DOTNET ASSIGNMENT

KHADEEJA BEEVI C N

ROLL NO :27

Design and implement a Windows Forms desktop application that serves as a "Simple Contact Manager". The application should allow users to add, view, and manage a list of contacts with the following features:

1. User Interface Elements:

o TextBoxes for entering the contact's Name, Email, and Phone Number.

o RadioButtons for selecting the contact's Gender (Male, Female, Other).

o CheckBox to indicate if the contact is a Favorite.

o ComboBox for selecting the contact's Country from a predefined list.

o Button to Add the contact to the list.

o ListView to display the list of added contacts with columns for Name, Email, Phone Number, Gender, Favorite status, and Country.

o TabControl with two tabs: one for Contact Entry and another for Contact List.

o GroupBox to organize the input fields and labels for better UI structure.

o Context Menu for the ListView to provide options like Edit, Delete, and Mark as Favorite.

2. Functionality:

o Implement event handling for the Add button to validate inputs and add the contact to the ListView.

o Enable context menu actions to allow users to edit or delete existing contacts from the ListView.

o Provide the ability to filter the displayed contacts based on the Favorite status.

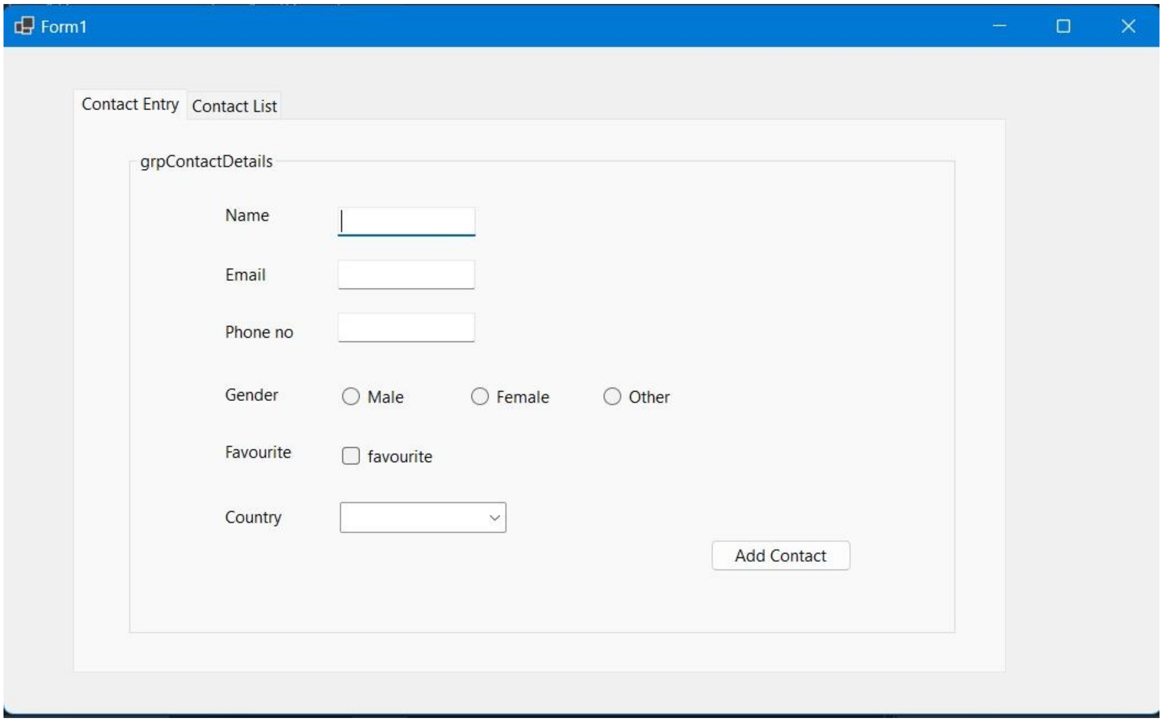
CODE :

```
namespace Simple_Contact_Manager
{
    public partial class Form1 : Form
    {

        public Form1()
        {
            InitializeComponent();
            editToolStripMenuItem.Click += EditToolStripMenuItem_Click;
            deleteToolStripMenuItem.Click += DeleteToolStripMenuItem_Click;
            markAsFavouriteToolStripMenuItem.Click += MarkAsFavouriteToolStripMenuItem_Click;

            listView1.MouseDown += listView1_MouseDown;

            // Context menu items
            contextMenuStrip1.Items.AddRange(new ToolStripItem[]
            {
                editToolStripMenuItem,
                deleteToolStripMenuItem,
                markAsFavouriteToolStripMenuItem
            });
            listView1.ContextMenuStrip = contextMenuStrip1;
        }

        private void listView1_MouseDown(object sender, MouseEventArgs e)
        {
            if (e.Button == MouseButtons.Right)
            {
                var hitTestInfo = listView1.HitTest(e.Location);
                if (hitTestInfo.Item != null)
                {
                    hitTestInfo.Item.Selected = true;
                }
            }
        }
        private void groupBox1_Enter(object sender, EventArgs e)
        {

        }
        private void radioButton1_CheckedChanged(object sender, EventArgs e)
        {

        }

        private void label3_Click(object sender, EventArgs e)
        {

        }

        private void contextMenuStrip1_Opening(object sender,
System.ComponentModel.CancelEventArgs e)
        {

        }

        private void Form1_Load(object sender, EventArgs e)
        {

        }
        private void textBox1_TextChanged(object sender, EventArgs e)
        {

        }

        private void button1_Click(object sender, EventArgs e)
        {
            if (string.IsNullOrEmpty(textBox1.Text) || string.IsNullOrEmpty(textBox2.Text) ||
string.IsNullOrEmpty(textBox3.Text) || comboBox1.SelectedIndex == -1)
            {
                MessageBox.Show("Please fill in all fields.");
                return;
            }
            string gender = radioButton1.Checked ? "Male" : radioButton2.Checked ? "Female" :
"Other";

            // Create new ListViewItem
            ListViewItem item = new ListViewItem(textBox1.Text);
            item.SubItems.Add(textBox2.Text);
            item.SubItems.Add(textBox3.Text);
            item.SubItems.Add(gender);
            item.SubItems.Add(checkBox1.Checked ? "Yes" : "No");
            item.SubItems.Add(comboBox1.SelectedItem.ToString());
            // Add item to ListView
            listView1.Items.Add(item);

            // Clear input fields
            textBox1.Clear();
            textBox2.Clear();
            textBox3.Clear();
            radioButton1.Checked = false;
            radioButton2.Checked = false;
            radioButton3.Checked = false;
            checkBox1.Checked = false;
            comboBox1.SelectedIndex = -1;

        }

        private void EditToolStripMenuItem_Click(object sender, EventArgs e)
        {
            if (listView1.SelectedItems.Count == 0) return;
            ListViewItem selectedItem = listView1.SelectedItems[0];

            // Fill input fields with selected item data
            textBox1.Text = selectedItem.SubItems[0].Text;
            textBox2.Text = selectedItem.SubItems[1].Text;
            textBox3.Text = selectedItem.SubItems[2].Text;
            switch (selectedItem.SubItems[3].Text)
            {
                case "Male":
                    radioButton1.Checked = true;
                    break;
                case "Female":
                    radioButton2.Checked = true;
                    break;
                case "Other":
                    radioButton3.Checked = true;
                    break;
            }
            checkBox1.Checked = selectedItem.SubItems[4].Text == "Yes";
            comboBox1.SelectedItem = selectedItem.SubItems[5].Text;

            // Remove selected item from ListView
            listView1.Items.Remove(selectedItem);
        }
```

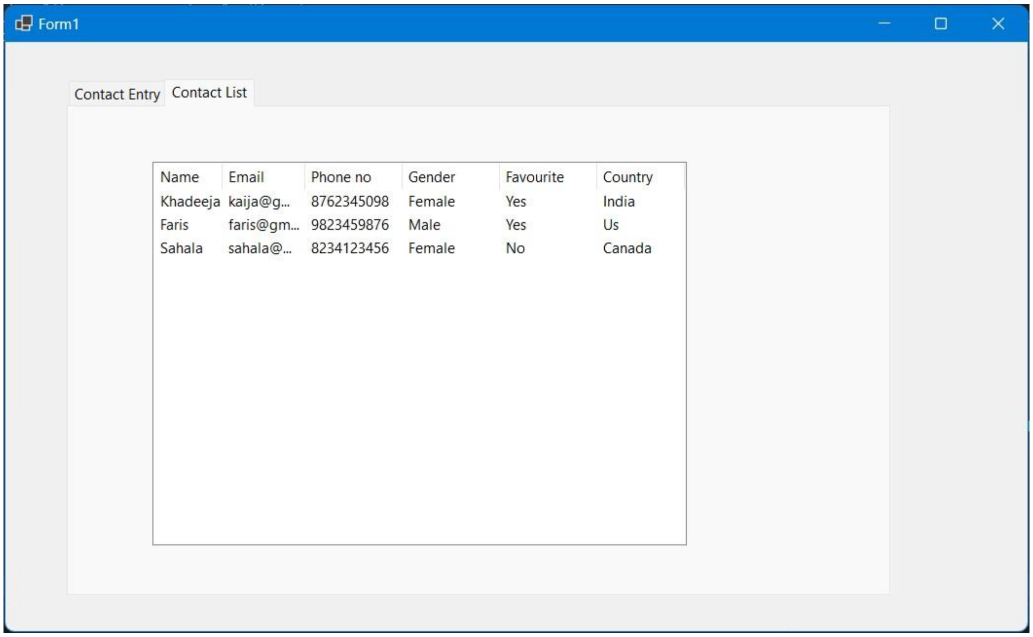
```
private void DeleteToolStripMenuItem_Click(object sender, EventArgs e)
{
   if (listView1.SelectedItems.Count == 0) return;
   listView1.Items.Remove(listView1.SelectedItems[0]);
}

private void MarkAsFavouriteToolStripMenuItem_Click(object sender, EventArgs e)
{
   if (listView1.SelectedItems.Count == 0) return;
   ListViewItem selectedItem = listView1.SelectedItems[0];
   selectedItem.SubItems[4].Text = "Yes";
}
private void FilterFavorites()
{
   var itemsToRemove = new List<ListViewItem>();

   // Collect items to remove
   foreach (ListViewItem item in listView1.Items)
   {
      if (item.SubItems[4].Text != "Yes")
      {
         itemsToRemove.Add(item);
      }
   }

   // Remove collected items
   foreach (var item in itemsToRemove)
   {
      listView1.Items.Remove(item);
   }
}

private void textBox2_TextChanged(object sender, EventArgs e)
{

}

private void textBox3_TextChanged(object sender, EventArgs e)
{

}

private void radioButton1_CheckedChanged_1(object sender, EventArgs e)
{
        }

        private void radioButton2_CheckedChanged(object sender, EventArgs e)
        {

        }

        private void radioButton3_CheckedChanged(object sender, EventArgs e)
        {

        }

        private void checkBox1_CheckedChanged(object sender, EventArgs e)
        {

        }

        private void comboBox1_SelectedIndexChanged(object sender, EventArgs e)
        {

        }
    }
}
```

OUTPUT :

FIRST SCREEN – CONTACT ENTRY :

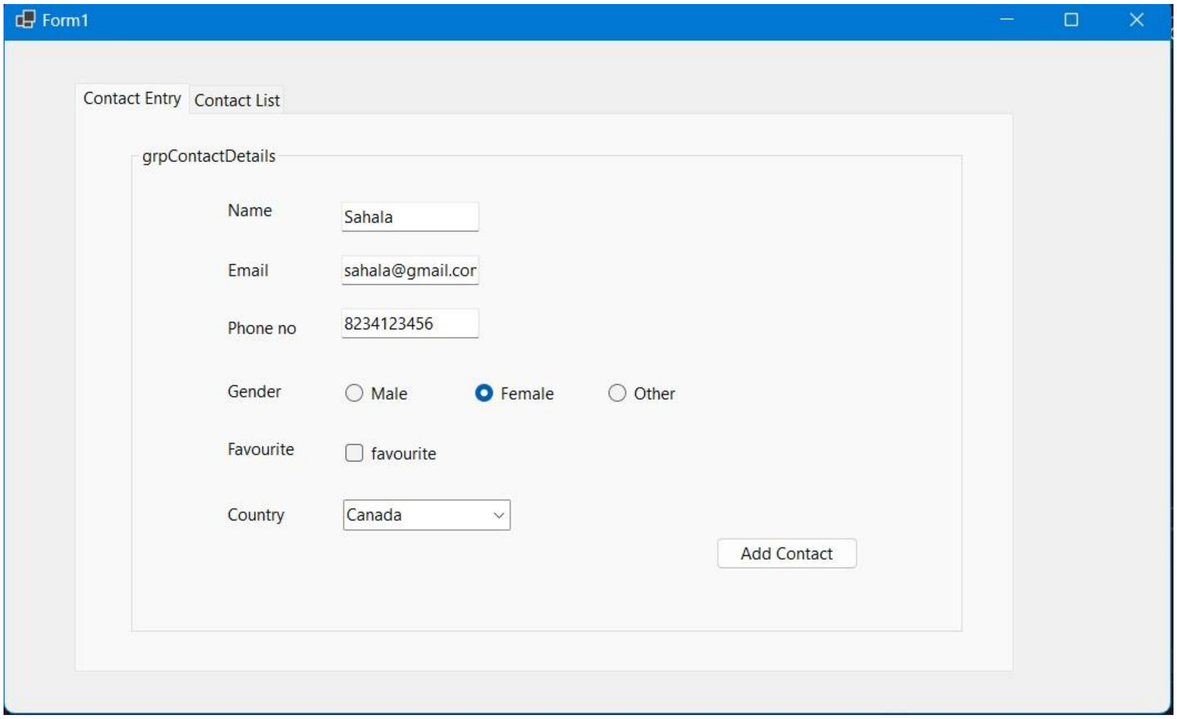


SECOND SCREEN – CONTACT LIST :



Name	Email	Phone no	Gender	Favourite	Country
Khadeeja	kaija@g...	8762345098	Female	Yes	India
Faris	faris@gm...	9823459876	Male	Yes	Us
Sahala	sahala@...	8234123456	Female	No	Canada

WHEN CLICK ON EDIT : (EDIT SAHALA)



Name: Sahala

Email: sahala@gmail.cor

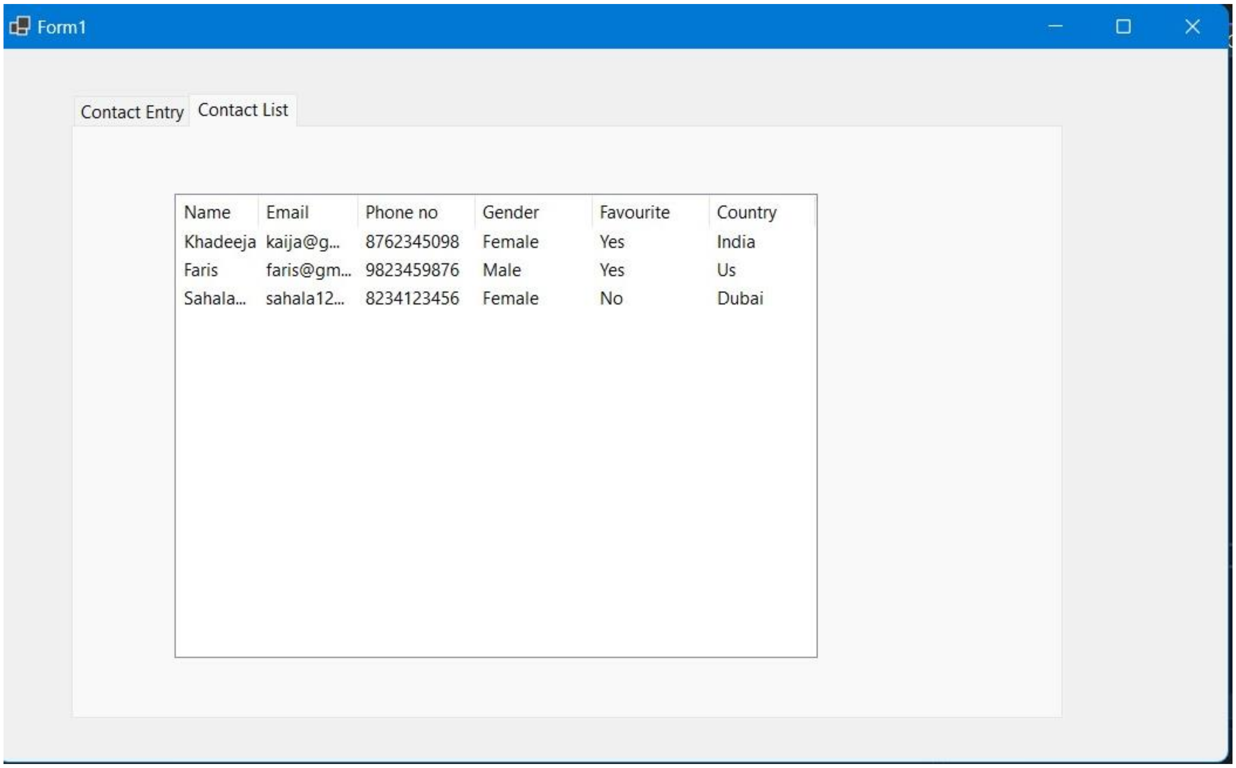
Phone no: 8234123456

Gender: ○ Male ● Female ○ Other

Favourite: □ favourite

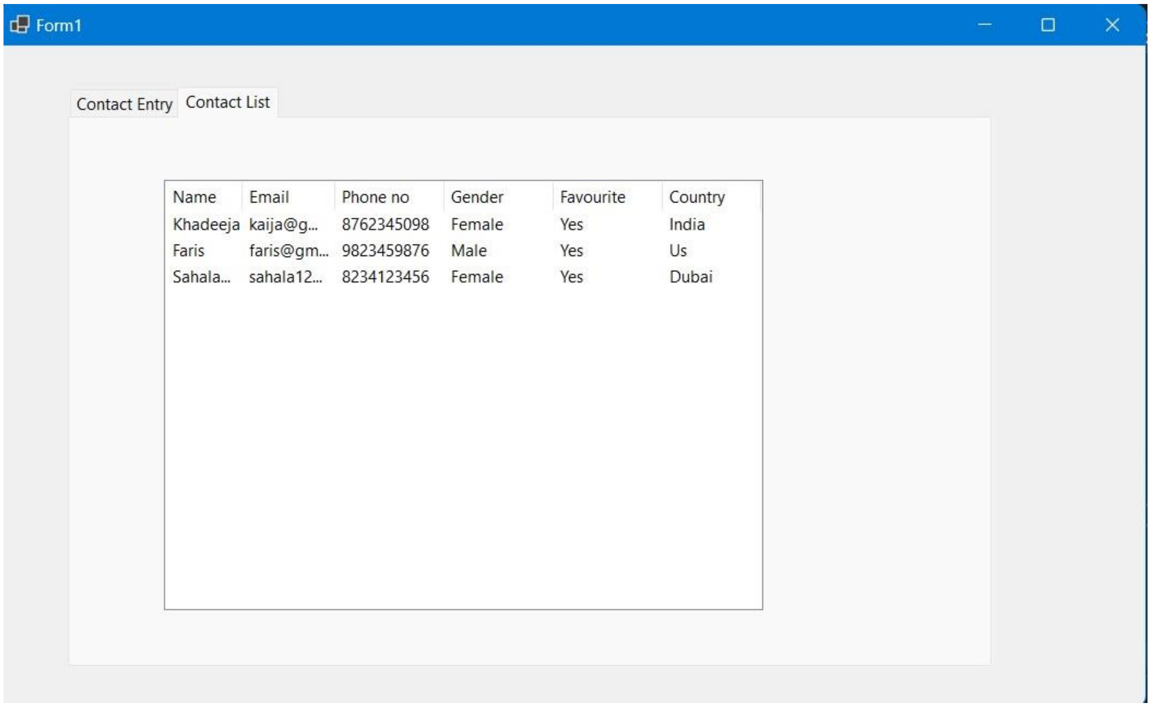
Country: Canada

AFTER EDITING : (UPDATED- COUNTRY & EMAIL FOR SAHALA)



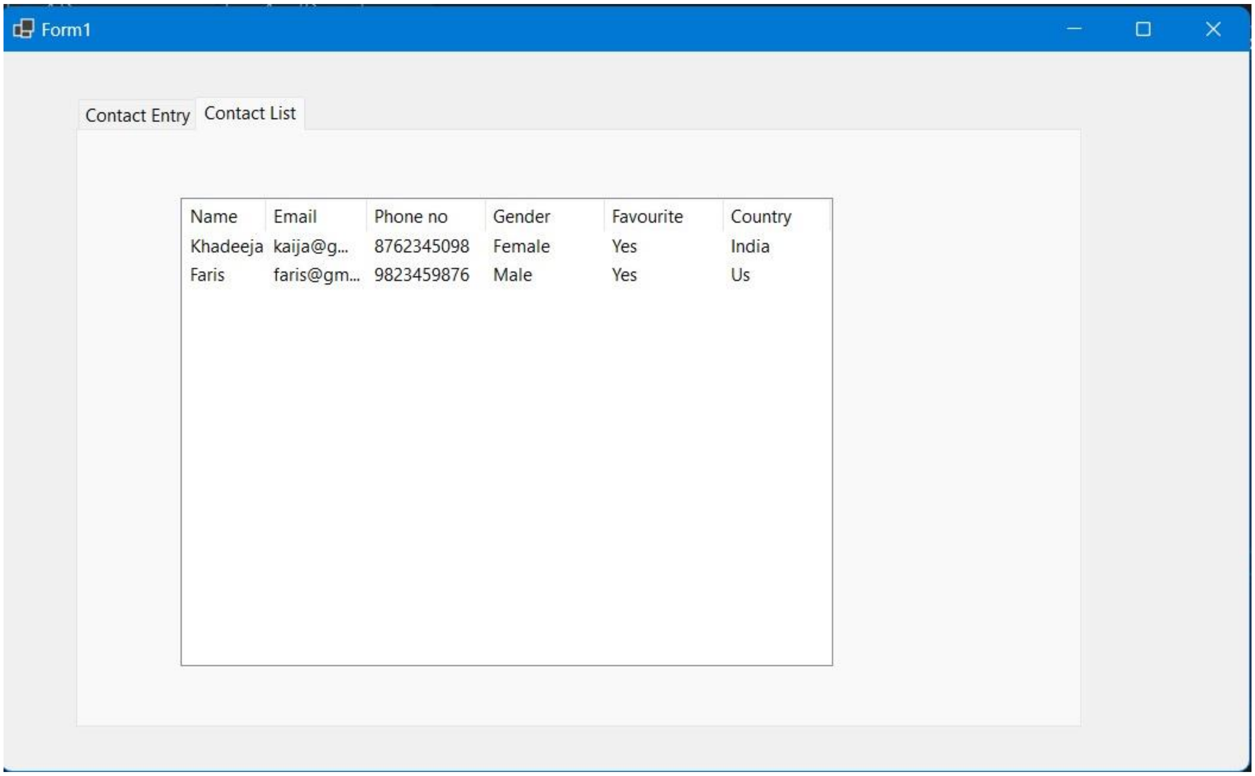
Name	Email	Phone no	Gender	Favourite	Country
Khadeeja	kaija@g...	8762345098	Female	Yes	India
Faris	faris@gm...	9823459876	Male	Yes	Us
Sahala...	sahala12...	8234123456	Female	No	Dubai

MARK AS FAVOURITE: (NO CHANGES TO YES FOR SAHALA)



Name	Email	Phone no	Gender	Favourite	Country
Khadeeja	kaija@g...	8762345098	Female	Yes	India
Faris	faris@gm...	9823459876	Male	Yes	Us
Sahala...	sahala12...	8234123456	Female	Yes	Dubai

AFTER DELETING : (SAHALA IS DELETED)



Name	Email	Phone no	Gender	Favourite	Country
Khadeeja	kaija@g...	8762345098	Female	Yes	India
Faris	faris@gm...	9823459876	Male	Yes	Us