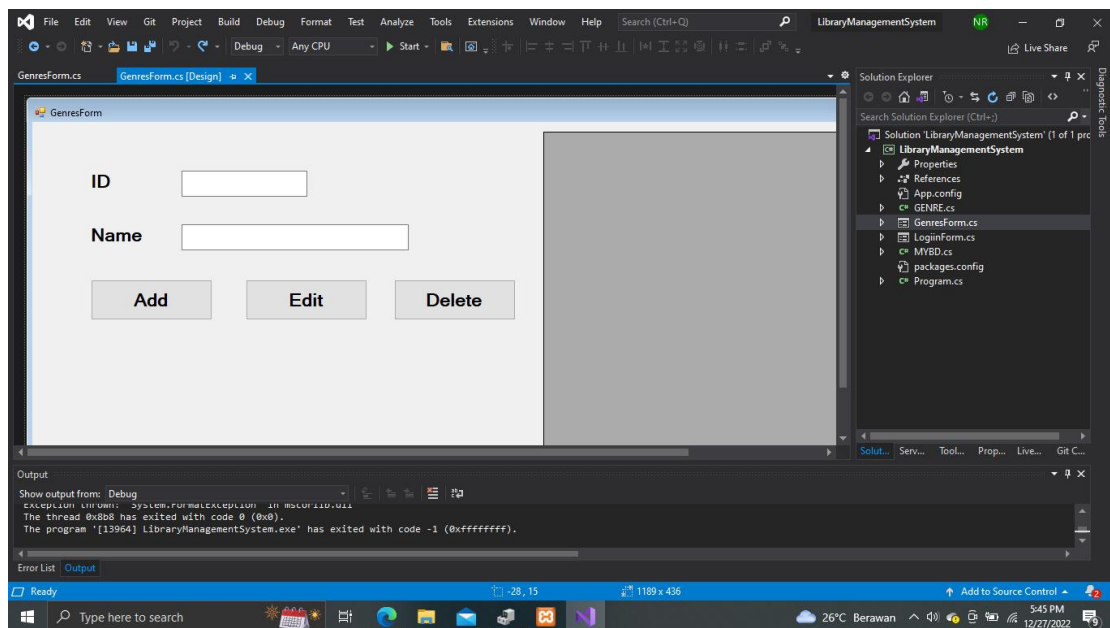
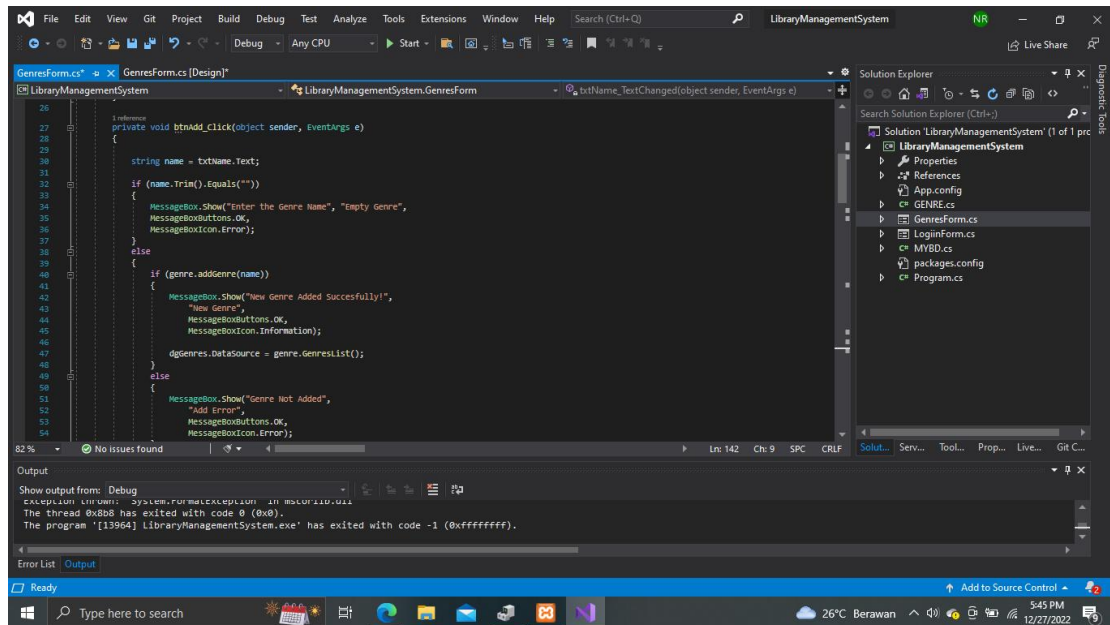


Nama : Najib Sauqi Rubbayani

Nim : 21.11.4406

Kelas : 21S1IF09

GenresForm



```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace LibraryManagementSystem
{
    public partial class GenresForm : Form
    {
        public GenresForm()
        {
            InitializeComponent();

            GENRE genre = new GENRE();

            private void label1_Click(object sender, EventArgs e)
            {
            }

            private void btnAdd_Click(object sender, EventArgs e)
            {
                string name = txtName.Text;

                if (name.Trim().Equals(""))
                {
                    MessageBox.Show("Enter the Genre Name", "Empty Genre",
                        MessageBoxButtons.OK,
                        MessageBoxIcon.Error);
                }
                else
                {
                    if (genre.addGenre(name))
                    {
                        MessageBox.Show("New Genre Added Successfully!",
                            "New Genre",
                            MessageBoxButtons.OK,
                            MessageBoxIcon.Information);

                        dgGenres.DataSource = genre.GenresList();
                    }
                    else
                    {
                        MessageBox.Show("Genre Not Added",
                            "Add Error",
                            MessageBoxButtons.OK,
                            MessageBoxIcon.Error);
                    }
                }
            }

            private void label2_Click(object sender, EventArgs e)
            {
            }

            private void btnEdit_Click(object sender, EventArgs e)
            {
                try
                {
                    int id = Convert.ToInt32(txtId.Text);
                    string name = txtName.Text;

                    if (name.Trim().Equals(""))
                    {
                        MessageBox.Show("Enter the Genre Name",
                            "Empty Genre",
                            MessageBoxButtons.OK,
                            MessageBoxIcon.Error);
                    }
                    else
                    {
                        if (genre.editGenre(id, name))
                        {
                            MessageBox.Show("Genre Name Updated Successfully",
                                "Edit Genre",
                                MessageBoxButtons.OK,
                                MessageBoxIcon.Information);

                            dgGenres.DataSource = genre.GenresList();
                        }
                        else
                        {
                            MessageBox.Show("Genre Not Updated",
                                "Edit Error",
                                MessageBoxButtons.OK,
                                MessageBoxIcon.Error);
                        }
                    }
                }
                catch (Exception ex)
                {
                    MessageBox.Show(ex.Message, "Invalid ID");
                }
            }
        }
    }
}

```

```

private void btnDelete_Click(object sender, EventArgs e)
{
    try
    {
        int id = Convert.ToInt32(txtId.Text);
        if (genre.removeGenre(id))
        {
            MessageBox.Show("Genre Deleted Successfully!",
                "Delete Genre",
                MessageBoxButtons.OK,
                MessageBoxIcon.Information);

            dgGenres.DataSource = genre.GenresList();
        }
        else
        {
            MessageBox.Show("Genre Not Deleted",
                "Delete Error",
                MessageBoxButtons.OK,
                MessageBoxIcon.Error);
        }
    }
    catch (Exception ex)
    {
        MessageBox.Show(ex.Message, "Invalid ID");
    }
}

private void dgGenres_CellClick(object sender, DataGridViewCellEventArgs e)
{
    txtId.Text = dgGenres.CurrentRow.Cells[0].Value.ToString();
    txtName.Text = dgGenres.CurrentRow.Cells[1].Value.ToString();
}

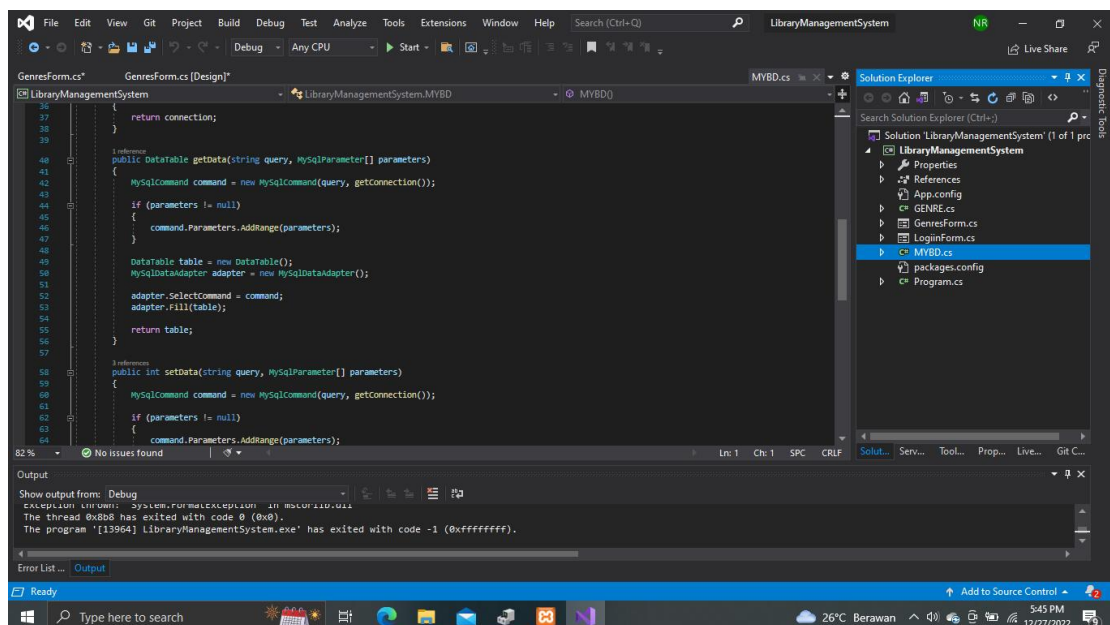
private void txtId_TextChanged(object sender, EventArgs e)
{
}

private void txtName_TextChanged(object sender, EventArgs e)
{
}

private void GenresForm_Load_1(object sender, EventArgs e)
{
    dgGenres.DataSource = genre.GenresList();
    dgGenres.ColumnHeadersDefaultCellStyle.ForeColor = Color.Blue;
    dgGenres.ColumnHeadersDefaultCellStyle.Font = new System.Drawing.Font("Arial",
        14, FontStyle.Bold);
    dgGenres.ColumnHeadersDefaultCellStyle.Alignment = DataGridViewContentAlignment.MiddleCenter;
    dgGenres.EnableHeadersVisualStyles = false;
}
}
}

```

MYDB



```

namespace LibraryManagementSystem
{
    using System;
    using System.Data;
    using System.Data.Entity;
    using System.Linq;
    using MySql.Data.MySqlClient;

    public class MYBD : DbContext
    {
        public MYBD()
            : base("name=MYBD")
        {
        }

        private MySqlConnection connection =
            new MySqlConnection("server=localhost;port=3306;username=root;password=;database=library");

        public void openConnection()
        {
            if (connection.State == System.Data.ConnectionState.Closed)
            {
                connection.Open();
            }
        }

        public void closeConnection()
        {
            if (connection.State == System.Data.ConnectionState.Open)
            {
                connection.Close();
            }
        }

        public MySqlConnection getConnection()
        {
            return connection;
        }

        public DataTable getData(string query, MySqlParameter[] parameters)
        {
            MySqlCommand command = new MySqlCommand(query, getConnection());

            if (parameters != null)
            {
                command.Parameters.AddRange(parameters);
            }

            DataTable table = new DataTable();
            MySqlDataAdapter adapter = new MySqlDataAdapter();

            adapter.SelectCommand = command;
            adapter.Fill(table);

            return table;
        }

        public int setData(string query, MySqlParameter[] parameters)
        {
            MySqlCommand command = new MySqlCommand(query, getConnection());

            if (parameters != null)
            {
                command.Parameters.AddRange(parameters);
            }

            openConnection();

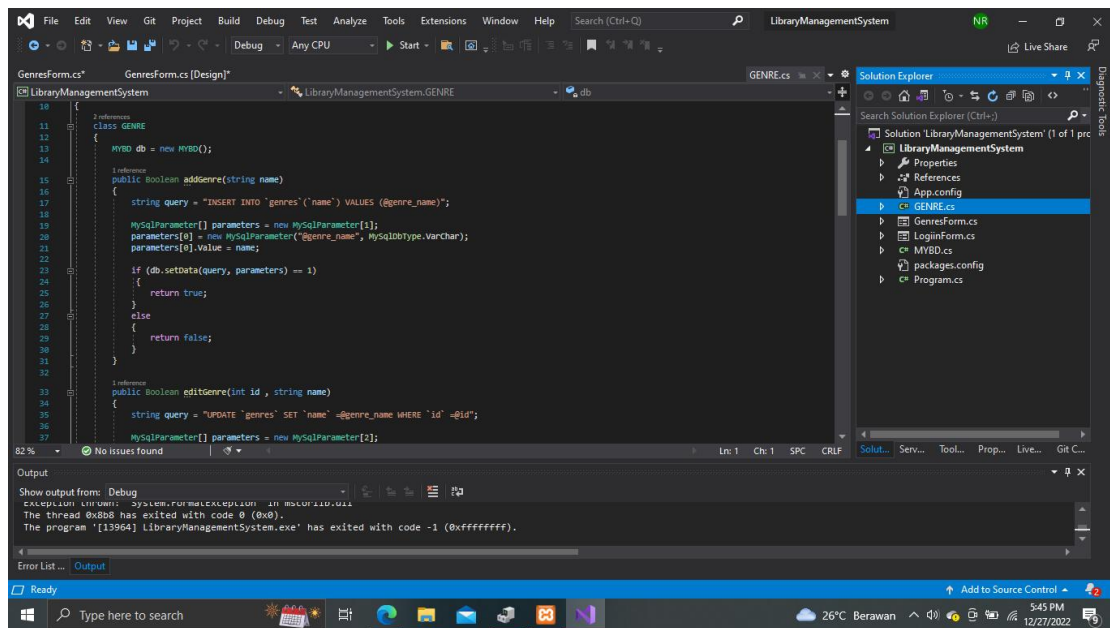
            int commandState = command.ExecuteNonQuery();

            closeConnection();

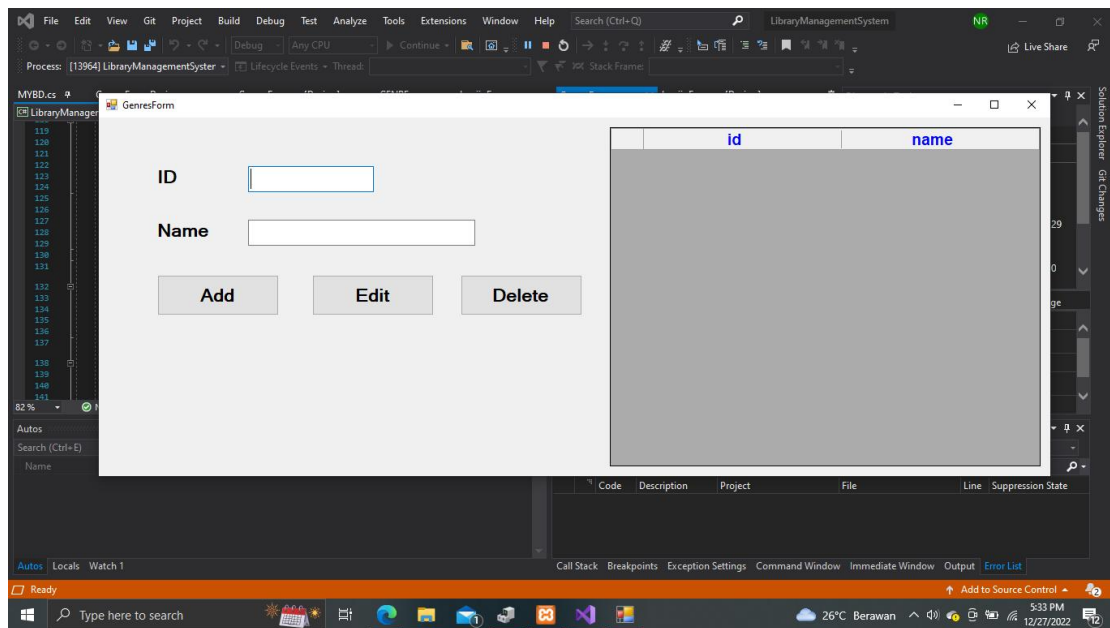
            return commandState;
        }
    }
}

```

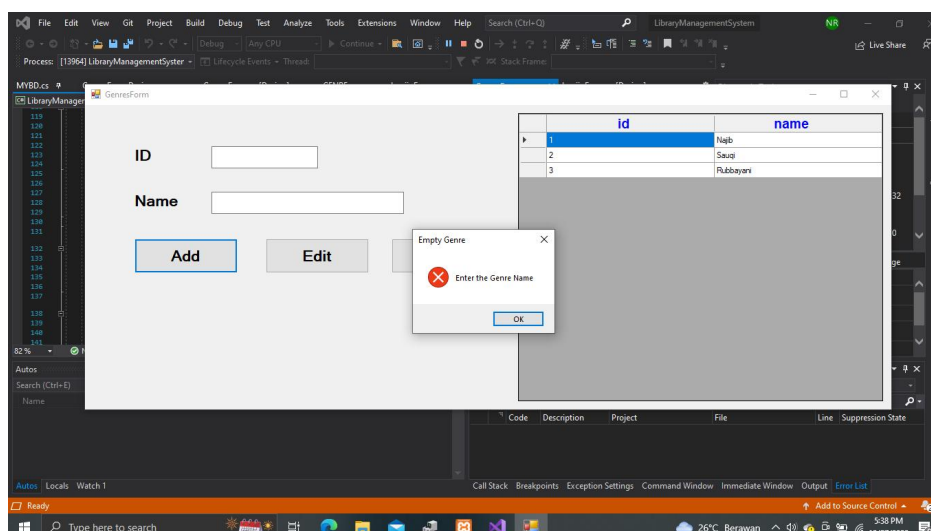
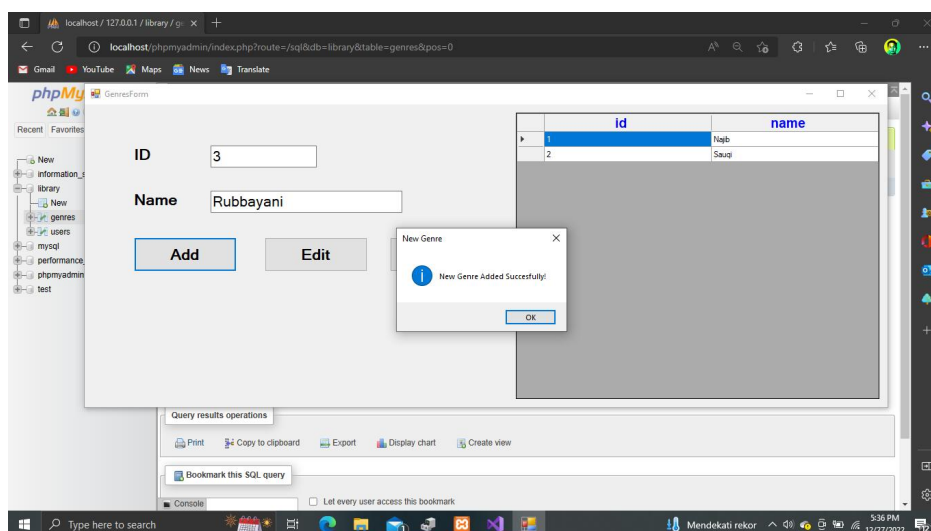
GENRE



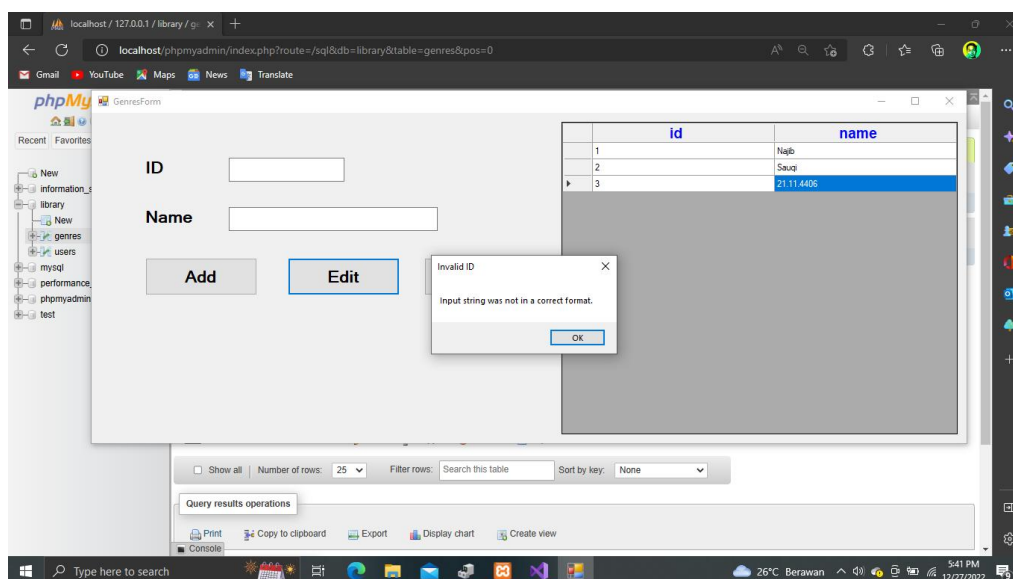
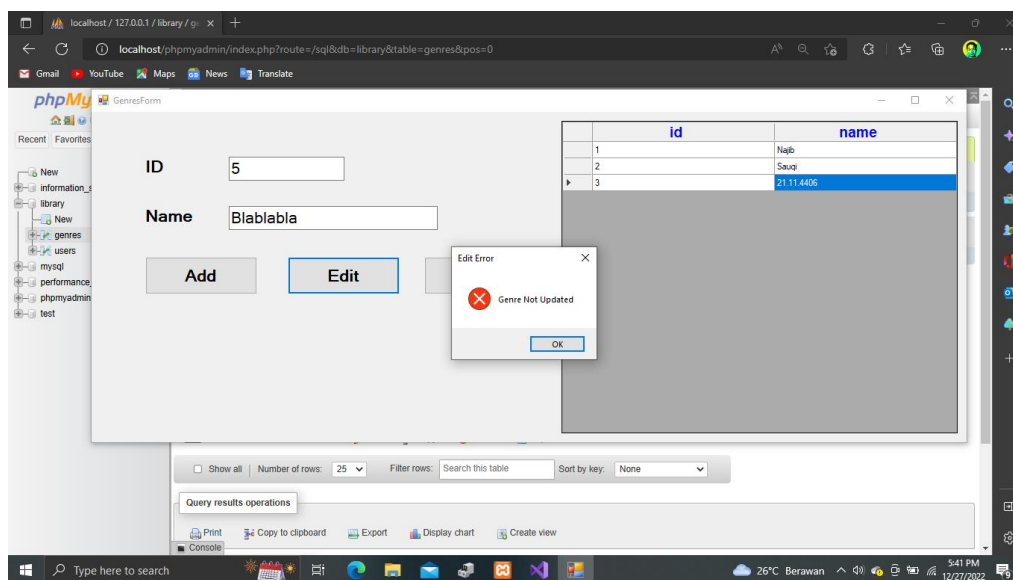
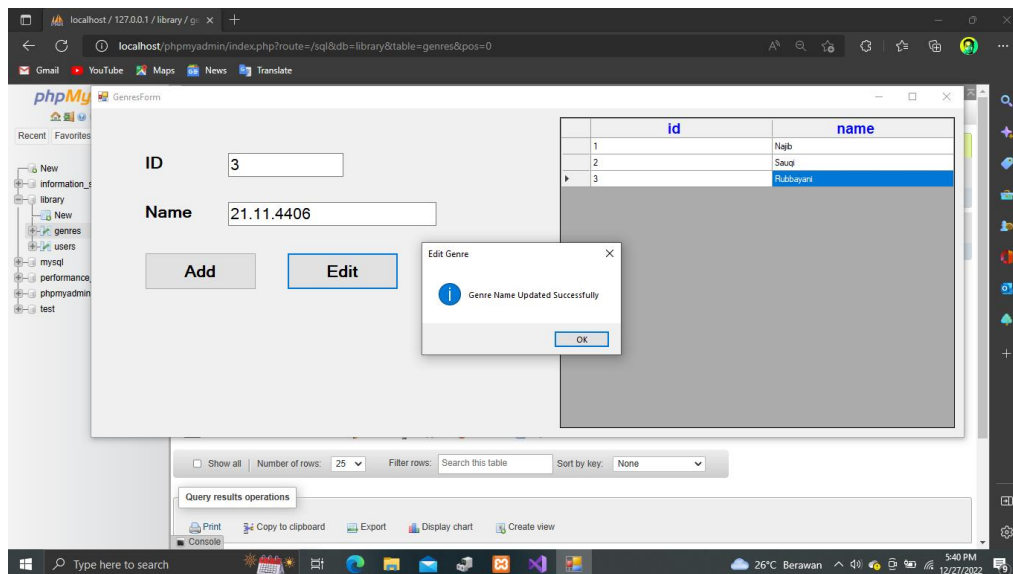
Start



Add



Edit



Delete

