

Week 7 C# In-Class Practice (Solution)

Task

```
// TODO: Implement using parameterized queries
// 1. Validate name is not null/empty
// 2. Create parameterized SQL query to add Name, Email, Age into Customers table
// 3. Use try-catch + using statements for connection and command
// 4. Use Add(parameter) approach
// 5. Handle NULL for Age properly with DBNull.Value
// 6. Return true if successful, false otherwise
```

Solution

```
using System;
using Microsoft.Data.Sqlite;

public class CustomerService
{
    private string _connectionString;

    public CustomerService(string connectionString)
    {
        _connectionString = connectionString ?? throw new ArgumentNullException(nameof(connectionString));
    }

    public bool RegisterCustomer(string name, string email, int? age)
    {
        // 1. Validate name is not null/empty
        if (string.IsNullOrWhiteSpace(name))
            throw new ArgumentException("Name cannot be empty or whitespace", nameof(name));

        // 2. Create SQL query to add
        // @ allows multiple lines
        string sql = @"
            INSERT INTO Customers (Name, Email, Age)
            VALUES (@Name, @Email, @Age)";

        try
        {
            // 3. Use using statements for connection and command
            using (SqliteConnection connection = new SqliteConnection(_connectionString))
            using (SqliteCommand command = new SqliteCommand(sql, connection))
            {
                connection.Open();

                // 4. Use Add(parameter) approach
                command.Parameters.Add(new SqliteParameter("@Name", DbType.String) {
                    Value = name.Trim()
                });

                command.Parameters.Add(new SqliteParameter("@Email", DbType.String) {
                    Value = string.IsNullOrWhiteSpace(email) ? (object)DBNull.Value : email.Trim()
                });

                // 5. Handle NULL age properly with DBNull.Value
                command.Parameters.Add(new SqliteParameter("@Age", DbType.Int32) {
                    Value = age ?? (object)DBNull.Value
                });

                int rowsInserted = command.ExecuteNonQuery();

                // 6. Return true if successful, false otherwise
                return rowsInserted > 0;
            }
        }
        catch (Exception ex)
        {
            Console.WriteLine($"Unexpected error: {ex.Message}");
            return false;
        }
    }
}
```