Week 07 : Programming Assignment 5

Due on 2025-03-13, 23:59 IST

Implement a Simple Password Validator

In this task, you need to implement a **password validation system** using Java. The goal is to check if a given password meets the following conditions:

1. **Minimum Length Requirement**: The password must be at least **8 characters long**.
2. **Uppercase Letter Requirement**: The password must contain **at least one uppercase letter (A-Z)**.
3. **Number Requirement**: The password must contain **at least one numeric digit (0-9)**.

If the password meets **all three conditions**, print `"Valid Password"`. Otherwise, print `"Invalid Password"`.

Input Format:

- A **single string** representing the password (can contain alphabets, numbers, and special characters).

Output Format:

- Print "`Valid Password`" if the password satisfies all the conditions.
- Otherwise, print "`Invalid Password`".

Example Input:
Password123

Example Output:
Valid Password

Private Test cases used for evaluation	Input	Expected Output	Actual Output	Status
Test Case 1	password123	Invalid Password	Invalid Password	Passed

The due date for submitting this assignment has passed.
1 out of 1 tests passed.
You scored 100.0/100.

Assignment submitted on 2025-03-06, 13:31 IST
Your last recorded submission was :

```java
import java.util.Scanner;

public class W07_5 {
    private String password;

    // Step 1: Constructor to initialize the password variable
    public W07_5(String password) {
        this.password = password; // Assign the passed password to the instance variable
    }
    // ================================
    // Note: Try solving it without hints first-only check if you're truly stuck.
    // Avoid AI or internet searches; quick answers won't build real skills.
    // Struggle a bit, learn for life! Be honest with yourself!
    //
public boolean isValidPassword(String password) {
        if (password.length() < 8) {
            return false;
        }

        boolean hasUppercase = false;
        boolean hasDigit = false;

        for (int i = 0; i < password.length(); i++) {
            char c = password.charAt(i);
            if (Character.isUpperCase(c)) {
                hasUppercase = true;
            } else if (Character.isDigit(c)) {
                hasDigit = true;
            }
        }

        return hasUppercase && hasDigit;
    }
public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        // Read password input from user
        String inputPassword = scanner.nextLine();
        scanner.close();
        W07_5 validator = new W07_5(inputPassword);

        // Check password validity and print result
        if (validator.isValidPassword(inputPassword)) {
            System.out.print("Valid Password");
        } else {
            System.out.print("Invalid Password");
        }

        scanner.close();
    }
}
```

Sample solutions (Provided by instructor)

```java
import java.util.Scanner;

public class W07_5 {
    private String password;

    // Step 1: Constructor to initialize the password variable
    public W07_5(String password) {
        this.password = password; // Assign the passed password to the instance variable
    }
    // ================================
    // Note: Try solving it without hints first-only check if you're truly stuck.
    // Avoid AI or internet searches; quick answers won't build real skills.
    // Struggle a bit, learn for life! Be honest with yourself!
    //
public boolean isValidPassword(String password) {
        // Step 1: Check if the password length is at least 8 characters
        if (this.password.length() < 8) {
            return false; // If password is too short, it's invalid
        }

        boolean hasUppercase = false; // Flag to track if there is an uppercase letter
        boolean hasDigit = false; // Flag to track if there is a number

        // Step 2: Loop through each character in the password
        for (char ch : this.password.toCharArray()) {
            if (Character.isUpperCase(ch)) {
                hasUppercase = true; // Found an uppercase letter
            }
            if (Character.isDigit(ch)) {
                hasDigit = true; // Found a number
            }

            // If both conditions are met, no need to check further
            if (hasUppercase && hasDigit) {
                return true;
            }
        }

        // Step 3: If either condition is not met, return false
        return false;
    }
public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        // Read password input from user
        String inputPassword = scanner.nextLine();
        scanner.close();
        W07_5 validator = new W07_5(inputPassword);

        // Check password validity and print result
        if (validator.isValidPassword(inputPassword)) {
            System.out.print("Valid Password");
        } else {
            System.out.print("Invalid Password");
        }

        scanner.close();
    }
}
```