

TEHNICI DE PROGRAMARE AVANSATA – CURS 2

Implementarea unui model simplu de vizualizare/actualizare date dintr-o tabela

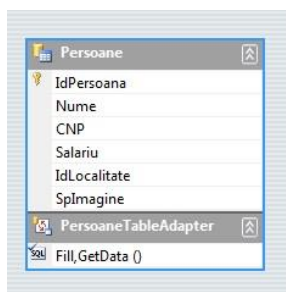
Modelul prezentat in continuare este un model simplu de vizualizare/actualizare tabela, ce functioneaza in urmatoarele conditii:

- Actualizarea se face direct in grid;
- Restrictiile de integritate (duplicat pe cheie semantica, stergere entitate referita in alta tabela) trebuiesc implementate la nivelul bazi de date.
- DataTable-ul nu provine dintr-un join intre mai multe tabele

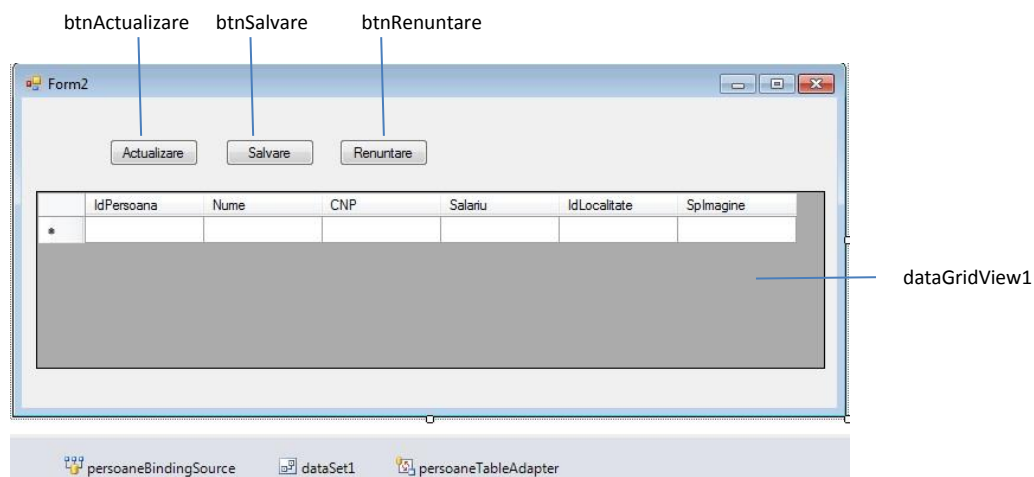
- Schema tablei din baza de date:

Persoane (IdPersoana, Nume, CNP, Salariu, IdLocalitate, SpImagine)

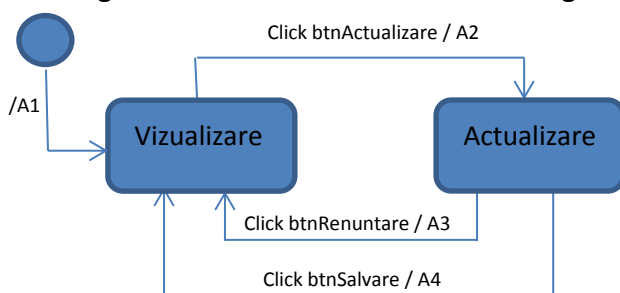
- Structura DataSet-ului (DataSet1):



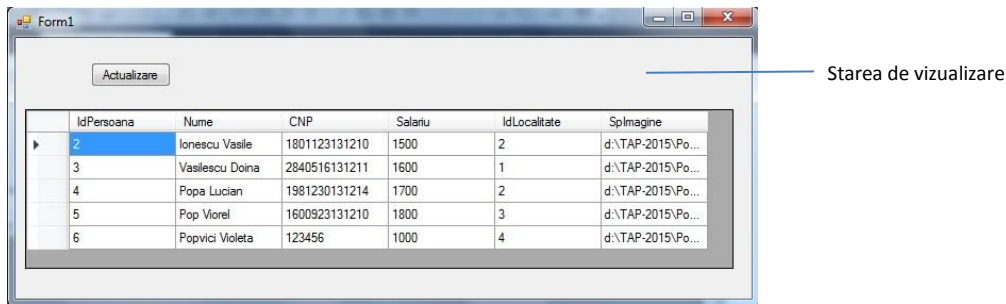
- Se structureaza form-ul dupa modelul urmator:



- Se conecteaza dataGridView1 la dataTable-ul Persoane din DataSet1.
- Logica de functionare este data de diagrama de stari:



- A1 - grid-ul nu permite actualizari; btnActualizare disponibil;
 btnRenuntare si btnSalvare invizibile; incarcare dataTable Persoane



- A2 - grid-ul permite actualizari; btnActualizare indisponibil;
 btnRenuntare, btnSalvare vizibile



- A3 - grid-ul nu permite actualizari; btnActualizare disponibil;
 btnRenuntare si btnSalvare invizibile; incarcare dataTable Persoane

- A4 - update baza de date;
 grid-ul nu permite actualizari; btnActualizare disponibil;
 btnRenuntare si btnSalvare invizibile; incarcare dataTable Persoane

▪ Optimizarea actiunilor:

Definim actiunea config avand un parametru de tip logic v.
 config (bool v):

```
//Grid-ul nu permite actualizari
dataGridView1.AllowUserToAddRows = !v;
dataGridView1.AllowUserToDeleteRows = !v;
dataGridView1.ReadOnly = v;
//btnActualizare disponibil
btnActualizare.Enabled = v;
//btnSalvare si btnRenuntare invizibile
btnSalvare.Visible = !v;
btnRenuntare.Visible = !v;
```

- Observatie: - pt. intrarea in starea de vizualizare se apeleaza metoda *config* cu parametrul *true* iar pt. intrarea in starea de actualizare parametrul este *false*.
 - pt. actiunea *incarcare dataTable Persoane* vom defini metoda *refresh*.

A1 – config(true); refresh

A2 - config(false)

A3 - config(true); refresh

A4 - update baza de date; config(true); refresh

▪ Codul :

```
using System;
using System.Windows.Forms;

namespace C02
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();

            private void config(bool v){
                dataGridView1.AllowUserToAddRows = !v;
                dataGridView1.AllowUserToDeleteRows = !v;
                dataGridView1.ReadOnly = v;

                btnActualizare.Enabled = v;

                btnSalvare.Visible = !v;
                btnRenuntare.Visible = !v;
            }

            private void refresh()
            {
                persoaneTableAdapter.Fill(dataSet1.Persoane);
            }

            private void Form1_Load(object sender, EventArgs e)
            {
                //A1
                config(true);
                refresh();
            }

            private void btnActualizare_Click(object sender, EventArgs e)
            {
                //A2
                config(false);
            }

            private void btnSalvare_Click(object sender, EventArgs e)
            {
                //A4
                try
                {
                    persoaneTableAdapter.Update(dataSet1.Persoane);
                    config(true);
                    refresh();
                }
                catch (Exception exc) {
                    MessageBox.Show(exc.Message);
                }
            }

            private void btnRenuntare_Click(object sender, EventArgs e){
                //A3
                config(true);
                refresh();
            }
        }
    }
}
```

- Tratarea erorilor de tip *format eronat* pentru date calendaristice si date numerice.
Se creaza metoda de tratare a evenimentului *DataError* pentru controlul *grid* .
Se completeaza codul astfel:

```
private void dataGridView1_DataError(object sender, DataGridViewDataErrorEventArgs e){
    MessageBox.Show("Format eronat");
}
```

- Confirmarea stgererii

```
private void dataGridView1_UserDeletingRow(object sender, DataGridViewRowCancelEventArgs e)
{
    const string mesaj = "Confirmati stergerea";
    const string titlu = "Stergere inregistrare";
    var rezultat = MessageBox.Show(mesaj, titlu, MessageBoxButtons.YesNo,
                                   MessageBoxIcon.Warning );
    if (rezultat == DialogResult.No) e.Cancel = true;
}
```

- Tratarea erorilor de tip
 - *Duplicat pe cheie semantica* si
 - *Stergere inregistrare referita din alta tabela*

Se definesc mai intai aceste restrictii la nivelul bazei de date:

Se inlocuieste corpul metodei *btnSalvare_Click* cu:

```
private void btnSalvare_Click(object sender, EventArgs e)
{
    try
    {
        persoaneTableAdapter.Update(dataSet1.Persoane);
        config(true);
        refresh();
    }
    catch (Exception exc) {
        string s = exc.Message;

        if (s.IndexOf("duplicate values") > 0)
            MessageBox.Show("Inregistrare deja existenta !");
        else if (s.IndexOf("cannot be deleted") > 0)
            MessageBox.Show("Ati sters inregistrari referite in alte tabele !");
    }
}
```

- Definirea unei clase (*Model1*) cu elementele generale ale modelului
- Click dr. pe numele aplicatiei (in Solution Explorer) => Add => Class (cu numele *Model1*)

```
using System;
using System.Windows.Forms;
```

```
namespace C02
```

```
{
    class Model1
    {
        public static void Config(bool v,DataGridView dg, Button[] l)
        {
            //Grid-ul nu permite actualizari
            dg.AllowUserToAddRows = !v;
            dg.AllowUserToDeleteRows = !v;
            dg.ReadOnly = v;
        }
    }
}
```

```

        //btnActualizare disponibil
        l[0].Enabled = v;
        //btnSalvare si btnRenuntare invizibile
        l[1].Visible = !v;
        l[2].Visible = !v;
    }

    public static void trateazaEroare(Exception e) {
        string s = e.Message;
        //MessageBox.Show(s);
        if (s.IndexOf("duplicate values") > 0)
            MessageBox.Show("Inregistrare deja existenta !");
        else if (s.IndexOf("cannot be deleted") > 0)
            MessageBox.Show("Ati sters inregistrari referite in alte tabele !");
    }

    public static bool confirmare()
    {
        const string mesaj = "Confirmati stergerea";
        const string titlu = "Stergere inregistrare";
        DialogResult rezultat = MessageBox.Show(mesaj, titlu,
            MessageBoxButtons.YesNo,
            MessageBoxIcon.Warning);
        if (rezultat == DialogResult.No) return false;
        else return true;
    }
}

```

- Modificam codul form-ului ce implementeaza modelul prezentat:

```

using System;
using System.Windows.Forms;

namespace C03
{
    public partial class Form2 : Form
    {
        private Button[] lb = new Button[3];

        public Form2()
        {
            InitializeComponent();
            lb[0] = btnActualizare;
            lb[1] = btnSalvare;
            lb[2] = btnRenuntare;
        }

        private void refresh()
        {
            persoaneTableAdapter.Fill(dataSet1.Persoane);
        }

        private void Form2_Load(object sender, EventArgs e)
        {
            Model1.Config(true, dataGridView1, lb);
            refresh();
        }

        private void btnActualizare_Click(object sender, EventArgs e)
        {
            Model1.Config(false, dataGridView1, lb);
        }

        private void btnSalvare_Click(object sender, EventArgs e)

```

```

{
    try
    {
        persoaneTableAdapter.Update(dataSet1.Persoane);
        Model1.Config(true, dataGridView1, lb);
        refresh();
    }
    catch (Exception exc)
    {
        Model1.trateazaEroare(exc);
    }
}

private void btnRenuntare_Click(object sender, EventArgs e)
{
    Model1.Config(true, dataGridView1, lb);
    refresh();
}

private void dataGridView1_DataError(object sender, DataGridViewDataErrorEventArgs e)
{
    MessageBox.Show("Format eronat");
}

private void dataGridView1_UserDeletingRow(object sender, DataGridViewRowCancelEventArgs e)
{
    if (!Model1.confirmare()) e.Cancel = true;
}
}
}

```