JAVA-Revision

Qustion 1

- code

java

```java
import java.util.Scanner;

public class RetailCalculator {

    // Method to read product number
    public static int readProductNumber(Scanner input) {
        System.out.print("Enter product number (1-5, or -9999 to stop): ");
        return input.nextInt();
    }

    // Method to determine price based on product number
    public static double getProductPrice(int productNumber) {
        switch (productNumber) {
            case 1:
                return 2.98;
            case 2:
                return 4.50;
            case 3:
                return 9.98;
            case 4:
                return 4.49;
            case 5:
                return 6.87;
            default:
                System.out.println("Invalid product number!");
                return 0.0;
        }
    }

    // Method to read quantity sold
    public static int readQuantity(Scanner input) {
        System.out.print("Enter quantity sold: ");
        return input.nextInt();
    }

    // Method to calculate total for this product
    public static double calculateTotal(double price, int quantity) {
        return price * quantity;
    }
```

```
    // Method to display total retail value
    public static void displayTotal(double total) {
        System.out.printf("Total retail value of all products: RM%.2f\n",
total);
    }

    public static void main(String[] args) {
        Scanner input = new Scanner(System.in);
        double grandTotal = 0.0;
        int productNumber;

        // Sentinel-controlled loop
        while (true) {
            productNumber = readProductNumber(input);
            if (productNumber == -9999)
                break;

            double price = getProductPrice(productNumber);
            if (price == 0)
                continue; // skip invalid product

            int quantity = readQuantity(input);
            double total = calculateTotal(price, quantity);
            grandTotal += total;

            System.out.printf("Product %d: RM%.2f x %d = RM%.2f\n",
                    productNumber, price, quantity, total);
        }

        // Display final result
        displayTotal(grandTotal);
        input.close();
    }
}
```

- output

```
PS C:\Users\zxYch\Desktop\3300lab> cd "c:\Users\zxYch\Desktop
lculator }
Enter product number (1-5, or -9999 to stop): 2
Enter quantity sold: 3
Product 2: RM4.50 x 3 = RM13.50
Enter product number (1-5, or -9999 to stop): -9999
Total retail value of all products: RM13.50
PS C:\Users\zxYch\Desktop\3300lab>
```

question 2

- code

```
// Base class: GradedActivity
class GradedActivity {
    protected double score; // holds the numeric score

    // Set score
    public void setScore(double score) {
        this.score = score;
    }

    // Get score
    public double getScore() {
        return score;
    }

    // Determine grade based on score
    public char getGrade() {
        if (score >= 90)
            return 'A';
        else if (score >= 80)
            return 'B';
        else if (score >= 70)
            return 'C';
        else if (score >= 60)
            return 'D';
        else
            return 'F';
    }
}
```

```
// Subclass: Essay
class Essay extends GradedActivity {
    private double grammar; // 0-30
    private double spelling; // 0-20
    private double correctLength; // 0-20
    private double content; // 0-30

    // Constructor
    public Essay(double grammar, double spelling, double correctLength,
double content) {
        this.grammar = grammar;
        this.spelling = spelling;
        this.correctLength = correctLength;
        this.content = content;

        // use super to update parent score
        double totalScore = grammar + spelling + correctLength + content;
        super.setScore(totalScore);
    }

    // Display breakdown and total
    public void displayEssayDetails() {
        System.out.println("Essay Score Details:");
        System.out.println("Grammar: " + this.grammar);
        System.out.println("Spelling: " + this.spelling);
        System.out.println("Correct Length: " + this.correctLength);
        System.out.println("Content: " + this.content);
        System.out.println("Total Score: " + super.getScore());
        System.out.println("Final Grade: " + super.getGrade());
    }
}

// Demo class
public class EssayTest {
    public static void main(String[] args) {
        // Example: Grammar 30, Spelling 20, Correct Length 12, Content 10 =
total 72
        Essay essay1 = new Essay(30, 20, 12, 10);
        essay1.displayEssayDetails();
    }
}
```

- output

```
Total retail value of all products: RM15.50
PS C:\Users\zxYch\Desktop\3300lab> cd "c:\User
Essay Score Details:
Grammar: 30.0
Spelling: 20.0
Correct Length: 12.0
Content: 10.0
Total Score: 72.0
Final Grade: C
PS C:\Users\zxYch\Desktop\3300lab>
```

Question3

- code

```
// Base class: GradedActivity
class GradedActivity {
    protected double score;

    // Set score
    public void setScore(double score) {
        this.score = score;
    }

    // Get score
    public double getScore() {
        return score;
    }

    // Get grade based on score
    public char getGrade() {
        if (score >= 90)
            return 'A';
        else if (score >= 80)
            return 'B';
        else if (score >= 70)
            return 'C';
        else if (score >= 60)
            return 'D';
        else
            return 'F';
    }
}
```

```
// Subclass: Essay
class Essay extends GradedActivity {
    private double grammar;
    private double spelling;
    private double correctLength;
    private double content;

    // Overload setScore(): accept 4 parameters
    public void setScore(double grammar, double spelling, double
correctLength, double content) {
        this.grammar = grammar;
        this.spelling = spelling;
        this.correctLength = correctLength;
        this.content = content;

        // Use super to update total score
        double totalScore = grammar + spelling + correctLength + content;
        super.setScore(totalScore);
    }

    // Override setScore(double): disallow single-parameter use
    @Override
    public void setScore(double score) {
        System.out.println(
                "⚠️ Error: Class Essay is not allowed to call setScore()
with a single value (calling with parameter "
                        + score + ").");
    }

    // Override getGrade(): force F if any component is 0
    @Override
    public char getGrade() {
        if (grammar == 0 || spelling == 0 || correctLength == 0 || content
== 0) {
            return 'F';
        } else {
            return super.getGrade();
        }
    }

    // Show breakdown
    public void showDetails() {
        System.out.println("\nEssay Score Breakdown:");
        System.out.println("Grammar: " + grammar);
        System.out.println("Spelling: " + spelling);
        System.out.println("Correct Length: " + correctLength);
        System.out.println("Content: " + content);
```

```
        System.out.println("Total Score: " + super.getScore());
        System.out.println("Final Grade: " + getGrade());
    }
}

// Demo class
public class EssayOverloadOverrideTest {
    public static void main(String[] args) {
        Essay essay = new Essay();

        // Case 1: valid input using 4 parameters
        essay.setScore(30, 20, 12, 10);
        essay.showDetails();

        // Case 2: invalid call with one parameter
        essay.setScore(-2);

        // Case 3: one part is 0 → should force F
        essay.setScore(30, 0, 20, 30);
        essay.showDetails();
    }
}
```

- output

```
PS C:\Users\zxYch\Desktop\3300lab> cd "c:\Users\zxYch\Desktop\3300lab\" ; if ($?) { javac EssayOverloadOverride
 EssayOverloadOverrideTest }

Essay Score Breakdown:
Grammar: 30.0
Spelling: 20.0
Correct Length: 12.0
Content: 10.0
Total Score: 72.0
Final Grade: C
?? Error: Class Essay is not allowed to call setScore() with a single value (calling with parameter -2.0).

Essay Score Breakdown:
Grammar: 30.0
Spelling: 0.0
Correct Length: 20.0
Content: 30.0
Total Score: 80.0
Final Grade: F
PS C:\Users\zxYch\Desktop\3300lab>
```