Jobsheet 5 Praktikum Daspro

Yanuar Thaif Chalil Candra

2241720004 / TI-1I / 29

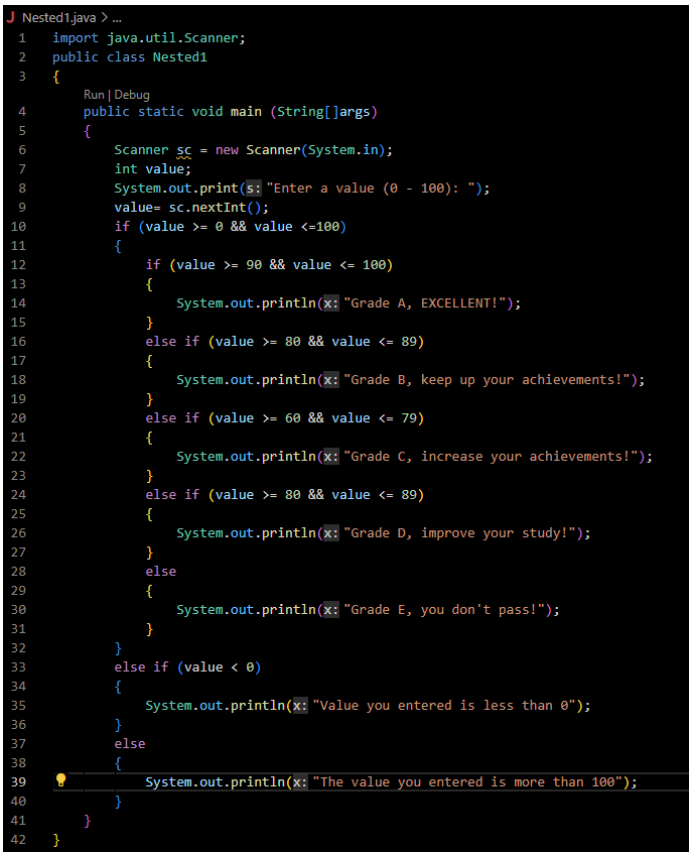
Experiment 1

```java
import java.util.Scanner;
public class Nested1
{
    Run | Debug
    public static void main (String[]args)
    {
        Scanner sc = new Scanner(System.in);
        int value;
        System.out.print(s: "Enter a value (0 - 100): ");
        value= sc.nextInt();
        if (value >= 0 && value <=100)
        {
            if (value >= 90 && value <= 100)
            {
                System.out.println(x: "Grade A, EXCELLENT!");
            }
            else if (value >= 80 && value <= 89)
            {
                System.out.println(x: "Grade B, keep up your achievements!");
            }
            else if (value >= 60 && value <= 79)
            {
                System.out.println(x: "Grade C, increase your achievements!");
            }
            else if (value >= 80 && value <= 89)
            {
                System.out.println(x: "Grade D, improve your study!");
            }
            else
            {
                System.out.println(x: "Grade E, you don't pass!");
            }
        }
        else
        {
            System.out.println(x: "The value you entered is invalid");
        }
    }
}
```

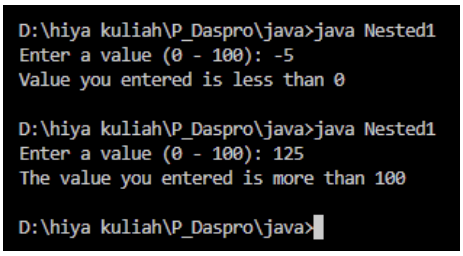
```
D:\hiya kuliah\P_Daspro\java>java Nested1
Enter a value (0 - 100): 80
Grade B, keep up your achievements!

D:\hiya kuliah\P_Daspro\java>java Nested1
Enter a value (0 - 100): 95
Grade A, EXCELLENT!

D:\hiya kuliah\P_Daspro\java>java Nested1
Enter a value (0 - 100): 35
Grade E, you don't pass!

D:\hiya kuliah\P_Daspro\java>
```

Question 1

1. if (value >= 0 && value <= 100), if used to use a syntax if meet specified condition, value >=0 means the value that inputted is more than or equal to 0, && means And, value <=100 means the value that inputted is less than or equal to 100

2. 

```java
import java.util.Scanner;
public class Nested1
{
    Run | Debug
    public static void main (String[]args)
    {
        Scanner sc = new Scanner(System.in);
        int value;
        System.out.print(s: "Enter a value (0 - 100): ");
        value= sc.nextInt();
        if (value >= 0 && value <=100)
        {
            if (value >= 90 && value <= 100)
            {
                System.out.println(x: "Grade A, EXCELLENT!");
            }
            else if (value >= 80 && value <= 89)
            {
                System.out.println(x: "Grade B, keep up your achievements!");
            }
            else if (value >= 60 && value <= 79)
            {
                System.out.println(x: "Grade C, increase your achievements!");
            }
            else if (value >= 80 && value <= 89)
            {
                System.out.println(x: "Grade D, improve your study!");
            }
            else
            {
                System.out.println(x: "Grade E, you don't pass!");
            }
        }
        else if (value < 0)
        {
            System.out.println(x: "Value you entered is less than 0");
        }
        else
        {
            System.out.println(x: "The value you entered is more than 100");
        }
    }
}
```



```
D:\hiya kuliah\P_Daspro\java>java Nested1
Enter a value (0 - 100): -5
Value you entered is less than 0

D:\hiya kuliah\P_Daspro\java>java Nested1
Enter a value (0 - 100): 125
The value you entered is more than 100

D:\hiya kuliah\P_Daspro\java>
```

3. The result is Grade E, you don't pass, because it's written as >= 0 or <= 100, so there is no limitation after 100, so it doesn't meet any condition whereby it goes into 'else' condition

Experiment 2

```java
Nested2.java > Nested2 > main(String[])
    Run | Debug
    public static void main (String[]args)
    {
        Scanner sc = new Scanner(System.in);
        String category;
        int income, netSalary;
        double tax = 0;
        System.out.print(s: "Enter a category: ");
        category = sc.nextLine();
        System.out.print(s: "Enter the amount of income: ");
        income = sc.nextInt();
        if (category.equalsIgnoreCase(anotherString: "worker"))
        {
            if (income <= 2000000)
            {
                tax = 0.1;
            }
            else if (income <= 3000000)
            {
                tax = 0.15;
            }
            else
            {
                tax = 0.2;
            }
            netSalary = (int) (income - (income * tax));
            System.out.println("The net salary you will recieve: " + netSalary);
        }
        else if (category.equalsIgnoreCase(anotherString: "businessman"))
        {
            if (income <= 2500000)
            {
                tax = 0.15;
            }
            else if (income <= 3500000)
            {
                tax = 0.2;
            }
            else
            {
                tax = 0.25;
            }
            netSalary = (int) (income - (income * tax));
            System.out.println("The net salary you will recieve: " + netSalary);
        }
        else
        {
            System.out.println(x: "The category you entered is wrong");
        }
    }
```

```
D:\hiya kuliah\P_Daspro\java>java Nested2
Enter a category: worker
Enter the amount of income: 2369000
The net salary you will recieve: 2013650

D:\hiya kuliah\P_Daspro\java>java Nested2
Enter a category: businessman
Enter the amount of income: 3000000
The net salary you will recieve: 2400000

D:\hiya kuliah\P_Daspro\java>
```

Question 2

1. Because the data value is using integers

```
D:\hiya kuliah\P_Daspro\java>java Nested2
Enter a category: worker
Enter the amount of income: 2048485
The net salary you will recieve: 1741212
```

2. Int is used to shrink down the result if it's a decimal number to integers number
3. equalsIgnoreCase means that it will ignore the use of case letter, so the result will same even if it's in case or not

```
D:\hiya kuliah\P_Daspro\java>java Nested2
Enter a category: BUSINESSMAN
Enter the amount of income: 2000000
The net salary you will recieve: 1700000

D:\hiya kuliah\P_Daspro\java>
```

4. Because equals means that the condition is only met if the inputted category is the same as written in the syntax

```
D:\hiya kuliah\P_Daspro\java>java Nested2
Enter a category: BUSINESSMAN
Enter the amount of income: 2000000
The category you entered is wrong

D:\hiya kuliah\P_Daspro\java>
```