In all future exercises you should consider if methods should be used to break up programs into logical sections.

1. Complete the following program that inputs and validates a mark (0-100). You need to include a method called getMark.

```
import java.util.Scanner;

public class Student {

   public static void main(String[] args)
   {
        int mark = getMark();

        System.out.print("The Mark is: " + mark);
   }

   // Add the getMark method here

}
```

2. Extend the above program to display the grade for the mark entered. The program should include a method that takes an integer parameter and return the grade (char). The rules for determining the grades are:

   A is 70 or more, B is 60-69, C is 50-59, D is 40-49 and F is 0-39.

3. Complete the following program that inputs the width and height of a rectangle and outputs the area. You need to include a method called calcArea that takes two integer parameters (width and height) and returns the area.

```
// Program to calculate area of a rectangle
import java.util.Scanner;

public class Area {

   public static void main(String[] args)
   {
        Scanner in = new Scanner(System.in);

        System.out.print("Please enter width: ");
        int width = in.nextInt();

        System.out.print("Please enter height: ");
        int height = in.nextInt();

        System.out.println("\nThe area is: " +
                calcArea(width, height) );
   }

   // Add the calcArea method here

}
```

Compile and test the program.

4. Modify the solution to question (3) by reusing the *readInteger()* method developed as part of the this week's videos.

5. Develop a program that asks the user to input a number between 1 and 12 (inclusive) and displays the month name. The program should include a method that given the month number prints out the name of the appropriate month.

   Test your program with valid and invalid values (eg: 13).

6. Write a program that asks the user to enter 3 numbers (1-100 inclusive) and outputs the smallest number. The program should include methods for:

   - Inputting and validating a number: Gets the user input and checks that it is between 1 and 100 (inclusive). Once a valid number has been entered the value is returned from the method.
   - Comparing the three numbers and find the smallest number. This method will take three parameters and return the smallest number.

7. **Employee (Case Study) – Validation Methods**

   Move the input and validation for employee name and staff number into methods. For each input create two methods, for example, employee name would have:

   - IsValidName: takes a single parameter (the employee name) and returns **true** for valid names.
   - GetName: prompts the user to enter a name and then validates it. If the name is not valid, an appropriate error message is displayed and the user is asked to input the name again. Once a valid name is entered it is returned from the method.

8. Binary to Decimal Conversion

Write and test a program that get a binary number as a string from the user and converts it to a decimal number. The program should include two methods:

- getBinary: reads binary number from the user and validates it.
- bin2Dec: converts a binary number (string) into a decimal value.

9. Week 2's Practical you were asked to:

   Write a program that asks the user to input a month number (eg: 2) and year (eg: 2000). The program then displays the number of days in the month (taking into account leap years).

   Leap years occur if the year is divisible by 4, except for the century years.
   Only the centuries divisible by 400 are leap years. e.g. 1900 was not a leap year; 2000 was a leap year.

Update the program to incorporate two methods: isLeapYear and daysInMonth.

10. Write a menu-driven program that provides three options:

    - Option 1 allows the user to enter a temperature in Celsius (C) and displays the corresponding Fahrenheit (F) temperature.
    - Option 2 allows the user to enter a temperature in Fahrenheit (F) and displays the corresponding Celsius (C) temperature.
    - Option 3 allows the user to quit.

    The formulae are: $F = 9 / 5 \times C + 32$ $C = 5 \times (F - 32) / 9$

    Adapt your program so that the user is not allowed to enter a temperature below absolute zero (this is -273.15 C or -459.67 F)

11. If a customer places order for more than 10 items he/she receives a 1% discount on the total (quantity x item price). Write and test a program to calculate the total price (including any discount).

    Update your program to include the following rule: Trade Customer receive an additional 3%.

12. Decimal to Binary Conversion

Write and test a program that get a decimal number as an integer from the user and display the binary equivalent. The program should include a method called dec2Bin that converts an integer to a binary number (string).

13. Hexadecimal Conversion

Write two programs similar to exercise 3 and 4 to convert a hexadecimal number to binary and vice-versa.

For students with Programming Experience:

14. Write a program that ask the user for two inputs (number and width) and outputs the "number" with required number of leading spaces to right align into the specified width.

    eg: 10 and 5 would output 2 leading spaces: `  10`

    eg: 123 and 4 would output a leading space: ` 123`