FINAL GRADE CALCULATOR

Anna Roszkiewicz

WHAT IT DOES

- Assigning grades to students

```
Please input grades for Math Calculus Exam:
John Smith: 3
Robert Johnson: 3,5
Mary Williams: 5
```

- Calculating the class grade average for each test
- Calculating each student's weighted grade average for each subject
- Assigning final grades based on grade averages

- Calculating final grade average

```
Final grade average for John Smith: 3,50
```

- Displaying grades in a list format

```
Showing all grades for Math Trigonometry Quiz
John Smith: 3.0
Robert Johnson: 2.0
Mary Williams: 6.0
```

HOW IT WORKS

```
public interface Person
{
    public String FullName();
    public void DisplayGrades();
    public double SubjectAverage(Subject subject);
    public void DisplayFinalGrades();
    public double FinalGradeAverage();
    public void DisplayFinalGradeAverage();
}
```

```
public interface Grade
{
    public void ChangeGrade(double newValue);
    public double GradeValue();
    public String StudentName();
    public String TestName();
    public Subject Subject();
    public int Weight();
}
```

```
public interface Test
{
    public void AssignGrades();
    public void DisplayGrades();
    public double TestAverage();
    public void DisplayTestAverage();
    public String Name();
    public Subject TestSubject();
    public int TestWeight();
}
```

```
public interface SchoolSubject
{
    public void StudentList();
    public void AssignFinalGrades();
    public int classSize();
    public String SubjectName();
}
```

WHAT I LEARNED

WHAT I LEARNED
(MORE SPECIFIC THIS TIME)

ArrayList

```
import java.util.ArrayList;
```

```
ArrayList <SingleGrade> grades;
ArrayList <Integer> finalGrades;
```

```
this.grades = new ArrayList <SingleGrade>();
this.finalGrades = new ArrayList <Integer>();
```

Methods: add(), get(), size()

PROBLEMS I HAD

Encapsulation – accessing private attributes by other classes

```
private String name;
private Subject subject;
private int weight;
ArrayList <SingleGrade> grades;
```

Solved by adding new methods

Is there a better solution?

```
public String StudentName()
{
    return this.student.FullName();
}

public String TestName()
{
    return this.test.Name();
}

public Subject Subject()
{
    return this.test.TestSubject();
}

public int Weight()
{
    return this.test.TestWeight();
}
```

WHAT COULD BE IMPROVED

- Making the program more versatile (support for various grading systems)

```
public void AssignFinalGrades()
{
    for(Student student: this.students)
    {
        double gradeAverage = student.SubjectAverage(this);
        int finalGrade = 0;
        if(gradeAverage<1.70) finalGrade=1;
        else if(gradeAverage<2.60) finalGrade=2;
        else if(gradeAverage<3.60) finalGrade=3;
        else if(gradeAverage<4.60) finalGrade=4;
        else if(gradeAverage<5.40) finalGrade=5;
        else finalGrade=6;

        student.finalGrades.add(finalGrade);
    }
}
```

- Additional features e.g. student ranking
- Project structure