Basic Programming Practicum Experiment Job sheet 5 Meeting 6

From:

AL AZHAR RIZQI RIFA'I FIRDAUS

Class:

1 I

Absence:

01

Major:

Information Technology

Study Program:

Informatic Engineering

Experiment 1

1. Open a text editor. Create a new file, name it Nested1.java

2. Write the basic structure of the Java programming language which contains the main() function

```
2
3   public class Nested1 {
        Run | Debug
4       public static void main(String[] args) {
5
6
7
8       }
9   }
10
```

3. Add the Scanner library.

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

4. Make a Scanner declaration with the name sc

```
Scanner sc = new Scanner(System.in);
```

5. Create an int variable with the name value

```
int value;
```

6. Write down the syntax for entering the value from keyboard

```
System.out.print(s: "Enter a value (0 - 100) : ");
value = sc.nextInt();
```

7. Create a nested selection structure. The first check is used to ensure that the value entered is in the range 0 - 100. If the value is in the range 0 - 100, then a student graduation status will be checked, i.e. if the value is between 90 - 100 then the value is A, if the value is between 80 - 89 then the value is B, if the value is between 60 - 79 then the value is C, if the value is between 50 - 59 then the value is D, and if the value is between 0 - 49 then the value is E. Whereas if the value is outside the range 0 - 100 , then displayed information stating that the value entered is invalid.

```
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 achievement!");
    } else if (value >= 60 && value <= 79) {
        System.out.println(x: "Grade C, increase your achievement!");
    } else if (value >= 50 && value <= 59) {
        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!");
}
```

8. Compile and run the program. Observe the results!

```
C:\Users\Al Azhar Rizqi\Documents\College\Basic Programming Practicum\Meet 6\coding>javac Nested1.java

C:\Users\Al Azhar Rizqi\Documents\College\Basic Programming Practicum\Meet 6\coding>java Nested1.java
Enter a value (0 - 100) : 90
Grade A, EXCELLENT!

C:\Users\Al Azhar Rizqi\Documents\College\Basic Programming Practicum\Meet 6\coding>
```

Questions!

1. Describe the following syntax functions! if (value >= 0 && value <= 100)

2. Modify the program code in Experiment 1 so that if the entered value is less than 0 the output "Value you entered is less than 0" and if the entered value is more than 100 the output will display "The value you entered is more than 100"!

3. Change the && operator to || on if (value >= 0 && value <= 100). Compile and run the program by entering the value = 105 using keyboard. Watch what happened! Why is the result like that?

Answer!

1. Its function is to combine the shading between more than equal to 0 and less than equal to 100 which results in 0 to 100
2. Code :

```java
    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 achievement!");
            } else if (value >= 60 && value <= 79) {
                System.out.println(x: "Grade C, increase your achievement!");
            } else if (value >= 50 && value <= 59) {
                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 if (value > 100) {
            System.out.println(x: "The value you entered is more than 100");
        } else {
            System.out.println(x: "The value you entered is invalid!");
        }

    }
}
```

Result :

```
C:\Users\Al Azhar Rizqi\Documents\College\Basic Programming Practicum\Meet 6\coding>java Nested1.java
Enter a value (0 - 100) : -23
Value you entered is less than 0

C:\Users\Al Azhar Rizqi\Documents\College\Basic Programming Practicum\Meet 6\coding>java Nested1.java
Enter a value (0 - 100) : 92384
The value you entered is more than 100

C:\Users\Al Azhar Rizqi\Documents\College\Basic Programming Practicum\Meet 6\coding>
```

3. Code :

```java
    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 achievement!");
            } else if (value >= 60 && value <= 79) {
                System.out.println(x: "Grade C, increase your achievement!");
            } else if (value >= 50 && value <= 59) {
                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 if (value > 100) {
            System.out.println(x: "The value you entered is more than 100");
        } else {
            System.out.println(x: "The value you entered is invalid!");
        }

    }
}
```

Result :

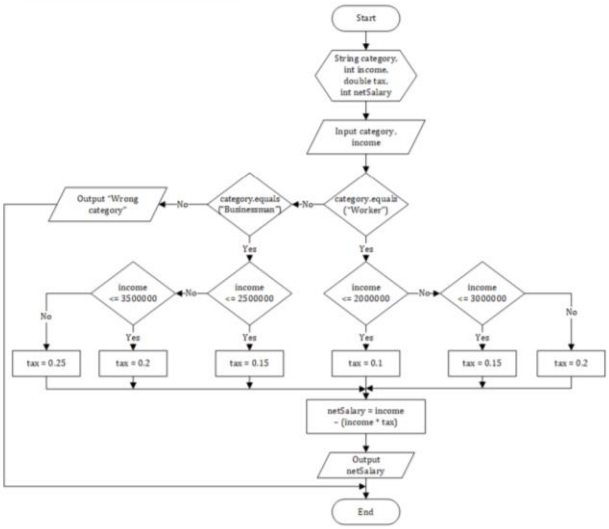
```
C:\Users\Al Azhar Rizqi\Documents\College\Basic Programming Practicum\Meet 6\coding>java Nested1.java
Enter a value (0 - 100) : 105
Grade E, you don't pass!

C:\Users\Al Azhar Rizqi\Documents\College\Basic Programming Practicum\Meet 6\coding>
```

Because if or only forms 2 shading numbers, it doesn't combine shading numbers.

Experiment 2

1. Observe the following flowchart!



The flowchart is used to calculate a person's net salary after taxes according to their category (worker and businessman) and the amount of income.

2. Open a text editor. Create a new file, name it Nested2.java

3. Write the basic structure of the Java programming language which contains the main() function

```
public class Nested2 {
    Run | Debug
    public static void main(String[] args) {

    }
}
```

4. Add the Scanner library.

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

5. Make a Scanner declaration with the name sc

```
Scanner sc = new Scanner(System.in);
```

6. Declare category, income, netSalary, and tax variables

```
System.out.print(s: "Enter a category : ");
category = sc.nextLine();
System.out.print(s: "Enter the amount of income : ");
income = sc.nextInt();
```

7. Write down the syntax for entering the value from keyboard

8. Create a nested selection structure. The first check is used to check the category (worker or businessman). Then a second check is carried out to determine the amount of tax based on the income that has been entered. Then add the program code to calculate the net salary received after taxes

```
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 receive : " + 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 receive : " + netSalary);
} else {
    System.out.println(x: "The category you entered is wrong");
}
```

9. Compile and run the program. Observe the results!

```
C:\Users\Al Azhar Rizqi\Documents\College\Basic Programming Practicum\Meet 6\coding>javac Nested2.java

C:\Users\Al Azhar Rizqi\Documents\College\Basic Programming Practicum\Meet 6\coding>java Nested2.java
Enter a category : worker
Enter the amount of income : 1500000
The net salary you will receive : 1350000

C:\Users\Al Azhar Rizqi\Documents\College\Basic Programming Practicum\Meet 6\coding>
```

Questions!

1. Run the program by entering category = worker and income = 2048485 using keyboard. Watch what happened! Why is the decimal number not displayed?

2. Describe the function of (int) in the following syntax! netSalary = (int) (income - (income * tax));

3. Run the program by entering category = BUSINESSMAN and income = 2000000. Watch what happens! What are the uses of equalsIgnoreCase?

4. Change equalsIgnoreCase to equals, then run the program by entering category = BUSINESSMAN and income = 2000000. Watch what happens! Why is the result like that? What are the uses of equals?

Answer!

1. Because the data type of the netSalary variable is integer not double or float whose function is to display decimal numbers.
2. The function from (int) to casting variable besides int to int.
3. Result :

```
C:\Users\Al Azhar Rizqi\Documents\College\Basic Programming Practicum\Meet 6\coding>java Nested2.java
Enter a category : BUSINESSMAN
Enter the amount of income : 2000000
The net salary you will receive : 1700000

C:\Users\Al Azhar Rizqi\Documents\College\Basic Programming Practicum\Meet 6\coding>
```

Uses of equalsIgnoreCase is method compares two strings, ignoring lower case and upper case differences.

4. Code :

```
if (category.equals(anObject: "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 receive : " + netSalary);
} else if (category.equals(anObject: "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 receive : " + netSalary);
} else {
    System.out.println(x: "The category you entered is wrong");
}
```

Result :

```
C:\Users\Al Azhar Rizqi\Documents\College\Basic Programming Practicum\Meet 6\coding>java Nested2.java
Enter a category : BUSINESSMAN
Enter the amount of income : 2000000
The category you entered is wrong

C:\Users\Al Azhar Rizqi\Documents\College\Basic Programming Practicum\Meet 6\coding>
```

Because without ignorecase, the method doesn't ignore the difference between lowercase and uppercase letters. The uses of equals only compare if it is equal to, nothing more.