

1. O aplicație conține o boxă textuală și 2 butoane. Boxa textuală legată de câmpul LL, al tabelului 33. Butoanele permit navigarea înainte și înapoi.

Codul programului:

```
String conStr;
OleDbConnection con;
OleDbDataAdapter da;
DataSet ds;
int row;
int max_row;

public Form1(){
    InitializeComponent();
    conStr = @"Provider=Microsoft.Jet.OLEDB.4.0;Data Source=D:\DBLab1.mdb";
    con = new OleDbConnection(conStr);
    Load();
}

private void Load() {
    string query = "SELECT * FROM 33";
    da = new OleDbDataAdapter(query, con);
    ds = new DataSet();
    da.FillSchema(ds, SchemaType.Source, "33");
    da.Fill(ds, "33");
    max_row=ds.Tables["33"].Rows.Count;
    row = 0;
    textBox1.Text = ds.Tables["33"].Rows[row]["LL"].ToString();
}

private void button1_Click(object sender, EventArgs e){
    if (row > 0) { row = row - 1; }
    textBox1.Text = ds.Tables["33"].Rows[row]["LL"].ToString();
}

private void button2_Click(object sender, EventArgs e){
    if (row < max_row-1){ row = row + 1; }
    textBox1.Text = ds.Tables["33"].Rows[row]["LL"].ToString();
}
```

2. Transformarea tabelului product în fișier Xml cu scrierea pe disc, 5 coloane care vor fi noduri, iar prima coloana va fi atributul nodului primar. Actiunea se realizează prin 2 metode: a) Metoda DOM, b) fără metoda DOM.

Codul programului. - AICI ESTE REALIZAT NUMAI METODA a)

```
OleDbDataAdapter da;
DataSet ds;
OleDbConnection con;

public Form1()
{
```

```

        InitializeComponent();
        con = new OleDbConnection(@"Provider=Microsoft.Jet.OLEDB.4.0;Data
Source='D:\DBLab1.mdb'");
        da = new OleDbDataAdapter();
        ds = new DataSet();
        con.Open();
    }

```

```

private void button1_Click_1(object sender, EventArgs e)
{
    if (richTextBox1.Text != "")
    {
        Stream stream;
        SaveFileDialog sd = new SaveFileDialog();
        sd.Filter = "xml files (*.xml)|*.xml|All files (*.*)|*.*";
        sd.FileName = "ID";

        if (sd.ShowDialog() == DialogResult.OK)
        {
            if ((stream = sd.OpenFile()) != null)
            {
                StreamWriter ftext = new StreamWriter(stream);
                ftext.Write(richTextBox1.Text);
                ftext.Close();
                stream.Close();
            }
        }
    }
}

```

```

private void button2_Click(object sender, EventArgs e)
{
    ds.Clear();
    da.SelectCommand = new OleDbCommand("SELECT ID, col1, col2, col3, col4 FROM
PRODUCT", con);
    da.Fill(ds, "PRODUCT");

    StringWriter str = new StringWriter();
    XmlTextWriter xw = new XmlTextWriter(str);
    xw.WriteStartDocument();
    xw.Formatting = Formatting.Indented;
    xw.WriteStartElement("ID");
    foreach (DataRow dr in ds.Tables["PRODUCT"].Rows)
    {
        xw.WriteStartElement("ID");
        foreach (DataColumn dc in ds.Tables["PRODUCT"].Columns)
        {
            if (dc.ColumnName == "ID")
            {
                xw.WriteAttributeString("id", dr["ID"].ToString());
            }
        }
    }
}

```

```
        else
        {
            xw.WriteElementString(dc.ColumnName, dr[dc.ColumnName].ToString());
        }
    }
    xw.WriteEndElement();
}
xw.WriteEndElement();
xw.WriteEndDocument();
richTextBox1.Text = str.ToString();
}
```