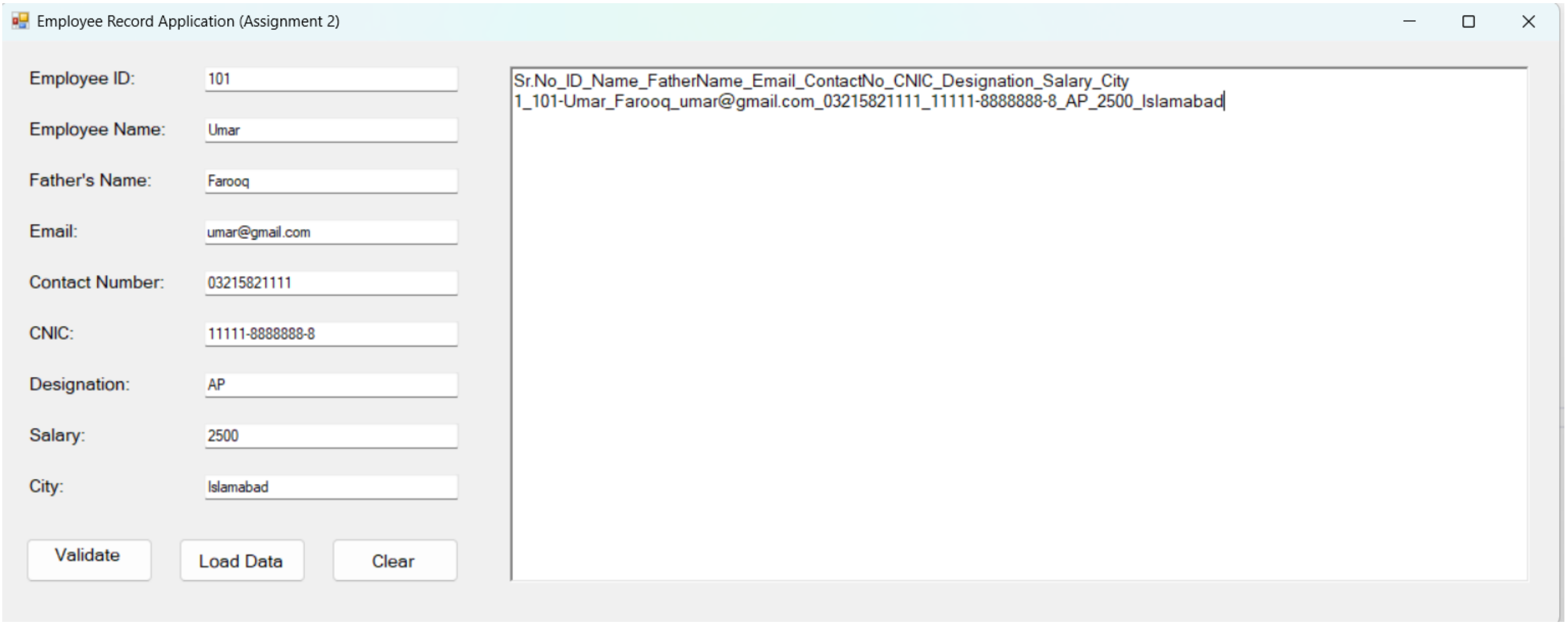
Advance Programming

Assignment 3

Submission Date: 16 Dec 2025

Submission Form: Soft Form (Word File)

Note: Any form of copying and use of chatGPT code is strictly not allowed.



Employee ID:	101
Employee Name:	Umar
Father's Name:	Farooq
Email:	umar@gmail.com
Contact Number:	03215821111
CNIC:	11111-8888888-8
Designation:	AP
Salary:	2500
City:	Islamabad

```
Sr.No_ID_Name_FatherName_Email_ContactNo_CNIC_Designation_Salary_City
1_101-Umar_Farooq_umar@gmail.com_03215821111_11111-8888888-8_AP_2500_Islamabad
```

Employee Record Application

Overview

In this assignment, you will create a simple desktop application that collects employee information through a form, validates each field using **Regular Expressions**, and then **saves and displays** the stored employee records using **file handling**.

Requirements

1. Form Fields (All Validated Using Regular Expressions)

Create a form with the following **TextBoxes**.
Each TextBox **must be validated using Regular Expressions** before saving data:

- **Employee ID** (must be a valid formatted ID)
- **Employee Name** (proper alphabetical name)
- **Father's Name**
- **Email Address**
- **Contact Number**
- **CNIC**
- **Designation**
- **Salary**
- **City**

If any field is invalid, the application must:

- Display a **MessageBox** showing the specific error
- Highlight the incorrect field (optional but preferred)

---

2. RichTextBox to Display Saved Data

Add a **RichTextBox** on the right side of the form.

This RichTextBox will:

- Show **all saved employee records** (one per line or multi-line format)

- Display the result of reading data back from a file

---

3. Buttons and Their Tasks

a. “Validate & Save” Button

When clicked:

1. Validate **each textbox** using **Regular Expressions**
2. If all fields are valid:
    - Generate a formatted employee record (single string or multi-line block)
    - Save the data to a file (e.g., `employees.txt`)
    - Append the newly saved record to the RichTextBox
3. If any validation fails:
    - Stop the process
    - Show message telling which field is invalid

---

b. “Load Data” Button

When clicked:

- Read the entire file (e.g., `employees.txt`)
- Display ALL stored employee records in the **RichTextBox**
- If file does not exist, show a message

---

c. “Clear” Button

When clicked:

- Clear all textboxes
- Clear RichTextBox

---

4. File Handling Requirements

You must use:

- **StreamWriter** (for saving data)
- **StreamReader** (for loading data)

Data must be **appended** to the file (not overwritten).
Use proper formatting to store the record clearly.

---

5. Validation Requirements

Every input field must be validated using **Regular Expressions**, covering:

- Only letters allowed
- Only digits allowed
- Pattern formats (email, CNIC, contact number, ID format)
- Minimum/maximum length checks
- No empty fields