

Homework 1 : Simple Game

This assignment is designed to give you practice writing code and applying lessons and topics for the current module.

This homework deals with the following topics:

- The Java Language
- Classes

The Assignment

In this assignment, you will implement a simple game in a class called SimpleGame. This game has 2 options for the user playing. Based on user input, the user can choose to either **convert time**, from seconds to hours, minutes, and seconds, or **calculate the sum of all digits in an integer**.

At the beginning of the game, the user will be prompted to input either **1** or **2**, to indicate which option of the game they want to play. **1** will indicate converting time, and **2** will indicate calculating the sum of digits in an integer.

For converting time, the user will be prompted to input a number of seconds (as an int) and the program will call a method that will convert the seconds to time, in the format hours:minutes:seconds, and print the result. For example, if the user enters **6734**, the program will print the time, **1:52:14**. As another example, if the user enters **10,000**, the program should print **2:46:39**.

For calculating the sum of digits in an integer, the user will be prompted to input a number (as an int) and the program will call a method to calculate the sum of the digits in that number, and print the result. For example, if the user enters **321**, the program will print the sum, **6**, because the individual digits in the number add up to 6. $3 + 2 + 1 = 6$.

There are 2 methods that need to be implemented in the SimpleGame class:

- `convertTime(int seconds)` - Method that converts a given number of seconds to time in the format hours:minutes:seconds.
- `digitsSum(int input)` - Method that adds all the digits in a given non-negative integer.

Each method has been defined for you, but without the code. See the javadoc for each method for instructions on what the method is supposed to do and how to write the code. It should be clear enough. In some cases, we have provided hints and example method calls to help you get started.

For example, we have defined a “convertTime” method for you (see below) which converts a given seconds value to time. For now, the method returns a null value as a placeholder. Read the javadoc, which explains what the method is supposed to do. Then write your code where it says “// TODO” to implement the method and replace the placeholder. You’ll do this for each method in the program.

```
/**
 * Write a method to convert the given seconds to
hours:minutes:seconds.
 * @param seconds to convert
 * @return string for the converted seconds in the format: 23:59:59
 *
 * Example(s):
 * - If input seconds is 1432, print and return output in the format:
0:23:52
 */
public String convertTime(int seconds){
    // TODO: Your code goes here
    return null;
}
```

We’ll run Java tests against your program to test whether each method implementation is correct and whether you have considered enough cases. Examples of cases have been provided in the javadocs.

The main method **has not been completely implemented for you** (see example below). You’ll need to write code (where it says “// TODO”) to use it to run and interact with your program, and to see if your methods are working as expected. Spend some time on testing.

```
public static void main(String[] args) {
    // Create an instance of the SimpleGame class.
    // TODO: Your code goes here

    Scanner sc = new Scanner(System.in);

    // Ask the user which game to play.
```

```
// Then ask the user for input and pass the value to the
corresponding method.

// If the user enters 1, ask for an integer to convert and call
the convertTime method.
// If the user enters 2, ask for an integer and call the digitsSum
method.

// TODO: Your code goes here

sc.close();
}
```

Tips for this Assignment

In this assignment, some tips are given as follows:

- Modulus operation in Java:
“%” is the modulus operator in Java. It returns the remainder, after division.
For example:

```
int result1 = 4 % 2; //result1 is 0
int result2 = 1000 % 90; //result2 is 10
```
- int to String conversion in Java:
There are multiple ways to convert an int to a String.
 - Use “Integer.toString(int)”
For example:

```
int num = 9;
String str = Integer.toString(num); //str is “9”
```
 - Use “String.valueOf(int)”
For example:

```
int num = 9;
String str = String.valueOf(num); //str is “9”
```
- String to int conversion in Java:
 - Use “Integer.parseInt(str)”
For example:

```
String str="0";
int num = Integer.parseInt(str); //num is 0
```
- Getting a specific char in a String by index in Java:
 - Use “charAt(int)”
For example:

```
String str = "cit";
char firstChar = str.charAt(0); //firstChar is 'c'
```

Submission

To complete the assignment, write the program as described in *SimpleGame.java* and implement the methods in it. Do not modify the name of the methods in *SimpleGame.java* or the automated testing will not recognize it. Submit the completed program using the steps outlined in the assignment in Coursera.

Evaluation

Points: Each method is worth 10 points.