# Exercise: Streams in Java

This document defines in-class exercise problems from the [“Java Fundamentals“ Course @ Software University](https://softuni.bg/trainings/1232/java-fundamentals-october-2015). You are presented with some problems and certain steps you need to take in order to accomplish the tasks.

## Problem 1. Create, Read and Write in Files

Create a program that reads a log about users that spent time playing some game. The format is **{user hh:mm hh:mm ... hh:mm}.** Calculate the total time spent for each user and write it to another file **total-played.txt**

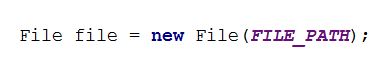
|  |
| --- |
| **Output (total-played.txt)** |
| gosho 1293 (0 days, 21 hours, 33 minutes)  pesho 468 (0 days, 7 hours, 48 minutes)  misho 797 (0 days, 13 hours, 17 minutes) |

### Step 1. Read and Process the Input

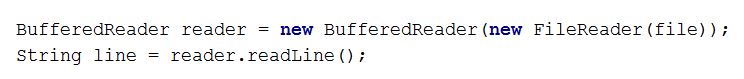
You can find the file users.txt in the exercise archive. Save it somewhere on your computer. It’s a good practice to use constants in your program for things that never change, so you can save the filepath in your java program as a final string. (Final is like const in C#)

C:\Users\Edu\Pictures\path.JPG

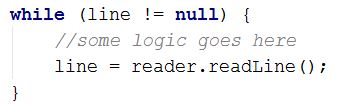
Your path may be different, depending on where you saved the file. Now you need to pass that path to the Java File class



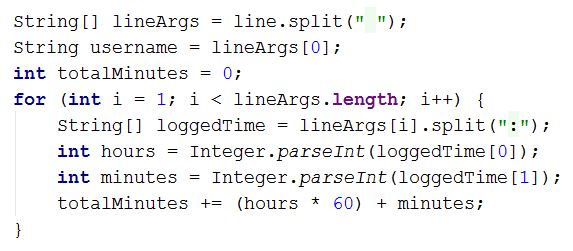
Reading the file info in java is easy. The **BufferedReader** class provides a handy method .**readLine**() that acts just like the Scanner’s .**nextLine**(). We are ready to read the first line. It’s a String, so save it in a variable of that type



This will read the first line, but there may be many more. Since we don’t know the exact amount of lines, we can use a while loop



### Step 2. Calculate the time spent

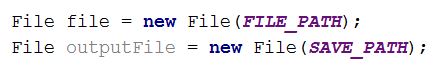
Each of the lines can be split be space. The first element of the resulting array is the username, the next elements will be the individual sessions. Each session can be split by “:”. The left part is the hours; the right part is the minutes. You can keep the total time as minutes and then do some calculations.

### Step 3. Write the output to another file

You need to create another file where the total time spent will be stored. You can use another constant

C:\Users\Edu\Pictures\file_create.JPG

Initialize the output file class:

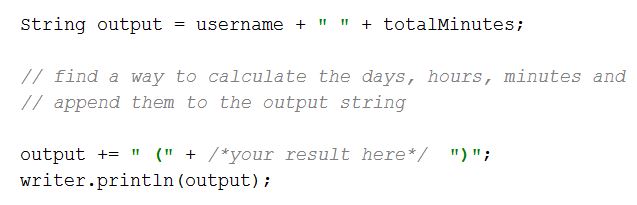


The **FileWriter** class will handle the output stream and will create the output file if it does not exist. The **PrintWriter** is a helper class with user-friendly methods. We will use the **.println()**

C:\Users\Edu\Pictures\writer.JPG

### Step 4. Print the Result

In the while loop, after reading from the users file and calculating the time, print the next line to the output file.



### Step 5. Close all resources

Finally, close all open streams

