# Министерство науки и высшего образования Российской Федерации

## Федеральное государственное бюджетное образовательное учреждение высшего образования

## «Новосибирский государственный технический университет»

NSTU_Logo_blue

## Кафедра теоретической и прикладной информатики

### Лабораторная работа № 8 по дисциплине «Базы данных»

Вариант 4

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| сигма градиент синий1 | Факультет: | ПМИ |  |  |
| Группа: | ПМ-63 |  |  |
| Студенты: | Кожекин М.В.  Утюганов Д.С. |  |  |
|  |  |  |  |
| Преподаватели: | Стасышина Т.Л.  Хайленко Е.А. |  |  |

Новосибирск

2019

**Разработать программу при помощи SQL и ADO NET:**

1. Получить информацию о выручке поставщиков, сделавших больше всего поставок для указанного изделия

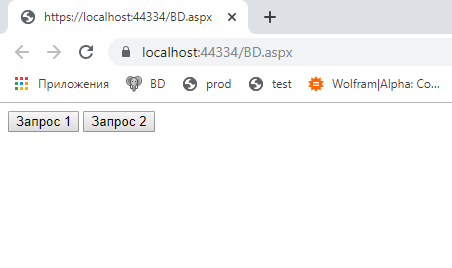
|  |
| --- |
| -- Получить информацию о выручке поставщиков, сделавших больше всего поставок  -- для указанного изделия  select n\_post, sum(t2.kol\*t2.cost) as revenue  from pmib6306.spj1 t2  where t2.n\_post = ( -- максимальное число поставленных деталей                      select t1.n\_post                      from pmib6306.spj1 t1                      where t1.n\_izd = ?                      group by t1.n\_post                      order by sum(t1.kol) desc                      limit 1 )  group by t2.n\_post |

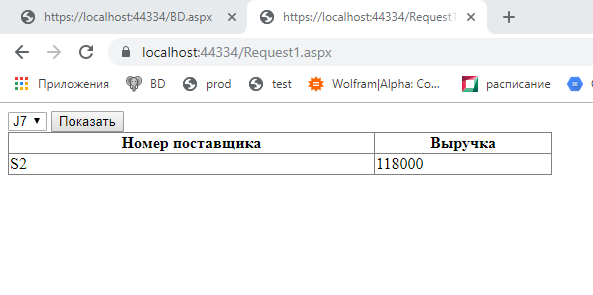
2. Для указанного поставщика удвоить количество деталей в поставках за заданный период.

|  |
| --- |
| -- Удвоение числа деталей  update pmib6306.spj1  set kol = kol \* 2  where n\_post = ? and date\_post between ? and ?; |

|  |
| --- |
| -- Просмотр  select \*  from pmib6306.spj1  where n\_post = ? and date\_post between ? and ? |

**Результат работы программы**





|  |  |
| --- | --- |
|  |  |

**Текст программы**

BD.aspx

|  |
| --- |
| <%@ Page Language="C#" AutoEventWireup="true" CodeBehind="BD.aspx.cs" Inherits="bd\_8.WebForm1" %>  <!DOCTYPE html>  <html xmlns="http://www.w3.org/1999/xhtml">  <head runat="server">      <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />      <title></title>  </head>  <body>      <form id="form1" runat="server" dir="auto">          <asp:Button ID="Button1" runat="server" OnClientClick="window.open('Request1.aspx');" Text="Запрос 1" />          <asp:Button ID="Button2" runat="server" OnClientClick="window.open('Request2.aspx');" Text="Запрос 2" />          <br />          <br />          <asp:SqlDataSource ID="SqlDataSource1" runat="server" ConnectionString="<%$ ConnectionStrings:studentsConnectionString\_pmib6306 %>" ProviderName="<%$ ConnectionStrings:studentsConnectionString\_pmib6306.ProviderName %>" SelectCommand="select \*  from pmib6306.spj1"></asp:SqlDataSource>      </form>  </body>  </html> |
| using System;  using System.Data;  using System.Collections.Generic;  using System.Data.Odbc;  using System.Linq;  using System.Web;  using System.Web.UI;  using System.Web.UI.WebControls;  using System.Windows.Forms;  namespace bd\_8  {      public partial class WebForm1 : System.Web.UI.Page      {          protected void Page\_Load(object sender, EventArgs e)          {          }        }  } |

Request1.aspx

|  |
| --- |
| <%@ Page Language="C#" AutoEventWireup="true" CodeBehind="Request1.aspx.cs" Inherits="bd\_8.Request1" %>  <!DOCTYPE html>  <html xmlns="http://www.w3.org/1999/xhtml">  <head runat="server">  <meta http-equiv="Content-Type" content="text/html; charset=utf-8"/>      <title></title>  </head>  <body>      <form id="form1" runat="server">          <div>          <asp:Label ID="Label1" runat="server" Text="Label" Visible="False"></asp:Label>              <asp:DropDownList ID="DropDownList1" runat="server" AutoPostBack="True"></asp:DropDownList>              <asp:Button ID="Button1" runat="server" OnClick="Button1\_Click" Text="Показать" />          <asp:GridView ID="GridView1" runat="server" AutoGenerateColumns="True" Width="544px" OnSelectedIndexChanged="GridView1\_SelectedIndexChanged">          </asp:GridView>          </div>      </form>  </body>  </html> |
| using System;  using System.Data;  using System.Collections.Generic;  using System.Data.Odbc;  using System.Linq;  using System.Web;  using System.Web.UI;  using System.Web.UI.WebControls;  using System.Windows.Forms;  namespace bd\_8  {      public partial class Request1 : System.Web.UI.Page      {          protected void Page\_Load(object sender, EventArgs e)          {              Reload();          }            protected void Request\_1()          {              // Создаем объект подключения              OdbcConnection conn = new OdbcConnection();              // Задаем параметр подключения – имя ODBC-источника              conn.ConnectionString = "Dsn=PostgreSQL30;database=students;server=students.ami.nstu.ru;port=5432;uid=pmi-b6306;pwd=Jemlift5;sslmode=disable;readonly=0;protocol=7.4;fakeoidindex=0;showoidcolumn=0;rowversioning=0;showsystemtables=0;fetch=100;unknownsizes=0;maxvarcharsize=255;maxlongvarcharsize=8190;debug=0;commlog=0;usedeclarefetch=0;textaslongvarchar=1;unknownsaslongvarchar=0;boolsaschar=1;parse=0;lfconversion=1;updatablecursors=1;trueisminus1=0;bi=0;byteaaslongvarbinary=1;useserversideprepare=1;lowercaseidentifier=0;d6=-101;xaopt=1";              // Подключаемся к БД              conn.Open();              // Определяем строку с текстом запроса              string strSQL = @"  -- Получить информацию о выручке поставщиков, сделавших больше всего поставок                                  -- для указанного изделия                                  select n\_post, sum(t2.kol\*t2.cost) as revenue                                  from pmib6306.spj1 t2                                  where t2.n\_post = ( -- максимальное число поставленных деталей                                                      select t1.n\_post                                                      from pmib6306.spj1 t1                                                      where t1.n\_izd = ?                                                      group by t1.n\_post                                                      order by sum(t1.kol) desc                                                      limit 1 )                                  group by t2.n\_post                                ";                // Создаем объект запроса              OdbcCommand cmd = new OdbcCommand(strSQL, conn);              // Создаем первый параметр              OdbcParameter par1 = new OdbcParameter();              par1.ParameterName = "@vn\_izd";              par1.OdbcType = OdbcType.Text;              par1.Value = DropDownList1.SelectedValue;              //Добавляем первый параметр в коллекцию              cmd.Parameters.Add(new OdbcParameter("@vn\_izd", DropDownList1.SelectedValue));              cmd.Parameters.Add(new OdbcParameter("@vn\_izd", DropDownList1.SelectedValue));              cmd.Parameters.Add(new OdbcParameter("@vn\_izd", DropDownList1.SelectedValue));              // Объявляем объект транзакции              OdbcTransaction tx = null;              try              {                  // Начинаем транзакцию и извлекаем объект транзакции из объекта подключения.                  tx = conn.BeginTransaction();                  // Включаем объект SQL-команды в транзакцию                  cmd.Transaction = tx;                  // Выполняем SQL-команду и получаем количество обработанных записей                  OdbcDataReader res = cmd.ExecuteReader();                  DataTable result\_table = new DataTable();                  using (DataColumn fNameColumn = new DataColumn())                  {                      fNameColumn.DataType = System.Type.GetType("System.String");                      fNameColumn.ColumnName = "Номер поставщика";                      result\_table.Columns.Add(fNameColumn);                  }                  using (DataColumn fNameColumn = new DataColumn())                  {                      fNameColumn.DataType = System.Type.GetType("System.String");                      fNameColumn.ColumnName = "Выручка";                      result\_table.Columns.Add(fNameColumn);                  }                  while (res.Read())                  {                      DataRow new\_row;                      new\_row = result\_table.NewRow();                      new\_row["Номер поставщика"] = res[0];                      new\_row["Выручка"] = res[1];                      result\_table.Rows.Add(new\_row);                  }                  // Подтверждаем транзакцию                  GridView1.DataSource = result\_table;                  GridView1.DataBind();                  tx.Commit();              }              catch (Exception ex)              {                  // При возникновении любой ошибки                  // Формируем сообщение об ошибке                  Label1.Text = ex.Message;                  // выполняем откат транзакции                  tx.Rollback();              }              //закрываем соединение              conn.Close();          }          protected void GridView1\_SelectedIndexChanged(object sender, EventArgs e)          {              // Создаем объект подключения              OdbcConnection conn = new OdbcConnection();              // Задаем параметр подключения – имя ODBC-источника              conn.ConnectionString = "Dsn=PostgreSQL30;database=students;server=students.ami.nstu.ru;port=5432;uid=pmi-b6306;pwd=Jemlift5;sslmode=disable;readonly=0;protocol=7.4;fakeoidindex=0;showoidcolumn=0;rowversioning=0;showsystemtables=0;fetch=100;unknownsizes=0;maxvarcharsize=255;maxlongvarcharsize=8190;debug=0;commlog=0;usedeclarefetch=0;textaslongvarchar=1;unknownsaslongvarchar=0;boolsaschar=1;parse=0;lfconversion=1;updatablecursors=1;trueisminus1=0;bi=0;byteaaslongvarbinary=1;useserversideprepare=1;lowercaseidentifier=0;d6=-101;xaopt=1";              // Подключаемся к БД              conn.Open();              // Определяем строку с текстом запроса              //string strSQL = "UPDATE pmib6306.p SET name = ?  where  n\_det = ? ";              string strSQL = "select \* from pmib6306.spj1";              // Создаем объект запроса              OdbcCommand cmd = new OdbcCommand(strSQL, conn);              // Создаем первый параметр              OdbcParameter par\_name = new OdbcParameter();              par\_name.ParameterName = "@vname";              par\_name.OdbcType = OdbcType.Text;              par\_name.Value = "Кулер";              // Добавляем первый параметр в коллекцию              // cmd.Parameters.Add(par\_name);              // Создаем второй параметр              OdbcParameter par\_town = new OdbcParameter();              par\_town.ParameterName = "@vn\_det";              par\_town.OdbcType = OdbcType.Text;              par\_town.Value = "P3";              // Добавляем второй параметр в коллекцию.              //  cmd.Parameters.Add(par\_town);              // Объявляем объект транзакции              OdbcTransaction tx = null;              try              {                  // Начинаем транзакцию и извлекаем объект транзакции из объекта подключения.                  tx = conn.BeginTransaction();                  // Включаем объект SQL-команды в транзакцию                  cmd.Transaction = tx;                  // Выполняем SQL-команду и получаем количество обработанных записей                  int i = cmd.ExecuteNonQuery();                  // Подтверждаем транзакцию                  tx.Commit();              }              catch (Exception ex)              {                  // При возникновении любой ошибки                  // Формируем сообщение об ошибке                  Label1.Text = ex.Message;                  // выполняем откат транзакции                  tx.Rollback();              }              //закрываем соединение              conn.Close();          }          protected void FillDropDownList1()          {              // Создаем объект подключения              OdbcConnection conn = new OdbcConnection();              // Задаем параметр подключения – имя ODBC-источника              conn.ConnectionString = "Dsn=PostgreSQL30;database=students;server=students.ami.nstu.ru;port=5432;uid=pmi-b6306;pwd=Jemlift5;sslmode=disable;readonly=0;protocol=7.4;fakeoidindex=0;showoidcolumn=0;rowversioning=0;showsystemtables=0;fetch=100;unknownsizes=0;maxvarcharsize=255;maxlongvarcharsize=8190;debug=0;commlog=0;usedeclarefetch=0;textaslongvarchar=1;unknownsaslongvarchar=0;boolsaschar=1;parse=0;lfconversion=1;updatablecursors=1;trueisminus1=0;bi=0;byteaaslongvarbinary=1;useserversideprepare=1;lowercaseidentifier=0;d6=-101;xaopt=1";              // Подключаемся к БД              conn.Open();              string strSQL = "select n\_izd from pmib6306.j";              // Создаем объект запроса              OdbcCommand cmd = new OdbcCommand(strSQL, conn);              OdbcTransaction tx = null;              try              {                  // Начинаем транзакцию и извлекаем объект транзакции из объекта подключения.                  tx = conn.BeginTransaction();                  // Включаем объект SQL-команды в транзакцию                  cmd.Transaction = tx;                  // Выполняем SQL-команду и получаем количество обработанных записей                  OdbcDataReader res = cmd.ExecuteReader();                    while (res.Read())                      DropDownList1.Items.Add(res[0].ToString());                  // Подтверждаем транзакцию                  tx.Commit();              }              catch (Exception ex)              {                  // При возникновении любой ошибки                  // Формируем сообщение об ошибке                  Label1.Text = ex.Message;                  // выполняем откат транзакции                  tx.Rollback();              }              //закрываем соединение              conn.Close();          }          protected void Button1\_Click(object sender, EventArgs e)          {              Request\_1();              Reload();          }          private void Reload()          {              var SelVal = DropDownList1.SelectedValue;              DropDownList1.Items.Clear();              FillDropDownList1();              DropDownList1.SelectedValue = SelVal;          }      }  } |

Request2.aspx

|  |
| --- |
| <%@ Page Language="C#" AutoEventWireup="true" CodeBehind="Request2.aspx.cs" Inherits="bd\_8.Request2" %>  <!DOCTYPE html>  <html xmlns="http://www.w3.org/1999/xhtml">  <head runat="server">  <meta http-equiv="Content-Type" content="text/html; charset=utf-8"/>      <title></title>  </head>  <body>      <form id="form1" runat="server">          <div>          <asp:Label ID="Label2" runat="server" Text="Label" Visible="False"></asp:Label>          <asp:TextBox ID="TextBox1" runat="server" Text="2011-01-11" Width="150px"></asp:TextBox>          <asp:DropDownList ID="DropDownList2" runat="server" Width="102px">          </asp:DropDownList>          <asp:TextBox ID="TextBox2" runat="server" Text="2013-01-11" Width="150px"></asp:TextBox>          <asp:Button ID="Button1" runat="server" OnClick="Button1\_Click" Text="Показать" Width="102px" />              <br />              <asp:Button ID="Button2" runat="server" OnClick="Button2\_Click" Text="Увеличить количество в 2 раза" Width="260px" />              <asp:Button ID="Button3" runat="server" OnClick="Button3\_Click" Text="Уменьшить количество в 2 раза" Width="260px" />          <asp:GridView ID="GridView2" runat="server" Width="539px">          </asp:GridView>          </div>      </form>  </body>  </html> |
| using System;  using System.Data;  using System.Collections.Generic;  using System.Data.Odbc;  using System.Linq;  using System.Web;  using System.Web.UI;  using System.Web.UI.WebControls;  using System.Windows.Forms;  namespace bd\_8  {      public partial class Request2 : System.Web.UI.Page      {          protected void Page\_Load(object sender, EventArgs e)          {              Reload();          }          protected void Request\_2()          {              // Создаем объект подключения              OdbcConnection conn = new OdbcConnection();              // Задаем параметр подключения – имя ODBC-источника              conn.ConnectionString = "Dsn=PostgreSQL30;database=students;server=students.ami.nstu.ru;port=5432;uid=pmi-b6306;pwd=Jemlift5;sslmode=disable;readonly=0;protocol=7.4;fakeoidindex=0;showoidcolumn=0;rowversioning=0;showsystemtables=0;fetch=100;unknownsizes=0;maxvarcharsize=255;maxlongvarcharsize=8190;debug=0;commlog=0;usedeclarefetch=0;textaslongvarchar=1;unknownsaslongvarchar=0;boolsaschar=1;parse=0;lfconversion=1;updatablecursors=1;trueisminus1=0;bi=0;byteaaslongvarbinary=1;useserversideprepare=1;lowercaseidentifier=0;d6=-101;xaopt=1";              // Подключаемся к БД              conn.Open();              // Определяем строку с текстом запроса              string strSQL = @"  -- Просмотр                                  select \*                                  from pmib6306.spj1                                  where n\_post = ? and date\_post between ? and ?;                                ";                // Создаем объект запроса              OdbcCommand cmd = new OdbcCommand(strSQL, conn);              // Создаем первый параметр              OdbcParameter par1 = new OdbcParameter();              par1.ParameterName = "@vn\_post";              par1.OdbcType = OdbcType.Text;              par1.Value = DropDownList2.SelectedValue;              //Добавляем первый параметр в коллекцию              cmd.Parameters.Add(new OdbcParameter("@vn\_post", DropDownList2.SelectedValue));              cmd.Parameters.Add(new OdbcParameter("@vdate\_post", TextBox1.Text));              cmd.Parameters.Add(new OdbcParameter("@vdate\_post", TextBox2.Text));              // Объявляем объект транзакции              OdbcTransaction tx = null;              try              {                  // Начинаем транзакцию и извлекаем объект транзакции из объекта подключения.                  tx = conn.BeginTransaction();                  // Включаем объект SQL-команды в транзакцию                  cmd.Transaction = tx;                  // Выполняем SQL-команду и получаем количество обработанных записей                  OdbcDataReader res = cmd.ExecuteReader();                  DataTable result\_table = new DataTable();                  using (DataColumn fNameColumn = new DataColumn())                  {                      fNameColumn.DataType = System.Type.GetType("System.String");                      fNameColumn.ColumnName = "Номер поставки";                      result\_table.Columns.Add(fNameColumn);                  }                  using (DataColumn fNameColumn = new DataColumn())                  {                      fNameColumn.DataType = System.Type.GetType("System.String");                      fNameColumn.ColumnName = "Номер поставщика";                      result\_table.Columns.Add(fNameColumn);                  }                  using (DataColumn fNameColumn = new DataColumn())                  {                      fNameColumn.DataType = System.Type.GetType("System.String");                      fNameColumn.ColumnName = "Номер детали";                      result\_table.Columns.Add(fNameColumn);                  }                  using (DataColumn fNameColumn = new DataColumn())                  {                      fNameColumn.DataType = System.Type.GetType("System.String");                      fNameColumn.ColumnName = "Номер изделия";                      result\_table.Columns.Add(fNameColumn);                  }                  using (DataColumn fNameColumn = new DataColumn())                  {                      fNameColumn.DataType = System.Type.GetType("System.String");                      fNameColumn.ColumnName = "Количество";                      result\_table.Columns.Add(fNameColumn);                  }                  using (DataColumn fNameColumn = new DataColumn())                  {                      fNameColumn.DataType = System.Type.GetType("System.String");                      fNameColumn.ColumnName = "Дата поставки";                      result\_table.Columns.Add(fNameColumn);                  }                  using (DataColumn fNameColumn = new DataColumn())                  {                      fNameColumn.DataType = System.Type.GetType("System.String");                      fNameColumn.ColumnName = "Цена";                      result\_table.Columns.Add(fNameColumn);                  }                  while (res.Read())                  {                      DataRow new\_row;                      new\_row = result\_table.NewRow();                      new\_row["Номер поставки"] = res[0];                      new\_row["Номер поставщика"] = res[1];                      new\_row["Номер детали"] = res[2];                      new\_row["Номер изделия"] = res[3];                      new\_row["Количество"] = res[4];                      new\_row["Дата поставки"] = res[5];                      new\_row["Цена"] = res[6];                      result\_table.Rows.Add(new\_row);                  }                  // Подтверждаем транзакцию                  GridView2.DataSource = result\_table;                  GridView2.DataBind();                  tx.Commit();              }              catch (Exception ex)              {                  // При возникновении любой ошибки                  // Формируем сообщение об ошибке                  Label2.Text = ex.Message;                  // выполняем откат транзакции                  tx.Rollback();              }              //закрываем соединение              conn.Close();          }          protected void Request\_3()          {              // Создаем объект подключения              OdbcConnection conn = new OdbcConnection();              // Задаем параметр подключения – имя ODBC-источника              conn.ConnectionString = "Dsn=PostgreSQL30;database=students;server=students.ami.nstu.ru;port=5432;uid=pmi-b6306;pwd=Jemlift5;sslmode=disable;readonly=0;protocol=7.4;fakeoidindex=0;showoidcolumn=0;rowversioning=0;showsystemtables=0;fetch=100;unknownsizes=0;maxvarcharsize=255;maxlongvarcharsize=8190;debug=0;commlog=0;usedeclarefetch=0;textaslongvarchar=1;unknownsaslongvarchar=0;boolsaschar=1;parse=0;lfconversion=1;updatablecursors=1;trueisminus1=0;bi=0;byteaaslongvarbinary=1;useserversideprepare=1;lowercaseidentifier=0;d6=-101;xaopt=1";              // Подключаемся к БД              conn.Open();              // Определяем строку с текстом запроса              string strSQL = @"  -- Удвоение числа деталей                                  update pmib6306.spj1                                  set kol = kol \* 2                                  where n\_post = ? and date\_post between ? and ?;                                ";                // Создаем объект запроса              OdbcCommand cmd = new OdbcCommand(strSQL, conn);              // Создаем первый параметр              OdbcParameter par1 = new OdbcParameter();              par1.ParameterName = "@vn\_post";              par1.OdbcType = OdbcType.Text;              par1.Value = DropDownList2.SelectedValue;              //Добавляем первый параметр в коллекцию              cmd.Parameters.Add(new OdbcParameter("@vn\_post", DropDownList2.SelectedValue));              cmd.Parameters.Add(new OdbcParameter("@vdate\_post", TextBox1.Text));              cmd.Parameters.Add(new OdbcParameter("@vdate\_post", TextBox2.Text));              // Объявляем объект транзакции              OdbcTransaction tx = null;              try              {                  // Начинаем транзакцию и извлекаем объект транзакции из объекта подключения.                  tx = conn.BeginTransaction();                  // Включаем объект SQL-команды в транзакцию                  cmd.Transaction = tx;                  // Выполняем SQL-команду и получаем количество обработанных записей                  OdbcDataReader res = cmd.ExecuteReader();                  tx.Commit();              }              catch (Exception ex)              {                  // При возникновении любой ошибки                  // Формируем сообщение об ошибке                  Label2.Text = ex.Message;                  // выполняем откат транзакции                  tx.Rollback();              }              //закрываем соединение              conn.Close();          }          protected void Request\_4()          {              // Создаем объект подключения              OdbcConnection conn = new OdbcConnection();              // Задаем параметр подключения – имя ODBC-источника              conn.ConnectionString = "Dsn=PostgreSQL30;database=students;server=students.ami.nstu.ru;port=5432;uid=pmi-b6306;pwd=Jemlift5;sslmode=disable;readonly=0;protocol=7.4;fakeoidindex=0;showoidcolumn=0;rowversioning=0;showsystemtables=0;fetch=100;unknownsizes=0;maxvarcharsize=255;maxlongvarcharsize=8190;debug=0;commlog=0;usedeclarefetch=0;textaslongvarchar=1;unknownsaslongvarchar=0;boolsaschar=1;parse=0;lfconversion=1;updatablecursors=1;trueisminus1=0;bi=0;byteaaslongvarbinary=1;useserversideprepare=1;lowercaseidentifier=0;d6=-101;xaopt=1";              // Подключаемся к БД              conn.Open();              // Определяем строку с текстом запроса              string strSQL = @"  -- Удвоение числа деталей                                  update pmib6306.spj1                                  set kol = kol / 2                                  where n\_post = ? and date\_post between ? and ?;                                ";                // Создаем объект запроса              OdbcCommand cmd = new OdbcCommand(strSQL, conn);              // Создаем первый параметр              OdbcParameter par1 = new OdbcParameter();              par1.ParameterName = "@vn\_post";              par1.OdbcType = OdbcType.Text;              par1.Value = DropDownList2.SelectedValue;              //Добавляем первый параметр в коллекцию              cmd.Parameters.Add(new OdbcParameter("@vn\_post", DropDownList2.SelectedValue));              cmd.Parameters.Add(new OdbcParameter("@vdate\_post", TextBox1.Text));              cmd.Parameters.Add(new OdbcParameter("@vdate\_post", TextBox2.Text));              // Объявляем объект транзакции              OdbcTransaction tx = null;              try              {                  // Начинаем транзакцию и извлекаем объект транзакции из объекта подключения.                  tx = conn.BeginTransaction();                  // Включаем объект SQL-команды в транзакцию                  cmd.Transaction = tx;                  // Выполняем SQL-команду и получаем количество обработанных записей                  OdbcDataReader res = cmd.ExecuteReader();                  tx.Commit();              }              catch (Exception ex)              {                  // При возникновении любой ошибки                  // Формируем сообщение об ошибке                  Label2.Text = ex.Message;                  // выполняем откат транзакции                  tx.Rollback();              }              //закрываем соединение              conn.Close();          }          protected void GridView2\_SelectedIndexChanged(object sender, EventArgs e)          {              // Создаем объект подключения              OdbcConnection conn = new OdbcConnection();              // Задаем параметр подключения – имя ODBC-источника              conn.ConnectionString = "Dsn=PostgreSQL30;database=students;server=students.ami.nstu.ru;port=5432;uid=pmi-b6306;pwd=Jemlift5;sslmode=disable;readonly=0;protocol=7.4;fakeoidindex=0;showoidcolumn=0;rowversioning=0;showsystemtables=0;fetch=100;unknownsizes=0;maxvarcharsize=255;maxlongvarcharsize=8190;debug=0;commlog=0;usedeclarefetch=0;textaslongvarchar=1;unknownsaslongvarchar=0;boolsaschar=1;parse=0;lfconversion=1;updatablecursors=1;trueisminus1=0;bi=0;byteaaslongvarbinary=1;useserversideprepare=1;lowercaseidentifier=0;d6=-101;xaopt=1";              // Подключаемся к БД              conn.Open();              // Определяем строку с текстом запроса              //string strSQL = "UPDATE pmib6306.p SET name = ?  where  n\_det = ? ";              string strSQL = "select \* from pmib6306.spj1";              // Создаем объект запроса              OdbcCommand cmd = new OdbcCommand(strSQL, conn);              // Создаем первый параметр              OdbcParameter par\_name = new OdbcParameter();              par\_name.ParameterName = "@vname";              par\_name.OdbcType = OdbcType.Text;              par\_name.Value = "Кулер";              // Добавляем первый параметр в коллекцию              // cmd.Parameters.Add(par\_name);              // Создаем второй параметр              OdbcParameter par\_town = new OdbcParameter();              par\_town.ParameterName = "@vn\_det";              par\_town.OdbcType = OdbcType.Text;              par\_town.Value = "P3";              // Добавляем второй параметр в коллекцию.              //  cmd.Parameters.Add(par\_town);              // Объявляем объект транзакции              OdbcTransaction tx = null;              try              {                  // Начинаем транзакцию и извлекаем объект транзакции из объекта подключения.                  tx = conn.BeginTransaction();                  // Включаем объект SQL-команды в транзакцию                  cmd.Transaction = tx;                  // Выполняем SQL-команду и получаем количество обработанных записей                  int i = cmd.ExecuteNonQuery();                  // Подтверждаем транзакцию                  tx.Commit();              }              catch (Exception ex)              {                  // При возникновении любой ошибки                  // Формируем сообщение об ошибке                  Label2.Text = ex.Message;                  // выполняем откат транзакции                  tx.Rollback();              }              //закрываем соединение              conn.Close();          }          protected void FillDropDownList2()          {              // Создаем объект подключения              OdbcConnection conn = new OdbcConnection();              // Задаем параметр подключения – имя ODBC-источника              conn.ConnectionString = "Dsn=PostgreSQL30;database=students;server=students.ami.nstu.ru;port=5432;uid=pmi-b6306;pwd=Jemlift5;sslmode=disable;readonly=0;protocol=7.4;fakeoidindex=0;showoidcolumn=0;rowversioning=0;showsystemtables=0;fetch=100;unknownsizes=0;maxvarcharsize=255;maxlongvarcharsize=8190;debug=0;commlog=0;usedeclarefetch=0;textaslongvarchar=1;unknownsaslongvarchar=0;boolsaschar=1;parse=0;lfconversion=1;updatablecursors=1;trueisminus1=0;bi=0;byteaaslongvarbinary=1;useserversideprepare=1;lowercaseidentifier=0;d6=-101;xaopt=1";              // Подключаемся к БД              conn.Open();              string strSQL = "select n\_post from pmib6306.s";              // Создаем объект запроса              OdbcCommand cmd = new OdbcCommand(strSQL, conn);              OdbcTransaction tx = null;              try              {                  // Начинаем транзакцию и извлекаем объект транзакции из объекта подключения.                  tx = conn.BeginTransaction();                  // Включаем объект SQL-команды в транзакцию                  cmd.Transaction = tx;                  // Выполняем SQL-команду и получаем количество обработанных записей                  OdbcDataReader res = cmd.ExecuteReader();                    while (res.Read())                      DropDownList2.Items.Add(res[0].ToString());                  // Подтверждаем транзакцию                  tx.Commit();              }              catch (Exception ex)              {                  // При возникновении любой ошибки                  // Формируем сообщение об ошибке                  Label2.Text = ex.Message;                  // выполняем откат транзакции                  tx.Rollback();              }              //закрываем соединение              conn.Close();          }          protected void Button1\_Click(object sender, EventArgs e)          {              Request\_2();              Reload();          }          protected void Button2\_Click(object sender, EventArgs e)          {              Request\_3();              Reload();              Button1\_Click(sender, e);          }          protected void Button3\_Click(object sender, EventArgs e)          {              Request\_4();              Reload();              Button1\_Click(sender, e);          }          private void Reload()          {              var SelVal = DropDownList2.SelectedValue;              DropDownList2.Items.Clear();              FillDropDownList2();              DropDownList2.SelectedValue = SelVal;          }      }  } |