using System;

using System.Text.RegularExpressions;

using System.Linq;

using System.Collections.Generic;

class PasswordChecker

{

static List<string> GetPasswordErrors(string password, string regNumber, string name)

{

List<string> errors = new List<string>();

string specialChars = "!@#$%^&\*()"; // Allowed special characters

// Condition 1: Must contain at least two characters from the registration number

int regCharCount = regNumber.Count(c => password.Contains(c));

if (regCharCount < 2)

errors.Add("Password must contain at least two characters from your registration number.");

// Condition 2: At least one uppercase letter

if (!Regex.IsMatch(password, "[A-Z]"))

errors.Add("Password must contain at least one uppercase letter.");

// Condition 3: At least two special characters in order

MatchCollection specialMatches = Regex.Matches(password, "[" + Regex.Escape(specialChars) + "]");

if (specialMatches.Count < 2)

{

errors.Add("Password must contain at least two special characters in order.");

}

else

{

// Ensure special characters appear in order

string foundSpecials = string.Concat(specialMatches.Cast<Match>().Select(m => m.Value));

if (!Regex.IsMatch(foundSpecials, "^[!@#$%^&()]$"))

errors.Add("Special characters in the password must appear in order (!@#$%^&\*()).");

}

// Condition 4: At least four lowercase letters from user's name

// Check lowercase letters from name that are present in password

int lowercaseCount = password.Count(c => name.Contains(c) && char.IsLower(c));

if (lowercaseCount < 4)

errors.Add("Password must contain at least four lowercase letters from your name.");

// Condition 5: Max length 12

if (password.Length > 12)

errors.Add("Password must not exceed 12 characters.");

return errors;

}

static void Main()

{

// User input for the details

Console.Write("Enter your name: ");

string userName = Console.ReadLine().ToLower(); // Convert to lowercase for comparison

Console.Write("Enter your registration number: ");

string regNumber = Console.ReadLine();

Console.Write("Enter your password: ");

string password = Console.ReadLine();

// Get validation errors for the password

List<string> errors = GetPasswordErrors(password, regNumber, userName);

// Output results

if (errors.Count == 0)

{

Console.WriteLine("✅ Valid password!");

}

else

{

Console.WriteLine("❌ Invalid password. Please check the following issues:");

foreach (string error in errors)

{

Console.WriteLine("- " + error);

}

}

Console.ReadKey();

}

