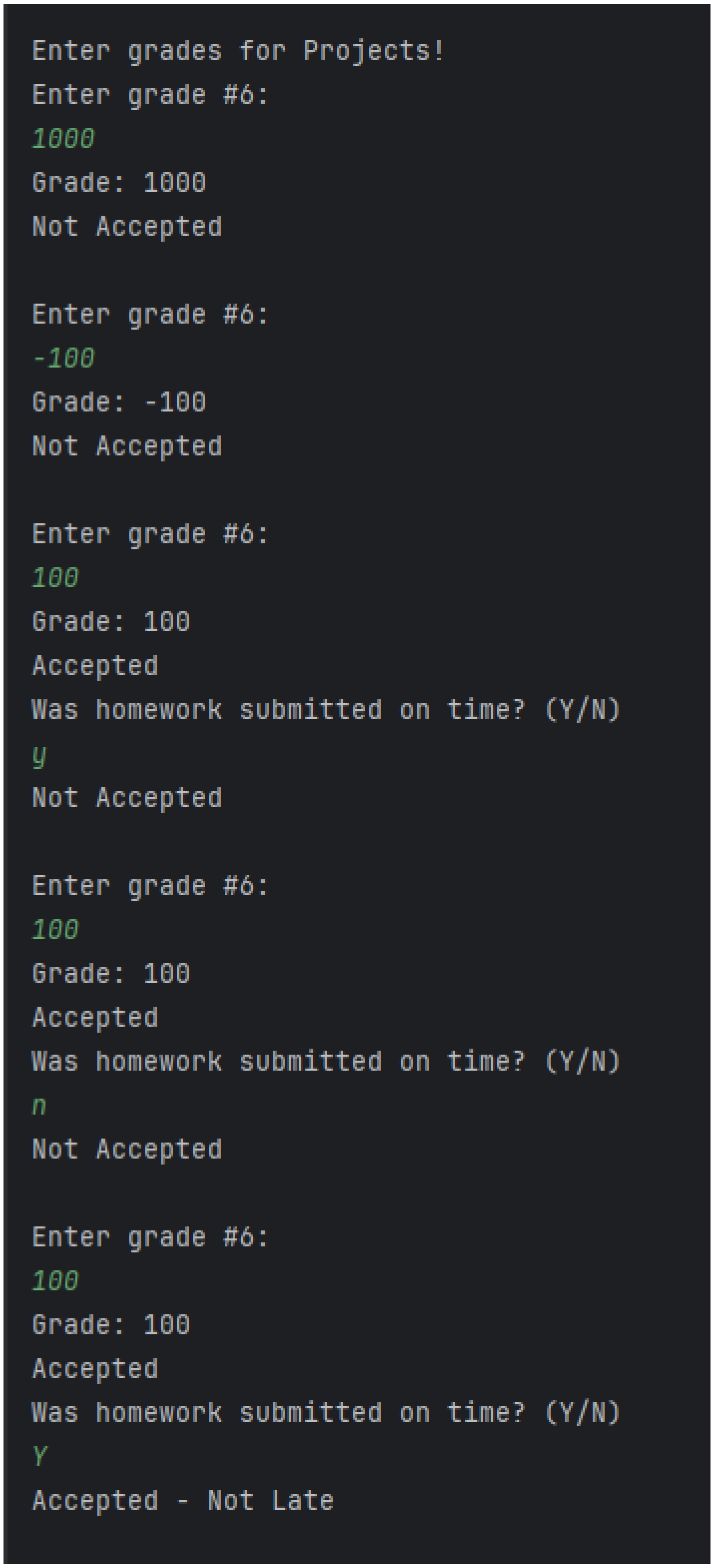
```
Enter grades for Assignments!
Enter grade #1:
-100
Grade: -100
Not Accepted

Enter grade #1:
10000
Grade: 10000
Not Accepted

Enter grade #1:
100
Grade: 100
Accepted
Was homework submitted on time? (Y/N)
y
Not Accepted

Enter grade #1:
100
Grade: 100
Accepted
Was homework submitted on time? (Y/N)
n
Not Accepted

Enter grade #1:
100
Grade: 100
Accepted
Was homework submitted on time? (Y/N)
Y
Accepted - Not Late
```



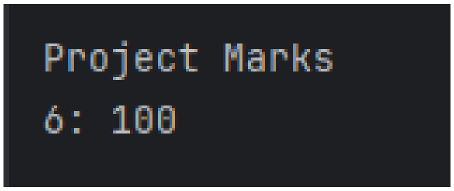
```
Enter grade #2:
100
Grade: 100
Accepted
Was homework submitted on time? (Y/N)
N
Accepted - Late
Grade: 80

Enter grade #3:
92
Grade: 92
Accepted
Was homework submitted on time? (Y/N)
Y
Accepted - Not Late

Enter grade #4:
89
Grade: 89
Accepted
Was homework submitted on time? (Y/N)
N
Accepted - Late
Grade: 71

Enter grade #5:
67
Grade: 67
Accepted
Was homework submitted on time? (Y/N)
Y
Accepted - Not Late
```



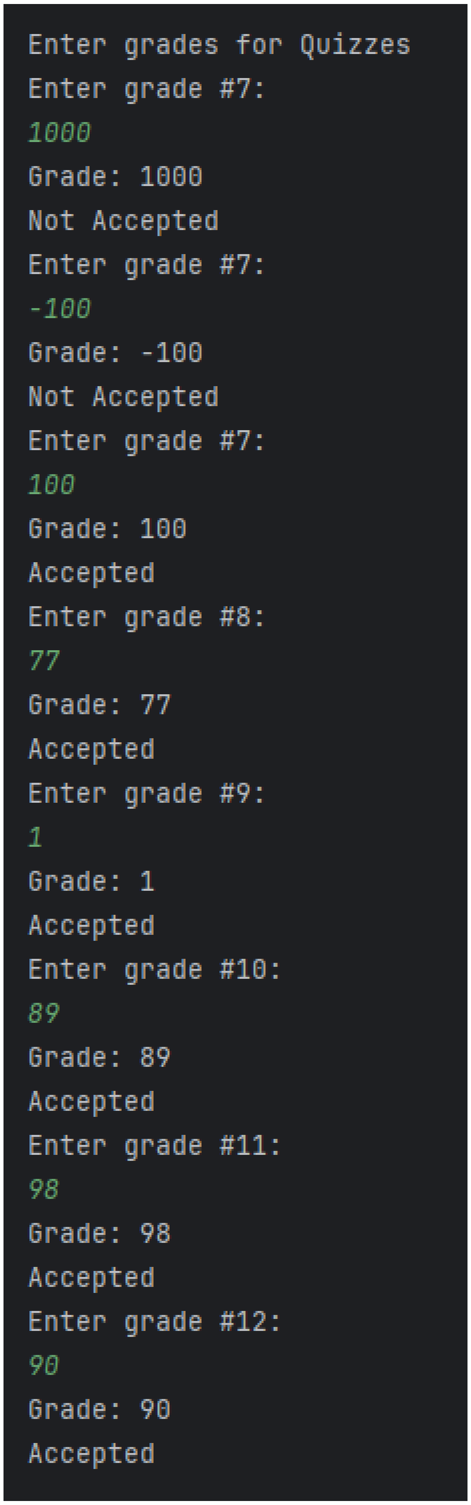
```
Assignment Marks
1: 100
2: 80
3: 92
4: 71
5: 67
```



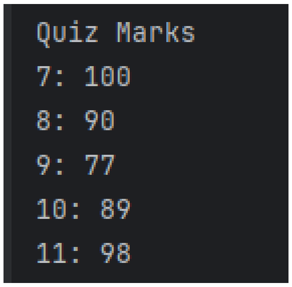
```
Enter grades for Projects!
Enter grade #6:
1000
Grade: 1000
Not Accepted

Enter grade #6:
-100
Grade: -100
Not Accepted

Enter grade #6:
100
Grade: 100
Accepted
Was homework submitted on time? (Y/N)
y
Not Accepted

Enter grade #6:
100
Grade: 100
Accepted
Was homework submitted on time? (Y/N)
n
Not Accepted

Enter grade #6:
100
Grade: 100
Accepted
Was homework submitted on time? (Y/N)
Y
Accepted - Not Late
```



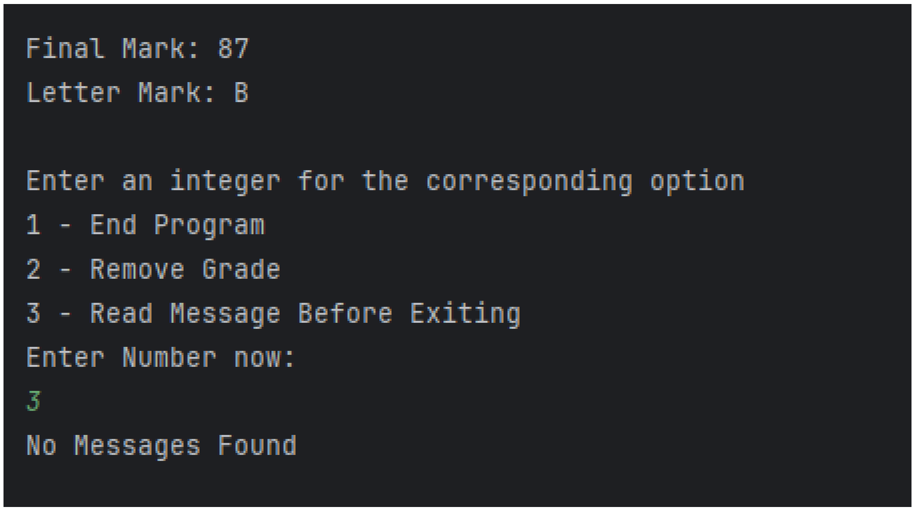
```
Project Marks
6: 100
```



```
Enter grades for Quizzes
Enter grade #7:
1000
Grade: 1000
Not Accepted
Enter grade #7:
-100
Grade: -100
Not Accepted
Enter grade #7:
100
Grade: 100
Accepted
Enter grade #8:
77
Grade: 77
Accepted
Enter grade #9:
1
Grade: 1
Accepted
Enter grade #10:
89
Grade: 89
Accepted
Enter grade #11:
98
Grade: 98
Accepted
Enter grade #12:
90
Grade: 90
Accepted
```



```
Quiz Marks
7: 100
8: 90
9: 77
10: 89
11: 98
```



```
Final Mark: 87
Letter Mark: B

Enter an integer for the corresponding option
1 - End Program
2 - Remove Grade
3 - Read Message Before Exiting
Enter Number now:
3
No Messages Found
```

Code

```
public class Main {

  public static void main (String[] ARGS) {
    System.out.println("Welcome to our class calculator!");

    allNumbers Number1 = new allNumbers();

    //I need error-handling for letter in number areas
    System.out.println("Enter grades for Assignments!");
    for(int i = 1; i < 6; i++) {
      Number1.assignmentNumber(i);
    }

    System.out.println("\nAssignment Marks");
    for(int i = 1; i < 6; i++) {
      Number1.assignmentMarks(i);
    }

    System.out.println("\nEnter grades for Projects!");
    for(int i = 6; i < 7; i++) {
      Number1.projectNumber(i);
    }

    System.out.println("\nProject Marks");
    for(int i = 6; i < 7; i++) {
      Number1.projectMarks(i);
    }

    //I need to lose the lowest mark and add 1 to i
    System.out.println("\nEnter grades for Quizzes");
    for(int i = 7; i < 13; i++) {
      Number1.quizNumber(i);
    }

    System.out.println("\nQuiz Marks");
    for(int i = 7; i < 12; i++) {
      Number1.quizMarks(i);
    }

    Number1.totalGrade();

    selection selectionOptions = new selection();
```

```
        selectionOptions.setOption();
        selectionOptions.caseStatement();
    }
}
```

```
import java.util.Objects;
import java.util.Scanner;

public class allNumbers {
    int[] marksArray = new int[12];
    int input_grade;

    public void assignmentNumber(int i) {
        input(i);
    }

    public void assignmentMarks(int i) {
        System.out.println((i) + ": " + marksArray[i]);
    }

    public void projectNumber(int i) {
        input(i);
    }

    public void projectMarks(int i) {
        System.out.println((i) + ": " + marksArray[i]);
    }

    public void quizNumber(int i) {
        inputQUIZ(i);
    }

    public void quizMarks(int i) {
        System.out.println((i) + ": " + marksArray[i--]);
    }

    public void input(int i) {

        while (true) {

            System.out.println("Enter grade #" + i + ": ");
            Scanner myObj = new Scanner(System.in);
            input_grade = Integer.parseInt(myObj.nextLine());
            System.out.println("Grade: " + input_grade);

            //verify number between 0 and 100 - be part of the other question
            if ((input_grade > -1) && (input_grade < 101))
                System.out.println("Accepted");
            else {
                System.out.println("Not Accepted\n");
                continue;
            }
        //was the homework submitted in time?
        System.out.println("Was homework submitted on time? (Y/N)");
        String choice = myObj.nextLine();  // Read user input

        if ((Objects.equals(choice, "Y")) || (Objects.equals(choice, "N"))) {
            if (Objects.equals(choice, "Y"))
                System.out.println("Accepted - Not Late\n");

            if (Objects.equals(choice, "N")) {
                System.out.println("Accepted - Late");
                System.out.println("Grade: " + (int) (input_grade * 0.80) + "\n");
                input_grade = (int) (input_grade * 0.80);
            }

            //System.out.println("\n\n");
            int j = i--;
            marksArray[j] = input_grade;
            //System.out.println(j + "  " + input_grade);
            //System.out.println("class value:" + marksArray[j]);
            return;
        } else
            System.out.println("Not Accepted\n");

    }

}

public void inputQUIZ(int i) {
    if(i!=12) {
        while (true) {

            System.out.println("Enter grade #" + i + ": ");
            Scanner myObj = new Scanner(System.in);
            input_grade = Integer.parseInt(myObj.nextLine());
            System.out.println("Grade: " + input_grade);

            //verify number between 0 and 100 - be part of the other question
            if ((input_grade > -1) && (input_grade < 101))
                System.out.println("Accepted");
            else {
                System.out.println("Not Accepted");
                continue;
            }

            //System.out.println("\n\n");
            int j = i--;
            marksArray[j] = input_grade;
            //System.out.println(j + "  " + input_grade);
```

```
                return;
            }
        }
        else{
            while(true) {
                int placeHolder;
                System.out.println("Enter grade #" + i + ": ");
                Scanner myObj = new Scanner(System.in);
                input_grade = Integer.parseInt(myObj.nextLine());
                System.out.println("Grade: " + input_grade);

                //verify number between 0 and 100 - be part of the other question
                if ((input_grade > -1) && (input_grade < 101))
                    System.out.println("Accepted");
                else {
                    System.out.println("Not Accepted");
                    continue;
                }

                for (int j = 6; j < 12; j++) {
                    if (input_grade > marksArray[j]) {
                        placeHolder = marksArray[j];
                        marksArray[j] = input_grade;
                        input_grade = placeHolder;
                    }
                }
                return;
            }
        }
    }

    public void totalGrade(){
        int avg, sum = 0;
        int length = marksArray.length;
        for(int i = 0; i < marksArray.length; i++) {
            sum += marksArray[i];
        }

        avg = sum / (length-1);
        //System.out.println("\nFinal Mark: "+ avg);

        letterGrades letters1 = new letterGrades();
        letters1.letterSelection(avg);
    }

}
```

```
import java.util.Scanner;

public class letterGrades {
  int option;
  public void letterSelection(int gradeAverage) {

    System.out.println("\n\nFinal Mark: " + gradeAverage);

    //letter grade
    if (gradeAverage < 60)
      System.out.println("Letter Mark: F\n");

    else if (gradeAverage < 70)
      System.out.println("Letter Mark: D\n");

    else if (gradeAverage < 80)
      System.out.println("Letter Mark: C\n");

    else if (gradeAverage < 90)
      System.out.println("Letter Mark: B\n");

    else if (gradeAverage < 100)
      System.out.println("Letter Mark: A\n");

    else if (gradeAverage == 100)
      System.out.println("Letter Mark A - Perfect Marks\n");
  }

  public void setOption() {
    System.out.println("Enter an integer for the corresponding option \n1 - End Program\n2 - Remove
Grade" +
        "\n3 - Read Message Before Exiting \nEnter Number now:");
    Scanner myObj = new Scanner(System.in);
    option = Integer.parseInt(myObj.nextLine());
  }

  public void caseStatement(){

    //Refer to readme file about this
    switch(option){
      case 1 :
        System.out.println("Ending Program");
        break;
      case 2 :
        //Does not actually remove grade - outside scope of class assignment
        System.out.println("Remove Grade");
        break;
      case 3 :
        //we can add a messaging system here if we wanted to expand code
        System.out.println("No Messages Found");
        break;
      default :
        System.out.println("Invalid");
    }
  }
}
```

```
public class selection extends letterGrades{
   @Override
   public void setOption() {
      super.setOption();
   }

   //switch statement in child class: submit info / delete info / ect.

   @Override
   public void caseStatement() {
      super.caseStatement();
   }
}
```