

Визуал програмчлал

Ғ.ІТ203

Лекц-6

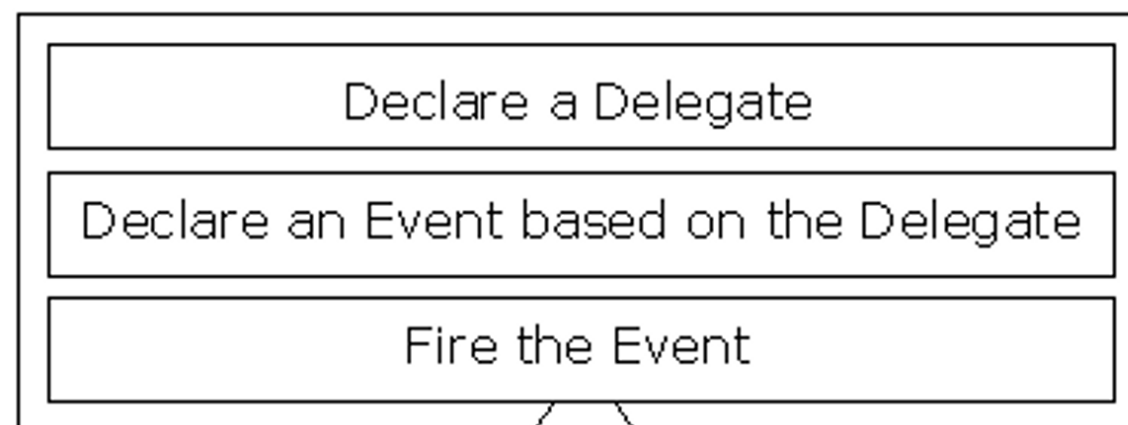
Гарчиг

- Хэрэглэгчийн контрол үүсгэх нь
 - Бэлтгэл ажил
 - Контрол үүсгэх
 - Атрибут
 - Кодчиллол
- Хэрэглэгчийн оролтыг шалгах

User Control - Хэрэглэгчийн контрол

- Стандарт контролуудыг сайжруулж, баяжуулсан хувилбар болох хэрэглэгчийн тодорхойлсон контролыг үүсгэх боломжтой.
- User controls нь System.Windows.Forms.UserControl-оос удамшсан.
- Хэрэглэгч нэгээс олон контролуудыг нэгтгэж дараа дахин ашиглах боломжтойгоор зохион байгуулж болно.
 - Жишээ нь: Хаягын мэдээллийг оруулах label, combobox, textbox-уудын нэгдэл
 - Login form-ийг контрол маягаар үүсгэх

User Control (Publisher)

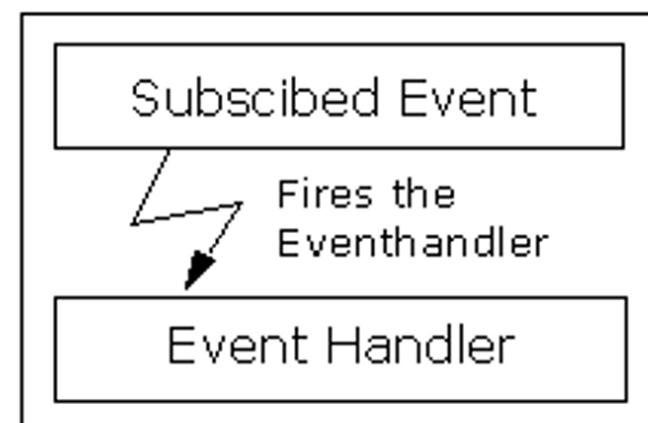


Notify
subscribed Events

Application 1 (Subscriber)



Application 2 (Subscriber)



Хэрэгжүүлэлт

Visual C#

Get Started

Windows Desktop

.NET Standard

Test









▸ Visual Basic

▸ Other Project Types

▸ Online

Not finding what you are looking for?

[Open Visual Studio Installer](#)

	Console App (.NET Framework)	Visual C#
	Class Library (.NET Framework)	Visual C#
	Windows Service (.NET Framework)	Visual C#
	Empty Project (.NET Framework)	Visual C#
	WPF Browser App (.NET Framework)	Visual C#
	WPF Custom Control Library (.NET Framework)	Visual C#
	WPF User Control Library (.NET Framework)	Visual C#
	Windows Forms Control Library (.NET Framework)	Visual C#

A project for creating control
Windows Forms applications

Name:	WindowsFormsControlLibrary1	
Location:	C:\Users\Dolgorsuren\source\repos	<input data-bbox="1479 1118 1649 1160" type="button" value="Browse..."/>
Solution name:	WindowsFormsControlLibrary1	
Framework:	.NET Framework 4.6.1	<input checked="" type="checkbox"/> Create directory for solution <input type="checkbox"/> Create new Git repository
		<input data-bbox="1663 1279 1818 1322" type="button" value="OK"/>

Хэрэгжүүлэлт

- UserControl-ийн контролуудаа өрж дизайн, кодчиллоо хийнэ.
- Дууссан гэж үзвэл Run хийж .dll үүсгээд бусад прожектоо оруулж ашиглахад бэлэн болно.
- Үүссэн хэрэглэгчийн контролоо прожектдоо оруулахдаа эхлээд toolbox-доо нэмэх шаардлагатай.
- Toolbox - баруун товчоо дараад Choose Items ... сонгоно.
 - .NET Framework Components-оос сонго.
 - Browse ... /.dll – ээ сонго/

Хэрэглэгчийн оролтыг шалгах

1. Хэрэглэгчийн оролтыг хориглох
2. Өгөгдлийн талбарыг шалгах
3. Форм өгөгдлийг шалгах
4. Жагсаалт хэлбэрийн контролууд

Хэрэглэгчийн оролтыг хориглох

- Хэрэглэгчийн оролтыг шалгах зарчим
- Мэдрэмжт шалгалт
- Textbox тодорхойломжуудыг хэрэглэх
- MaskedTextBox контролыг ашиглах
- Regular Expression ашиглах

Хэрэглэгчийн оролтыг шалгах үндэс

- Хэрэглэгчийг буруу өгөгдөл оруулахаас сэргийлэх
- Хэрэглэгчээр зөв өгөгдөл оруулахыг шаардах
- Хэрэглэгчийг зөв өгөгдөл оруулах заавар гаргах
- Програмын загварыг гаргахдаа шалгах нөхцлүүдийг тодорхойлох
- Таны програмын шаардлагаас хамааран тодорхой хэсэгт шалгах код суурилуулах

Мэдрэмжит шалгалт

- Хэрэглэгчийн оролтыг шалгах, хориглох урьдчилан тодорхойлогдсон контролууд
- Түгээмэл хэрэглэгддэг контролууд өөртөө мэдрэмжит шалгалтын хэсэгтэй байдаг.

Контрол	Шалгах техник
RadioButton	Зөвхөн идэвхтэй эсвэл идэвхгүй
Checkbox	Сонгогдсон эсвэл сонгогдоогүй
CheckedListBox	Сонгох хэсгүүдээс тогтсон жагсаалт
ListBox	Зөв элементүүдээс тогтсон жагсаалтаас сонгуулах
DateTimePicker	Зөвхөн огноо эсвэл цаг сонгуулах
MonthCalendar	Календар мэдээлэл сонгох
Textbox	Хориглох эсвэл өгөгдөл засварлах

Тоон өгөгдлийг форматлах

- String.Format метод нь огноо болон тоон өгөгдлийг дэлгэцэнд дүрслэхэд зориулж форматлахад ашиглагдана.

String s = String.Format("{0}-н квадрат язгуур нь {1}", 64, 8);

- Хэв нь угалзан хаалтанд бичигдсэн индекс, өргөн, хувиргагчаас тогтоно.

- Индекс нь ямар аргумент дээр тухайн хэв хэрэглэгдэхийг
- Индексийн арын бүхэл тоон утга нь хэвлэгдэх утгын өргөнийг
- Хувиргагч нь ямар төрлөөр заасан аргументыг форматлахыг заах ба хэрэглэхгүй байж болно.

String f=String.Format("Шалгалтанд {0} оюутан {1,2:P} тай тэнцэв.", 5, .75)

“Шалгалтанд 5 оюутан 75.00% тай тэнцэв.”

Тоон өгөгдлийг форматлах

Хувиргагч	Тайлбар	Хэвийн жишээ	Гаралт
C, c	Мөнгөн утга. Тоон утгыг мөнгөн утгаар дүрсэлнэ. Нарийвчлалд бутархай орны тоог зааж болно.	{0:C2}, 1458.75	\$1,458.75
D, d	Бутархай. Том тоон утга. Нарийвчлал нь үзэгдэх тэмдэгтийн тоог заах ба зүүн талд нь шаардлагатай хэмжээгээр 0-үүдийг нэмдэг.	{0:D5}, 455	00455
E, e	Зэрэгт, Тоон утгыг e зэрэгт хэлбэрт дүрсэлнэ.	{0,10:E2}, 3298.78	3.30+E003
F, f	Хөвөгч цэгтэй. Тоон утгыг ddd.ddd хэвэнд оруулна.	{0,10:F0},162.57	162
G, g	Ерөнхий. Тоон утгаас хамаараад зэрэгт эсвэл бутархай тоон хэлбэрт дүрслэгдэнэ.	{0,10:G}, 0.0000099	9.9E-06
N, n	Тоо. Мянганы таслалтай дүрслэл рүү хувиргана.	{0,10:N}, 1045.78	1,045.78
P, p	Зууны хувь. Зууны хувь хэлбэрт дүрслэх ба нарийвчлал нь бутархай орны тоог заана.	{0,10:P}, 0.78	78.00%
R, r	Раунд-трип. Бутархай орны цифрүүдийг байгаагаар нь дүрсэлдэг	{0,10:R}, 1.62736	1.62736
X, x	16-тын утга. Тоон утгыг 16-тын дүрслэлд хувиргана.	{0,10:X4}, 25	0019

Огноо өгөгдлийг форматлах

- **Жишээ**

```
DateTime curDate = DateTime.Now;
```

```
Console.WriteLine("Date: {0:d} ", curDate); //1/19/2004
```

- Хэрэв дээрх стандарт хэвүүд нь таны хэрэгцээнд таарахгүй бол өөрийн хэвийг зохиож болно.

- **Жишээ**

```
DateTime curDate = DateTime.Now;
```

```
string f = currDate.ToString("dd MMM yyyy") //19 Jan 2004
```

Огноо өгөгдлийг форматлах

Хувиргагч	Тайлбар	Хэвийн жишээ
d	Товчилсон огноо	1/19/2004
D	Бүтэн огноо	Monday, January 19, 2004
f	Бүтэн огноо /товчилсон цаг/	Monday, January 19, 2004 4:05 PM
F	Бүтэн огноо /бүтэн цаг/	Monday, January 19, 2004 4:05:02 PM
g	Ерөнхий огноо /товчилсон цаг/	1/19/2004 4:05 PM
G	Ерөнхий огноо /бүтэн цаг/	1/19/2004 4:05:20 PM
M, m	Сар-өдөр	January 19
Y, y	Жил-сар	January, 2004
t	Товчилсон цаг	4:05 PM
T	Бүтэн цаг	4:05:20 PM
s	Нийтийн огноо /ISO 8601/	2004-01-19T16:05:20
u	Нийтийн огноо	2004-01-19 16:05:20Z
U	Нийтийн огноо, нийтийн цаг	Monday, January 19, 2004 21:05:20 PM

Textbox-ийн шинж байдлыг ашиглах

- Та Textbox-ийн дараах шинж байдлыг ашиглаж хэрэглэгчийн оролтыг хориглох, завсарлах боломжтой.

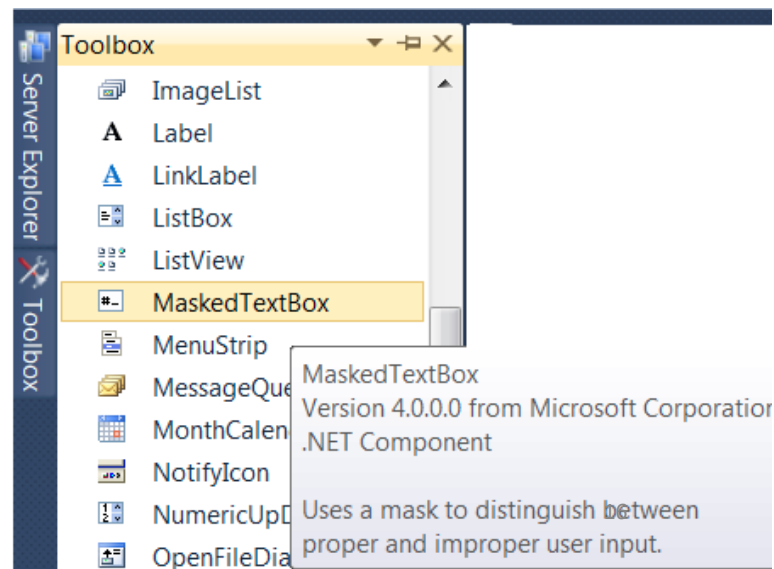
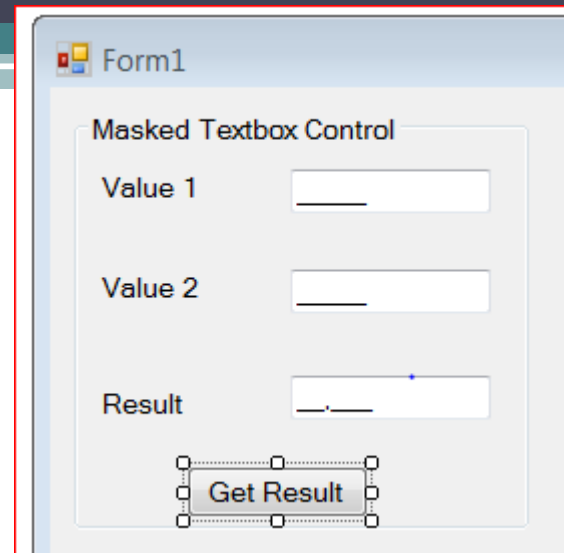
Шинж чанар	Шалгах техник
PasswordChar	Нуух эсвэл маскласан тэмдэгтүүдийг textbox- контролд оруулах
MaxLength	Оруулж болох хамгийн их тэмдэгт мөрийн урт
ReadOnly	Урьдчилан бэлдсэн зөв хариулт мэдээлэх бол
CharacerCasting	Оруулах тэмдэгтүүдийг жижиг болон том үсэг хэлбэрт шилжүүлнэ.

MaskedTextBox Control

- MaskedTextBox контролыг нэмнэ.
- MaskedTextBox контролыг формд нэмнэ.
- Mask шинж байдлыг засварлана.
 - SetMask
 - Properties → Mask
- Format шинж байдлыг засварлана.
- Хэрэглэгч өгөгдөл оруулна.

```
MaskedTextBox1.Mask = "0000000"
```

```
MaskedTextBox2.Mask = "0000000"
```



Mask тэмдэгтүүд

- 0 - Digit, required. Value between 0 and 9.
- 9 - Digit or space, optional.
- # - Digit or space, optional. If this position is blank in the mask, it will be rendered as a space in the Text property.
- L - Letter, required. Restricts input to the ASCII letters a-z and A-Z.
- ? - Letter, optional. Restricts input to the ASCII letters a-z and A-Z.
- & - Character, required.
- C - Character, optional. Any non-control character.
- A - Alphanumeric, required.
- a - Alphanumeric, optional.
- . - Decimal placeholder.
- , - Thousands placeholder.
- : - Time separator.
- / - Date separator.
- \$ - Currency symbol.
- < - Shift down. Converts all characters that follow to lowercase.
- > - Shift up. Converts all characters that follow to uppercase.
- | - Disable a previous shift up or shift down.
- \ - Escape. Escapes a mask character, turning it into a literal. "\\\" is the escape sequence for a backslash.

Өгөгдлийн талбар шалгах

- Boolean функц хэрэглэх
- ErrorProvider компонентыг ашиглах
- Контролыг идэвхжүүлэх
- Хэрэглэгчийн оролтыг засварлах
- Үйл явцыг ашиглах шалгалт хийх

Boolean функц хэрэглэх

- Түгээмэл Boolean функц

Функц	Тайлбар
IsNumeric	Оруулсан илэрхийлэл нь тоо эсэхээс хамааран буль утга буцаана.
IsDate	Оруулсан илэрхийлэл огноо эсэхээс хамааран буль утга буцаана.

- Жишээ

```
var text = "I am not a number";  
text.IsNumeric() //<--- returns false
```

ErrorProvider компонентыг ашиглах

- ErrorProvider компонентыг формд тавина
 - ToolBox-ын Windows Forms хэсэгт байгаа болно.
- SetError гишүүн функцийг дуудна
 - Эхний параметр нь алдаа илрэх контролыг төлөөлдөг бол хоёрдох параметр нь дүрслэх алдааны мэдээллийг заадаг

- **Жишээ**

ErrorProvider1.SetError(Textbox1, "Please enter a valid date.");



Контролыг идэвхжүүлэх

- Фокус гэж юу вэ?
 - Контрол фокустай болох үед хэрэглэгч гар болон хулганы тусламжтайгаар өгөгдөл оруулах боломжтой болно.
 - Хэрэглэгч буруу мэдээлэл оруулвал та алдаа засагдтал фокусыг тухайн контролд олгох хэрэгтэй.
- Жишээ
 - TextBox контролыг идэвхжүүлэхдээ Focus гишүүн функцыг ашиглана.
TextBox1.Focus();
 - SelectAll функцыг ашиглан буруу мэдээллийг сонгох
TextBox1.SelectAll();

Хэрэглэгчийн оролтыг засварлах

- Дараах функцүүдээр хэрэглэгчийн оруулсан өгөгдлийг засварлаж болно.

Функц	Тайлбар
Ucase	Харгалзах тэмдэгт мөрийг том үсэгт хөрвүүлнэ.
Lcase	Харгалзах тэмдэгт мөрийг жижиг үсэгт хөрвүүлнэ.
Trim	Харгалзах тэмдэгт мөрийн эхэнд болон төгсгөлд буй хоосон зайг устгана.

- **Жишээ**

```
String LowerCase = "Hello World 1234";
```

```
UpperCase = UCase(LowerCase);
```

```
//Returns "HELLO WORLD 1234"
```

Үйл явцыг ашиглан шалгах

- Validating үйл явц

```
private void textBox1_Validating(object sender,
                                System.ComponentModel.CancelEventArgs e)
{
    string errorMsg;
    if(!ValidEmailAddress(textBox1.Text, out errorMsg))
    {
        // Cancel the event and select the text to be corrected by the user.
        e.Cancel = true;
        textBox1.Select(0, textBox1.Text.Length);

        // Set the ErrorProvider error with the text to display.
        this.errorProvider1.SetError(textBox1, errorMsg);
    }
}
```

- Validated event

```
private void textBox1_Validated(object sender, System.EventArgs e)
{
    // If all conditions have been met, clear the ErrorProvider of errors.
    errorProvider1.SetError(textBox1, "");
}
```

Формын олон талбарт шалгалт хийх

- Хэрэглэгчийг визуалиар хориглох
 - **Жишээ нь:** Хэрэглэгч харгалзах бүх талбарт өгөгдөл оруулж дуустал товчийг **OK** идэвхигүй болгох
- Формын бүх талбарыг нэг агшинд шалгах
 - **Жишээ нь:** Бүх шалгалтын кодыг **OK** товчийн **Click** үйл явцын ард

Regular илэрхийлэл - Regex класс

- Regex классыг регуляр илэрхийллийг үнэлж, түүнтэй таарах утгыг өгөгдсөн мөр дээр самнаж, боловсруулалт хийдэг.
- Энэ класс нь текст хайлт, задлан ялгал, солилт хийхэд зориулсан статик болон объект методуудаар хангадаг.

```
using System.Text;  
using System.Text.RegularExpressions;
```

- Бичгийн дүрэм

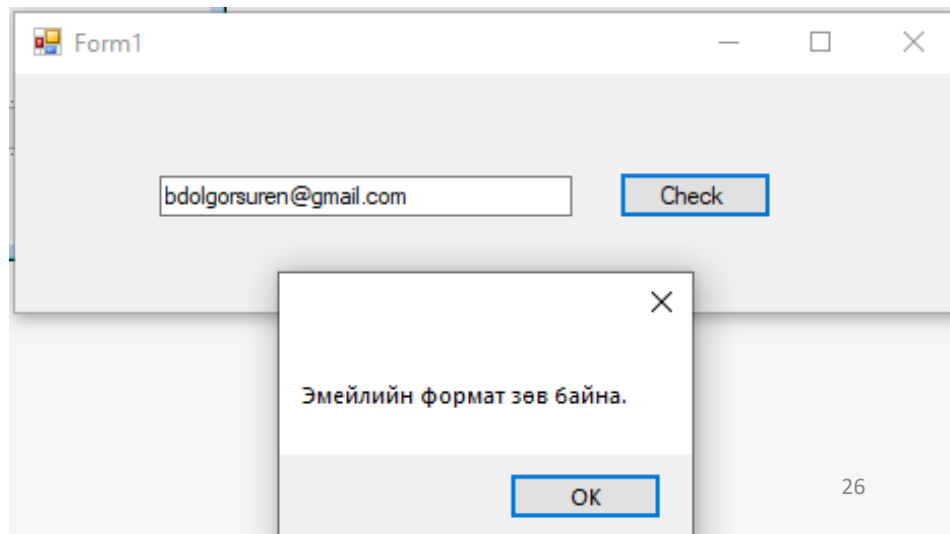
Regex (string pattern)

Regex (string pattern, RegexOptions)

RegexOption
CultureInvariant
IgnoreCase
RightToLeft

Жишээ

```
private void button1_Click(object sender, EventArgs e){  
    Regex RX = new Regex("^[a-zA-Z0-9]{1,20}@[a-zA-Z0-9]{1,20}.[a-zA-Z]{2,3}$");  
    if (!RX.IsMatch(textBox1.Text))  
    {  
        MessageBox.Show("Эмейлийн формат буруу байна.");  
    }  
    else  
    {  
        MessageBox.Show("Эмейлийн формат зөв байна.");  
    }  
}
```



Regex method-ууд

- Regex класс нь текст хайлт болон засварлалт хийхэд зориулагдсан
 - `isMatch`
 - `Replace`
 - `Split`
 - `Match`
 - `Matches` гэх мэт методуудтай.

```
string myStr = textBox1.Text;  
string newStr = Regex.Replace(myStr, "a", "o");
```

Regex – Илэрц таарах байрлалыг заах ТЭМДЭГТҮҮД

Тэмдэгт	Тайлбар
^	Илэрц текстийн эхэнд таарна.
\$	Илэрц текстийн төгсгөлд таарна.
\A	Өмнөх хэвтэй тохирох илэрц мөрийн эхэнд олдох
\b \B	Үгэн (үсэг, тооноос бүтсэн) хүрээнд байрлуулах (\b), үгэн бус хүрээнд байрлуулах (\B)
\z \Z	Илэрц мөрийн төгсгөлд таарна (\z), мөрийн төгсгөлийн тэмдэгтийн өмнө

```
String car = "УБ3-11-21";  
String carPatt = @"^([A-Я]{3})-(\d{2})-(\d{2})$";  
Match carMatch = Regex.Match(car,carPatt);
```

Анхаарал тавьсанд баярлалаа.