Edward Kuisseu Tatchim

Computer Science 1-INT 2210

Lab Assignment #5

Dr. Homer Sharafi

November 19, 2018

**Solution to Lab Assignment #5**

I named this program ***userPassword****.* This name rightly conveys the purpose of the program. This program is written in such a way to receive a user input for a password and it performs validation of this password based on pre-set conditions.

For my program to perform everything as required by the lab assignment, I declared a string variable *password* that would store the user’s input to be validated as a good password. I then created a class called *isValid* that would perform several operations on the input data to validate it. Firstly, I declared Boolean variables to hold the truth values for each stated variable. I then printed a message that would tell the user the value of the password that he typed in. Furthermore, I used a for loop to convert the value of password string to a character array. For every letter in the character array, I performed a check for the length, upper case, lower case, and digit. If the password contains all these requirements, then it is valid otherwise it isn’t. I also declared 3 counter variables to keep track of the number of times that each previously mentioned requirement was met while the array was searched. If any requirement is not met at least once, then the password cannot be valid. The counter variables must be greater than 0 for the password to be valid.

**Input Data:** Password string from the user

**Processing Data:** The password string that is input by the user goes through loops and logical lines of code as processing data. This data is processed and passed through multiple test conditions in order to arrive at the desired conclusion. Every string that is tested is considered as processing data.

**Output Data:** This program outputs a message to the user to determine if his/her string input is valid or not. This is our output information.

***userPassword (Written in Java)***

**package Lab5;**

**/\***

**\*This program was written by Edward Tatchim on November 11, 2018.**

**\* The purpose of this program is to verify that user passwords meet the following requirements:**

**\* should be 6 characters long**

**\* should contain at least one uppercase letter**

**\* should contain at least one lowercase letter**

**\* should have at least one digit**

**\*/**

**import java.util.Scanner; //importing the scanner class**

**import java.util.regex.Pattern; //importing the pattern verifier class**

**import javax.swing.JOptionPane; //importing a class for the dialog box pane**

**public class UserPassword { //user password class created**

**public static String password=""; //string variable to contain the password input by user**

**public static void isValid(String password) //method to determine if the the user input string meets the set requirements**

**{**

**boolean hasAppropriateLength= password.length()>=6; //boolean variable to hold required password size. The value is stored as boolean (TRUE or FALSE)**

**boolean hasDigit=false; //boolean variable to store digit check as false at first**

**boolean hasUpperCase= false; //boolean variable to store upper case check as false at first**

**boolean hasLowerCase= false; //boolean variable to store lower case check as false at first**

**int countUpperCase=0; // integer variable to store the number of times upper case letters appear in the password**

**int countLowerCase=0; //integer variable to store the number of times lower case letters appear in the password**

**int countDigit=0; //integer variable to store the number of times digits appear in the password**

**System.out.println(" Your password is: "+password); //after collecting the user input, this line will print the content of this input to the screen for user to see**

**if (hasAppropriateLength) //check that password has minimum length**

**for(char userInput:password.toCharArray()) //for loop that transforms the content of password to a character array and names this array userInput**

**{**

**boolean failedUpperCaseCheck=false;**

**boolean failedLowerCaseCheck=false;**

**boolean failedDigitCheck=false;**

**boolean failedLengthCheck=false;**

**if(userInput>='A'&&userInput<='Z') //conditional check for Upper case letter**

**countUpperCase++;**

**else{ failedUpperCaseCheck=true;}// incrementing upper case appearance counter**

**if (userInput>='a'&&userInput<='z') //conditional check for Lower case letter**

**countLowerCase++; //incrementing lower case appearance counter**

**else {failedLowerCaseCheck=true;}**

**if(userInput>='0'&&userInput<='9') //conditional check for digit appearance in the password**

**countDigit++; //incrementing the digit appearance counter**

**else {failedDigitCheck=true;}**

**}**

**{**

**if(countUpperCase==0||countLowerCase==0||countDigit==0) //if any of the checks equal 0, then the password is invalid**

**{**

**System.out.println("Your password is invalid");**

**}**

**else {System.out.println("Congrats! Your password is valid");} //otherwise if ALL the checks are greater than 0, then the password is valid**

**}**

**}**

**public static void main(String[] args) { //main method to create userPassword objects and test for their validity**

**Scanner keyboard=new Scanner(System.in); //keyboard object created to store the user input**

**System.out.println("Dear user, please enter your password to be verified"); //output string to prompt the user to enter his/her password**

**String password=keyboard.next(); //sends the input stored in the keyboard variable into the password variable**

**isValid(password); //makes a call to the isValid method with password passed as parameter**

**}**

**}**