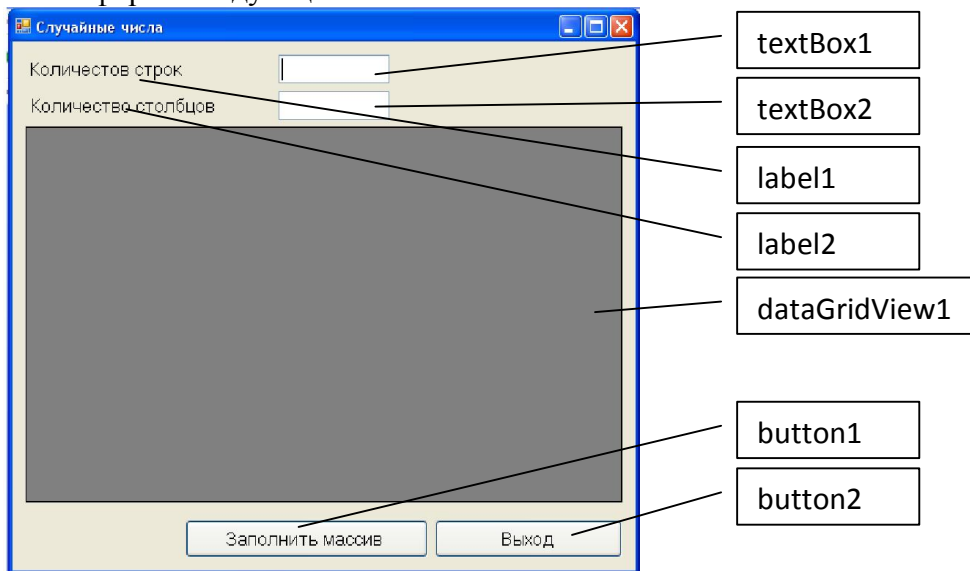


## Элемент управления DataGridView. Отображение и ввод массивов в PascalABC.NET

Элемент управления DataGridView является последней новинкой в серии табличных элементов DataGrid, позволяющих отображать таблицы. Главное назначение этих элементов - связывание с таблицами внешних источников данных, прежде всего с таблицами баз данных. Мы же сейчас рассмотрим другое его применение - в интерфейсе, позволяющем пользователю вводить и отображать матрицы - двумерные массивы.

### 1. Заполнение массива случайными числами.

Разместим на форме следующие компоненты:



Двойным щелчком по кнопке «Заполнить массив» создадим событие и запишем в него следующий код.

```

procedure Form1.button1_Click(sender: Object; e: EventArgs);
  var i,j,n,m:integer;
  var a:array[1..100,1..100] of integer;
begin
  n:=strtoint(textBox1.Text); // определяем число строк
  m:=strtoint(textBox2.Text); // определяем число столбцов
  // задаем размеры таблицы
  dataGridView1.RowCount :=n;
  dataGridView1.ColumnCount :=m;

  for i := 1 to n do
  begin
    for j:= 1 to m do
    begin
      a[i,j]:=PABCSystem.Random(21)-10;
      //Заполняем массив случайными числами
      dataGridView1[j - 1, i - 1].Value := inttostr(a[i,j]);
      // Переносим элементы массива в таблицу
    end;
  end;
end;

```

Далее на форме можно разместить еще один DataGridView и кнопку и решать задачу на заполнение массива случайными числами его обработку.

А у нас должно получиться следующее.

6	0	7	7
-10	10	-6	6
10	2	8	-7
-6	4	-3	-1
0	-4	8	-5
*	0	7	1

## 2. Заполнение массива с клавиатуры.

Разместим на форме следующие компоненты

Внимание! Все ячейки исходного массива должны быть заполнены

button1

textBox1

textBox2

label1

label2

dataGridView1

label3

button2

dataGridView2

button3

В этом примере DataGridView1 будет использоваться для заполнения массива с клавиатуры, а DataGridView2 для контроля, на этот компонент после можно вывести измененный (обработанный массив)

Опишем глобальные переменные, в первом примере их так же можно сделать глобальными.

```
var n,m:integer;
var a:array[1..100,1..100] of integer;
```

Двойным щелчком по кнопке «Создать массив» создадим событие и запишем в него следующий код.

```
procedure Form1.button1_Click(sender: Object; e: EventArgs);
begin
  n:=strtoint(textBox1.Text);
  m:=strtoint(textBox2.Text);

  dataGridView1.RowCount :=n;
```

```
datagridview1.ColumnCount :=m;
end;
```

Если запустить программу то у нас должна получиться таблица, которую можно заполнять с клавиатуры числами. В принципе не только числами, на ее основе можно писать базы данных.

Двойным щелчком по кнопке «Считать массив» создадим событие и запишем в него следующий код.

```
procedure Form1.button2_Click(sender: Object; e: EventArgs);
var s:string;
begin
// Готовим вторую таблицу, задаем число строк и столбцов
datagridview2.RowCount :=n;
datagridview2.ColumnCount :=m;

for var i := 1 to n do
begin
for var j:= 1 to m do
begin
// таким образом, преобразует тип Object в тип Integer;
// сначала к строковому типу
s:=datagridview1[j - 1, i - 1].Value.ToString;
a[i,j]:=strtoint(s);// затем к целому числу
end;
end;
// здесь может быть обработка массива...
for var i := 1 to n do
begin
for var j:= 1 to m do
begin
// затем из массива заполняем вторую таблицу
datagridview2[j - 1, i - 1].Value := inttostr(a[i,j]);
end;
end;
end;
```

Такое преобразование, на мой взгляд, более простое, я не лезу в дебри модуля .NET. А со строками можно работать при создании базы данных.

Должно получиться следующее.

