Java Programming Tutorial

1. Exercises on Flow Controls

1.1 Exercises on Conditional (Decision)

Exercise 1.1: `CheckPassFail` (if-else): Write a program called **`CheckPassFail`** which prints "`PASS`" if the `int` variable "`mark`" is more than or equal to `50`; or prints "`FAIL`" otherwise.

Hints:

```
public class CheckPassFail { // saved as "CheckPassFail.java"
   public static void main(String[] args) {
      int mark = 49;         // set the value of mark here!
      System.out.println("The mark is " + mark);

      if ( ...... ) {
         System.out.println( ...... );
      } else {
         System.out.println( ...... );
      }
   }
}
```

Exercise 1.2: `CheckOddEven` (if-else): Write a program called **`CheckOddEven`** which prints "`Odd Number`" if the `int` variable "`number`" is odd, or "`Even Number`" otherwise.
Hints: `n` is an even number if `(n % 2)` is `0`.

```
public class CheckOddEven {   // saved as "CheckOddEven.java"
   public static void main(String[] args) {
      int number = 49;       // set the value of number here!
      System.out.println("The number is " + number);
      if ( ...... ) {
         System.out.println( ...... );
      } else {
         System.out.println( ...... );
      }
   }
}
```

Exercise 1.3a: `PrintNumberInWord` (nested-if, switch-case): Write a program called **`PrintNumberInWord`** which prints "ONE", "TWO",… , "NINE", "OTHER" if the `int` variable "`number`" is 1, 2,… , 9, or other, respectively. Use (a) a "nested-if" statement; (b) a "switch-case" statement.

Hints:

```
public class PrintNumberInWord {   // saved as "PrintNumberInWord.java"
   public static void main(String[] args) {
      int number = 5;

      // Using nested-if
      if (number == 1) {
         System.out.println("ONE");
      } else if (......) {
         ......
      } else if (......) {
         ......
         ......
      } else {
         ......
      }

      // Using switch-case
      switch(number) {
         case 1: System.out.println("ONE"); break;
         case 2: ......
         ......
         ......
         default: System.out.println("OTHER");
      }
   }
}
```

Exercise 1.3b: Similarly, write a program called **`PrintDayInWord`**, which prints “Sunday”, “Monday”, ... “Saturday” if the `int` variable "`day`" is 0, 1, ..., 6, respectively. Otherwise, it shall print “Not a valid day”.

1.2 Exercises on Loop (Iteration)

Exercise 1.2.1a `SumAndAverage` (Loop): Write a program called **`SumAndAverage`** to produce the sum of 1, 2, 3, ..., to an upperbound (e.g., 100). Also compute and display the average. The output shall look like:

```
The sum is 5050

The average is 50.5
```

Hints:

```
public class SumAndAverage {   // saved as "SumAndAverage.java"
   public static void main (String[] args) {
      int sum = 0;          // store the accumulated sum, init to 0
      double average;       // average in double
      int lowerbound = 1;   // the lower bound to sum i.e. the minimum number
      int upperbound = 100; // the upper bound to sum i.e. the maximum number

      for (int number = lowerbound; number <= upperbound; ++number) { // for loop
         sum += number;     // same as "sum = sum + number"
      }
      // Compute average in double. Beware that int/int produces int.
      ......
      // Print sum and average.
      ......
   }
}
```

Exercise 1.2.1b TRY:

Modify the program to use a "while-do" loop instead of "for" loop.

```
int number = lowerbound;
int sum = 0;
while (number <= upperbound) {
   sum += number; // same as "sum=sum + number"
   ++number; // same as "number=number+1"
}
```

Exercise 1.2.1c

1. Modify the program to use a "do-while" loop.

```
int number = lowerbound;
int sum = 0;
do {
   sum += number; //same as "sum=sum + number"
   ++number; // same as "number=number+1"
```

```
} while (number <= upperbound);
```

2. What is the difference between "for" and "while-do" loops? What is the difference between "while-do" and "do-while" loops?

1.3 Exercises on Nested-Loop

Exercise 1.3.1: `SquareBoard` (nested-loop): Write a program called **`SquareBoard`** that displays the following *n*×*n* (*n*=5) pattern using two nested for-loops.

```
# # # # #
# # # # #
# # # # #
# # # # #
# # # # #
```

Your program should use only two output statements, one EACH of the followings:

```
System.out.print("# ");    // print # and a space, without newline
System.out.println();      // print a newline
```

Hints:

```
public class SquareBoard {   // saved as "SquareBoard.java"
```

```
   public static void main (String[] args) {
      int size = 5;    // size of the board
      for (int row = 1; ......; ......) {
         for (int col = 1; ......; ......) {
            ......
         }
         ......
      }
   }
}
```