Министерство науки и высшего образования Российской Федерации

**Федеральное государственное бюджетное образовательное учреждение  
высшего образования**

**ИРКУТСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**

Институт информационных технологий и анализа данных

Кафедра информационных систем и технологий.

Наименование работы

**Использование функций**

**Отчет по лабораторной работе №6 вариант – 4**

по дисциплине

**«Программирование на языке высокого уровня»**

|  |  |  |  |
| --- | --- | --- | --- |
| Выполнил студент группы ИСТБ-19-2 |  |  | Козлов Максим Васильевич |
| подпись |  | И. О. Фамилия |
|  |  |  |  |
| Принял  Должность |  |  | Столбов Александр Борисович |
| подпись |  | И. О. Фамилия |

**Иркутск 2020**

Номер варианта работы:4.4.4.

* Выбор марки автомобиля из 5
* Любимые домашние животные из 6
* Количество упаковок лекарства и стоимость одной упаковки – Подсчет уплаченной суммы
* Растущая елка.
* Информация по экзаменам (название предмета, ФИО преподавателя, число пятерок, число четверок, число троек, число двоек, число не явившихся, общее число студентов группы – вычисляется)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Имя окна | Имя элемента контроля | Имя события | Словесное описание действия | Программный код |
| **Элементы управления** | RadioButton | Выбор марки автомобиля из 5.  Вывод сообщения посредством нажатия button1. | «Выберите марку автомобиля:»  Сообщение:  "Вы выбрали марку автомобиля: " + результат(ы) выбора пользователя | private void button1\_Click(object sender, EventArgs e)  {  {  string Col = "";  if (this.radioButton1.Checked)  Col = this.radioButton1.Text;  if (this.radioButton2.Checked)  Col = this.radioButton2.Text;  if (this.radioButton3.Checked)  Col = this.radioButton3.Text;  if (this.radioButton4.Checked)  Col = this.radioButton4.Text;  if (this.radioButton5.Checked)  Col = this.radioButton5.Text;  MessageBox.Show("Вы выбрали марку автомобиля: "+ Col);  }  } |
| CheckButton | Любимые домашние животные из 6.  Вывод сообщения посредством нажатия button2 | «Какие цветы ваши любимые?»  Сообщение: "Ваши любимые виды животных: "+ результат(ы) выбора пользователя | private void button2\_Click(object sender, EventArgs e)  {  string Col = "";  string Col1 = "";  string Col2 = "";  string Col3 = "";  string Col4 = "";  string Col5 = "";  if (this.checkBox1.Checked)  Col = this.checkBox1.Text + " ";  if (this.checkBox2.Checked)  Col1 = this.checkBox2.Text + " ";  if (this.checkBox3.Checked)  Col2 = this.checkBox3.Text + " ";  if (this.checkBox4.Checked)  Col3 = this.checkBox4.Text + " ";  if (this.checkBox5.Checked)  Col4 = this.checkBox5.Text + " ";  if (this.checkBox6.Checked)  Col5= this.checkBox6.Text + " ";  MessageBox.Show( "Ваши любимые виды животных: " + Col + Col1 + Col2 + Col3 + Col4 + Col5);  } |
| maskedTextBox | Ввод количества упаковок лекарства и стоимости одной упаковки пользователем с помощью maskedTextBox и последующий подсчет уплаченной суммы посредством нажатия кнопки button3 | Расчёт уплаченной с учётом стоимости 1 упаковки и количеством упаковок | private void button3\_Click(object sender, EventArgs e)  {  bool IsDigit1 = Regex.IsMatch(maskedTextBox1.Text, @"^[0-9]{1,8}[,]{0,1}[0-9]{0,8}$");  bool IsDigit2 = maskedTextBox2.Text.All(char.IsDigit);  if ((IsDigit1 == true)&&(IsDigit2==true))  {  if (!String.IsNullOrEmpty(maskedTextBox1.Text)  || !String.IsNullOrWhiteSpace(maskedTextBox1.Text)  || !String.IsNullOrEmpty(maskedTextBox2.Text)  || !String.IsNullOrWhiteSpace(maskedTextBox2.Text))  {  float cost = float.Parse(maskedTextBox1.Text);  float number = float.Parse(maskedTextBox2.Text);  float buy = cost \* number;  textBox1.Text = (Convert.ToString(buy));  }  }  else  {  MessageBox.Show("Введите корректные данные");  }  } |
| **Вид рисунка** | Form2 | Функция DrawTree, рисующая растущую ёлку | Функция, рисующая растущую ёлку, на сером фоне. | float x0 = 300;  float y0 = 320;  float t = 0;  int xRec = 295;  int yRec = 320;  int wRec = 10;  int hRec = 0;  void DrawTree(Graphics g, float x0, float y0, float t)  {  Pen triangle = new Pen(Color.Green, 3); // заготавливаем карандаш для ёлки  SolidBrush b1 = new SolidBrush(Color.Green);// закрашиваем ёлку зелёным цветом  Rectangle r1 = new Rectangle(xRec- Convert.ToInt32(0.1\*t), yRec- Convert.ToInt32(t), wRec+ Convert.ToInt32(0.2 \* t), hRec+ Convert.ToInt32(0.9\*t));// данные роста ствола дерева  Rectangle r2 = new Rectangle(0, 300, 818, 177);// данные для отрисовки снега  g.FillRectangle(new SolidBrush(Color.White), r2);// закрашиваем снег белым цветом  g.DrawRectangle(new Pen(Color.Brown), r1);// рисуем ствол дерева по данным r1  g.FillRectangle(new SolidBrush(Color.Brown), r1);// закрашиваем ствол дерева по данным r1  PointF[] tr1 = new PointF[] { new PointF(x0 - Convert.ToInt32(1.6 \* t), y0 - t), new PointF(x0, y0 - Convert.ToInt32(t\*2)), new PointF(x0 + Convert.ToInt32(1.6 \* t), y0 - t) };// данные роста первой части ёлки  PointF[] tr2 = new PointF[] { new PointF(x0 - Convert.ToInt32(1.3 \* t), y0 - Convert.ToInt32(t \* 2)), new PointF(x0, y0 - Convert.ToInt32(t \* 3)), new PointF(x0 + Convert.ToInt32(1.3 \* t),y0- Convert.ToInt32(t \* 2)) };// данные роста второй части ёлки  PointF[] tr3 = new PointF[] { new PointF(x0 - Convert.ToInt32(1.0 \* t), y0 - Convert.ToInt32(t \* 3)), new PointF(x0, y0 - Convert.ToInt32(t \* 4)), new PointF(x0 + Convert.ToInt32(1.0 \* t), y0 - Convert.ToInt32(t \* 3)) };// данные роста третьей части ёлки  PointF[] tr4 = new PointF[] { new PointF(x0 - Convert.ToInt32(0.7 \* t), y0 - Convert.ToInt32(t \* 4)), new PointF(x0, y0 - Convert.ToInt32(t \* 5)), new PointF(x0 + Convert.ToInt32(0.7 \* t), y0 - Convert.ToInt32(t \* 4)) };// данные роста четвёртой части ёлки  g.DrawPolygon(triangle, tr1);// отрисовка первой части ёлки по tr1  g.FillPolygon(b1, tr1);// закрашивание первой части ёлки по tr1  g.DrawPolygon(triangle, tr2);// отрисовка первой части ёлки по tr2  g.FillPolygon(b1, tr2);// закрашивание первой части ёлки по tr2  g.DrawPolygon(triangle, tr3);// отрисовка первой части ёлки по tr3  g.FillPolygon(b1, tr3);// закрашивание первой части ёлки по tr3  g.DrawPolygon(triangle, tr4);// отрисовка первой части ёлки по tr1=4  g.FillPolygon(b1, tr4);// закрашивание первой части ёлки по tr4  } |
| Взаимодействие Form2 и функции DrawTree | Нанесение картинки на Form2 | private void Form2\_Paint(object sender, PaintEventArgs e)  {  this.BackColor = Color.FromArgb(111, 121, 131);// Цвет заднего фона Form 2  Graphics g = e.Graphics;  DrawTree(g,x0,y0,t);// рисуем ёлку  } |
| Timer, с помощью которого идет анимирование картинки | Анимация роста ёлки | int checkTime = 0; // счётчик, посредством которого определяется время работы таймера  private void timer1\_Tick(object sender, EventArgs e)  {  if (checkTime < 50)  {  t++;  checkTime++;  Invalidate();// вызывает перерисовку Form 2  }  } |
| Клик по кнопке button1 вызывает анимацию | Вызов анимации | private void button1\_Click(object sender, EventArgs e)  {  if (checkTime < 50)  {  Timer timer = new Timer(); // при нажатии button1 задаём новый таймер  timer.Enabled = true;// активируем таймер  timer.Interval = 1000;// задаём интервал в 1 сек  timer.Tick += new EventHandler(timer1\_Tick);// вызываем событие timer1\_Tick  }  else  {  timer1.Stop();// при срабатывании условия отключаем таймер  MessageBox.Show("Ёлка выросла!");// выводим сообщение об окончании работы программы  }  } |
| **Структура таблицы** | dataGridView1 | Таблица, заполняемая вводимой пользователем информацией и подсчётом на основе этой  информации из строк textbox и maskedtextbox,посредством нажатия button1 | Информация по экзаменам (название предмета, ФИО преподавателя, число пятерок, число четверок, число троек, число двоек, число не явившихся, общее число студентов группы – вычисляется)у | private void button1\_Click(object sender, EventArgs e)  {  string str = maskedTextBox6.Text;  string s1 = " ";  string s2 = ",";  string s3 = ".";  str = str.Replace(s2, s3);// замеяем запятые на точки  str = str.Replace(s1,"");// удаляем пробелы  bool SubjectName = Regex.IsMatch(textBox1.Text,@"[А-Яа-я]");// создаём шаблон для проверки введённой строки textBox1  bool TeacherName = Regex.IsMatch(str, @"^[А-Я]{1}[.]{1}[А-Я]{1}[.]{1}[А-Я]{1}[а-я]{2,10}$");// создаём шаблон для проверки введённой строки maskedTextBox6  if (textBox1.Text == "" || maskedTextBox6.Text == "" || maskedTextBox1.Text == " "  || maskedTextBox2.Text == " "  || maskedTextBox3.Text == " "  || maskedTextBox4.Text == " "  || maskedTextBox5.Text == " ")// проверка на пустые строки  {  MessageBox.Show("Введите все поля соотвествующими данными");  }  else  {  string sum = Convert.ToString(Convert.ToInt32(maskedTextBox1.Text) + Convert.ToInt32(maskedTextBox2.Text) +  Convert.ToInt32(maskedTextBox3.Text) + Convert.ToInt32(maskedTextBox4.Text) + Convert.ToInt32(maskedTextBox5.Text));// нахождение общего количества студентов  if (SubjectName == true && TeacherName == true)  {  dataGridView1.Rows.Add(textBox1.Text, maskedTextBox6.Text, Convert.ToInt32(maskedTextBox1.Text), Convert.ToInt32(maskedTextBox2.Text),  Convert.ToInt32(maskedTextBox3.Text), Convert.ToInt32(maskedTextBox4.Text), Convert.ToInt32(maskedTextBox5.Text),sum);// ввод в dataGridView1  }  else  {  MessageBox.Show("ФИО преподавателя или название предмета введено неправильно!");  }  } |