

CSCD 437
Lab 7
Defend Your Code

SPECIFICATIONS

Write a program Java that does the following. Be sure and make clear what is expected of the user. Do this with excellent prompting.

- You will not use try/catch except for the cryptographically secure methods for hashing the password.
- I have provided a main with menu choices. You will need to add the appropriate Java calls to your methods. You should not need to add to my main. It all should happen in your classes.
- There are 9 tasks below to complete. Each teammate must do 3 separate items from their other teammates. (For the team of two one does 5 methods and the other does 4 methods). Each will create their own Java file name Team Number Member Number.java (Member number is determined based on alphabetical order by last name)

Example: Team16Member1.java

- Your Java file will contain your name and a description of any problems/something not working
- You will need to add the appropriate calls in main under the appropriate menu choice.
- Your code will be in a package named lab7

You need to:

- 1) Prompt for and read the user's first name and then last name
 - Each should be at most 50 characters
 - You must use regex to verify the name is at least two letters. A valid name contains letters, spaces, possibly a dash (-), and possibly numbers as roman numerals. The name can't contain the word hitler.

Example: Jane Petra-Soveig Foster II is a valid name

- 2) Prompt for and read in a username.
 - Username will not exceed 25 characters
 - You must use regex to verify the username starts with a letter and only contains other letters, numbers and the underscore.
 - You will write the username to db.txt (NOTE: additional items will also be written to db.txt)

- 3) Prompt for and read the user's password
 - You must use regex to verify the password contains at least one uppercase letter, one lower case letter, two numbers that are not repeated or consecutive, special character but not *, @, \$ or !.
 - Ask the user to re-enter the password
 - Verify the passwords match before continuing
 - Store the cryptographically secured read in password
 - Store to a file name db.txt, the username (from #2) a colon, the salt, a colon, and the hashed password.

- 4) Prompt for and read in the user's email.
 - Use regex to verify the email contains @ and more than 1 character after the at symbol.
 - A dot or dash is optional in a string before the @ symbol, but it is not allowed as the character right before the at symbol. A dot or dash can't be used more than once consecutively.
- 5) Prompt for and read two int values from the user
 - You must use regex to verify the number entered is within -2 billion and change to +2billion and change.
 - The number can start with a negative symbol (dash)
 - The number can start with a plus symbol
 - The number can start without a plus symbol
- 6) Prompt for and read the name of an input file from the user
 - You must use regex to verify the filename contains only letters and/or numbers and ends in .txt
 - You must confirm the file exists
 - You must use regex to ensure the file is located in the same directory as your Java code. You won't let the user abuse the filesystem with some file buried outside the folder from your Java code
- 7) Prompt for and read the name of an output file from the user
 - You must use regex to verify the filename contains only letters and/or numbers and ends in .txt
 - You must use regex to ensure the file is located in the same directory as your Java code. You won't let the user abuse the filesystem with some location buried outside the folder from your Java code
- 8) Open the output file:
 - Write the user's name – last name, first name
 - Write the email
 - The result of adding the two integer values
 - You must verify that the numbers, when added will not overflow
 - The result of multiplying the two integer values
 - You must verify that the numbers, when added will not overflow
 - The contents of the input file
- 9) Ask the user to "relogin" by entering their username and password.
 - Same regex rules as before apply to the username (#2 and #3)
 - Verify the username and password are correct

Any errors, invalid input, etc will be reported to an error log file named "errors.txt".

TO TURN IN

Turn in a single zip file that contains:

- CSCD437Lab7.java and all the other Java files in the package
- Your input, output, db.txt and errors.txt files
- Output captures from systematically testing your solutions – Clearly denoted with what is being tested
- A README.txt that includes:
 - team member names
 - shortcomings
 - any compilation instructions your code requires.

Name your zip your team number lab7.zip (Example team16lab7.zip)

I should be able to unzip your solution, compile your code and execute your program.

FINAL NOTE: I will not attack your input file, output file, db.txt or errors.txt