

# Визуал програмчлал

Ү.ИТ203

Лекц-4

# Агуулга

- Form –ийн бүрэлдэхүүн хэсэг
  - Контролуудын төрөл
  - Контролуудын төрлөөс хамааран групп контролуудыг ашиглах
- Нэг Form-оос нөгөө Form-ыг дуудах
- Харилцах цонхтой ажиллах

# Контролуудын төрөл

Контролын төрөл	Тайлбар
Тексттэй ажиллах	Хэрэглэгч өгөгдөл оруулах болон өөрчлөх, Хэрэглэгчид мэдээллийг харуулах
Сонголт	Жагсаалтаас сонгох
Graphics	Дүрстэй ажиллах
RichText контролууд	Тусгай зориулалттай
Value setting	Өгөгдлүүдийг тохируулах

# Контролуудын төрөл

Контролын төрөл	Тайлбар
Commands	Програмтай ажиллаж байгаа хэрэглэгчийн үйл ажиллагааг харуулах
Харилцах цонхны	Програмын үндсэн үйлдлүүдийг тодорхойлох контролууд. Жишээ нь: Save, Open, Print
Menu controls	Меню-тай ажиллах
Grouping	Контруулын групп үүсгэх
Print support controls	Програмын үйл ажиллагаатай холбоотой мэдээллийг хэвлэх

# Form-ын бүрэлдэхүүн хэсэг

Search

Around The World Travels

Search Options

☐ By Travel Type ☐ By Continents ☐ All Packages

Travel Type

Continents

View Information Options

☐ Picture ☐ Text

View

RadioButton

GroupBox

Panel

CheckBox

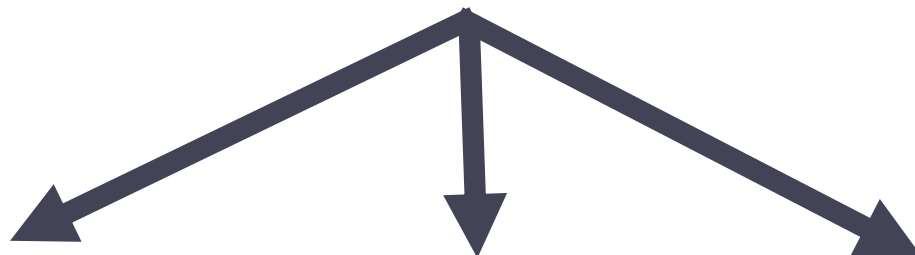
# Тексттэй ажиллах контролууд

Текст өөрчлөх



TextBox

Текст харуулах



Label

LinkLabel

StatusBar

# Сонголтын контролууд

- ListBox
- ComboBox
- CheckedListBox
- ListView
- TreeView
- NumericUpDown
- DomainUpDown

# Toolbar

- Бүх програмуудад байдаг toolbar цэсны үйл ажиллагаануудыг тохируулах.
- Шаардлагатай тохиолдолд товчнуудын байрлалыг солих боломжтой байна.



A sample toolbar from a word document



# Өгөгдөл тохируулах контролууд

- CheckBox
- CheckedListBox
- RadioButton
- TrackBar

# Групп контролууд

- Form
- Panel
- GroupBox
- Tab Control

# RichText контролууд

- RichTextBox
- DateTimePicker
- MonthCalender



# Нэмэлт контролууд

- Timer
- Scrollbar
- ProgressBar
- ToolTip

# Form –уудын хооронд шилжих

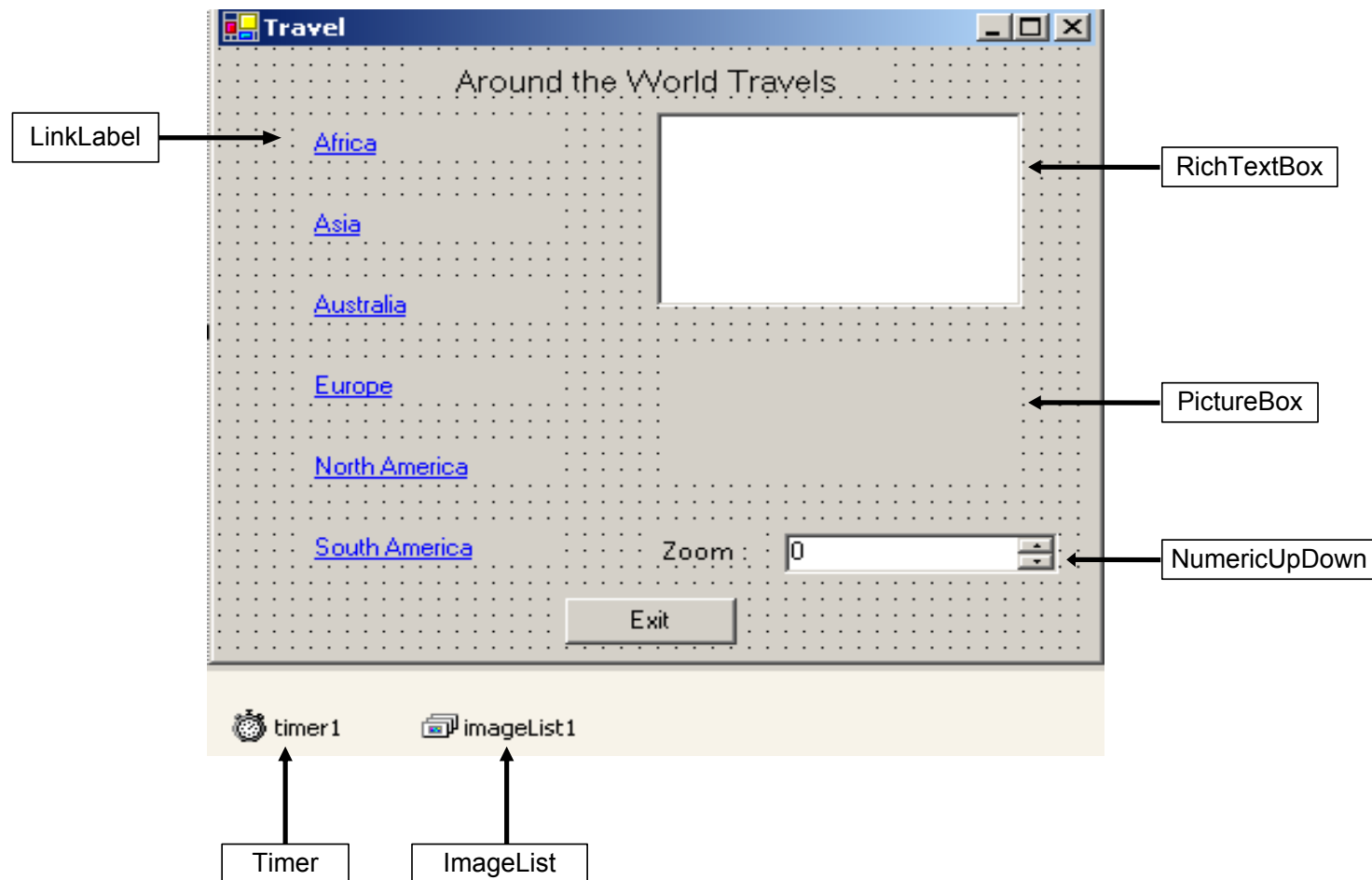
- Дуудах Form-ын
- Show функцийг ашиглана

```
[calledform] [forminstance] = new [calledform] ();  
[forminstance].Show();
```

**Жишээ нь:**

```
Search NewForm = new Search ();  
NewForm.Show();
```

# Жишээ нь:

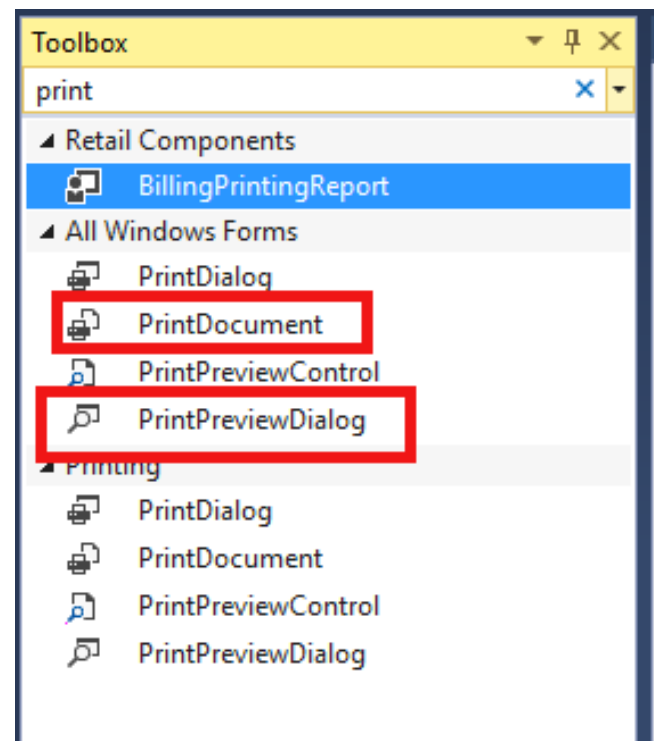


# Хэвлэх контролууд

- PrintDocument



- PrintPreviewControl



# PrintDocument

## Properties

DocumentName

PrintController

PrinterSettings

DefaultPageSettings

## Events

PrintPage

BeginPrint

EndPrint

## Methods

QueryPageSettings

GetType

Print ToString





# Жишээ нь

```
private void printButton_Click(object sender, EventArgs e) {  
    try {  
        streamToPrint = new StreamReader ("C:\\My Documents\\MyFile.txt");  
        try {  
            printFont = new Font("Arial", 10);  
            PrintDocument pd = new PrintDocument();  
            pd.PrintPage += new PrintPageEventHandler(this.pd_PrintPage);  
            pd.Print();  
        }  
        finally {  
            streamToPrint.Close();  
        }  
    }  
    catch (Exception ex) {  
        MessageBox.Show(ex.Message);  
    }  
}
```

# PrintPreview Control

## Properties



Zoom

Document

StartPage

Rows

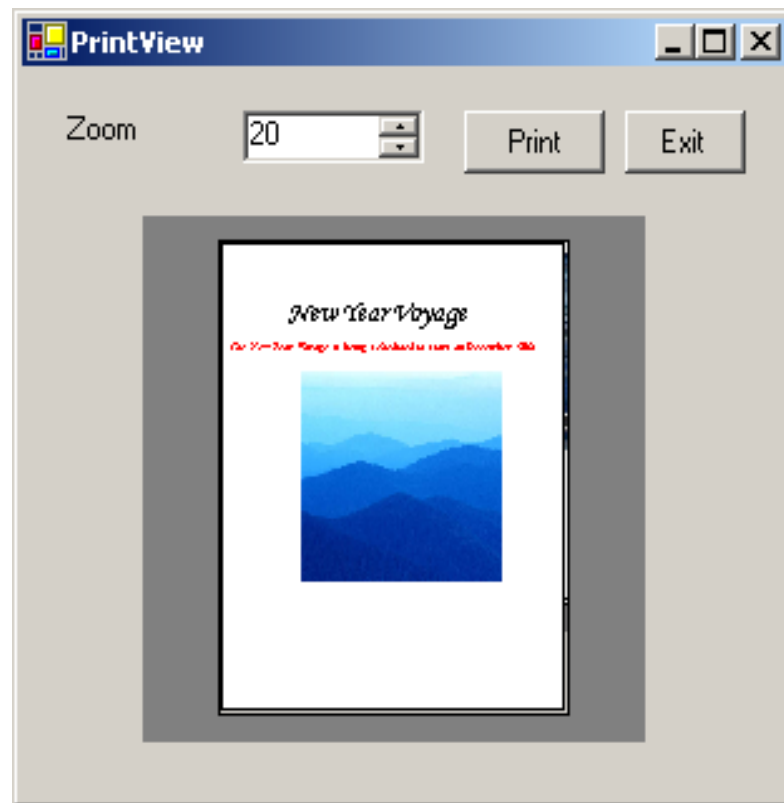
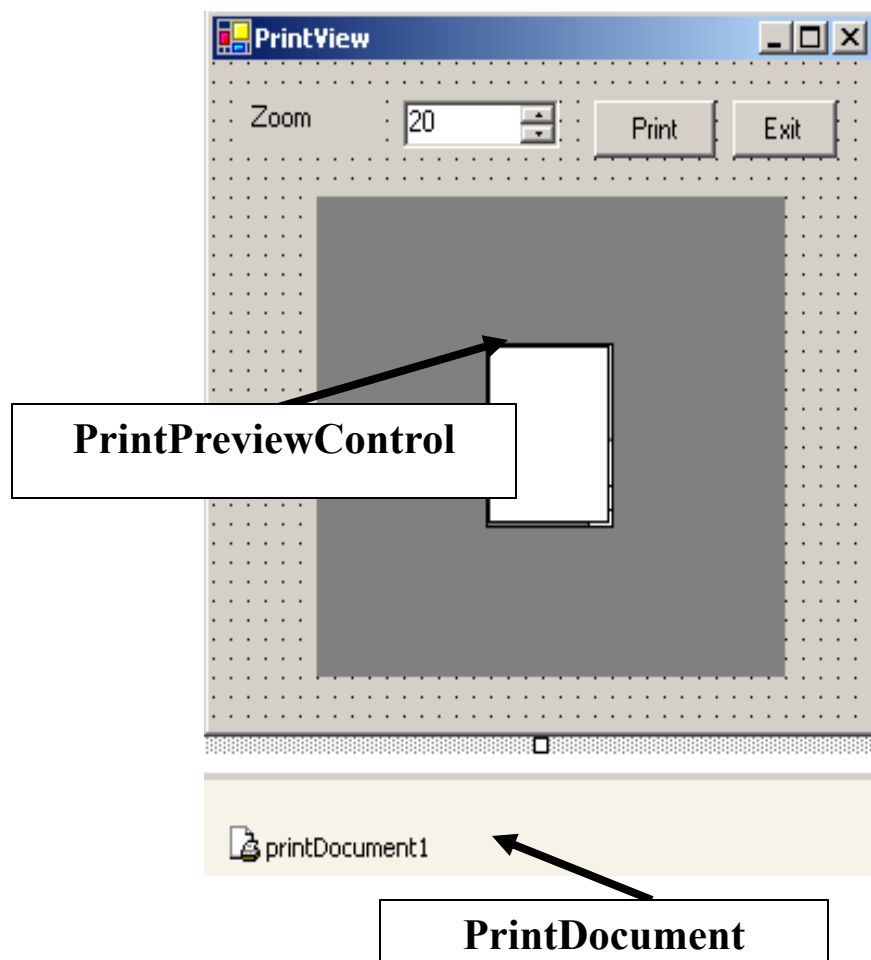
## Methods

AutoZoom

Columns

InvalidatePreview

# Жишээ нь:



# Харилцах цонх

ХЦ



- Хэрэглэгчид мэдээлэл харуулах цонх
- Харуулах мэдээллээс хамааран зарим үйлдлүүдийг гүйцэтгэдэг

Хэлбэр (харагдах)

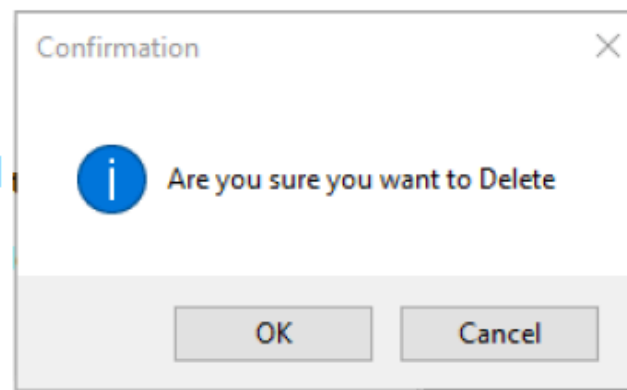
Modal  
Modal

Modeless  
Modeless

# Харилцах цонхны жишээ

## Dialog Box жишээ

```
DialogResult res = MessageBox.Show("Are you sure you want to Delete", "Confirmation", MessageBoxButtons.OKCancel, MessageBoxIcon.Information);  
if (res == DialogResult.OK) {  
    MessageBox.Show("You have clicked Ok Button");  
    //Some task...  
}  
if (res == DialogResult.Cancel) {  
    MessageBox.Show("You have clicked Cancel");  
    //Some task...  
}
```



# Харилцах цонхны хэлбэрүүд

- Харилцах цонх нь үндсэн хоёр хэлбэртэй байна
  - ◆ Хэрэглэгчийн тодорхойлсон харилцах цонх
  - ◆ Үндсэн харилцах цонх
    - OpenFileDialog
    - PageSetUpDialog
    - FontDialog
    - ColorDialog
    - SaveFileDialog
    - PrintPreviewDialog
    - PrintDialog

# OpenFileDialog

ShowReadOnly



RestoreDirectory

AddExtension

*Properties*

FileName

ReadOnlyChecked

CheckPathExists

Title

ShowHelp

InitialDirectory

FilterIndex

FileNames

*CheckFileExists*

MultiSel

DefaultExt

ValidatesNames

Filter

# OpenFileDialog – үргэжлэл ...

*Events*



*Methods*

**HelpRequest**

**OpenFile**

FileOk

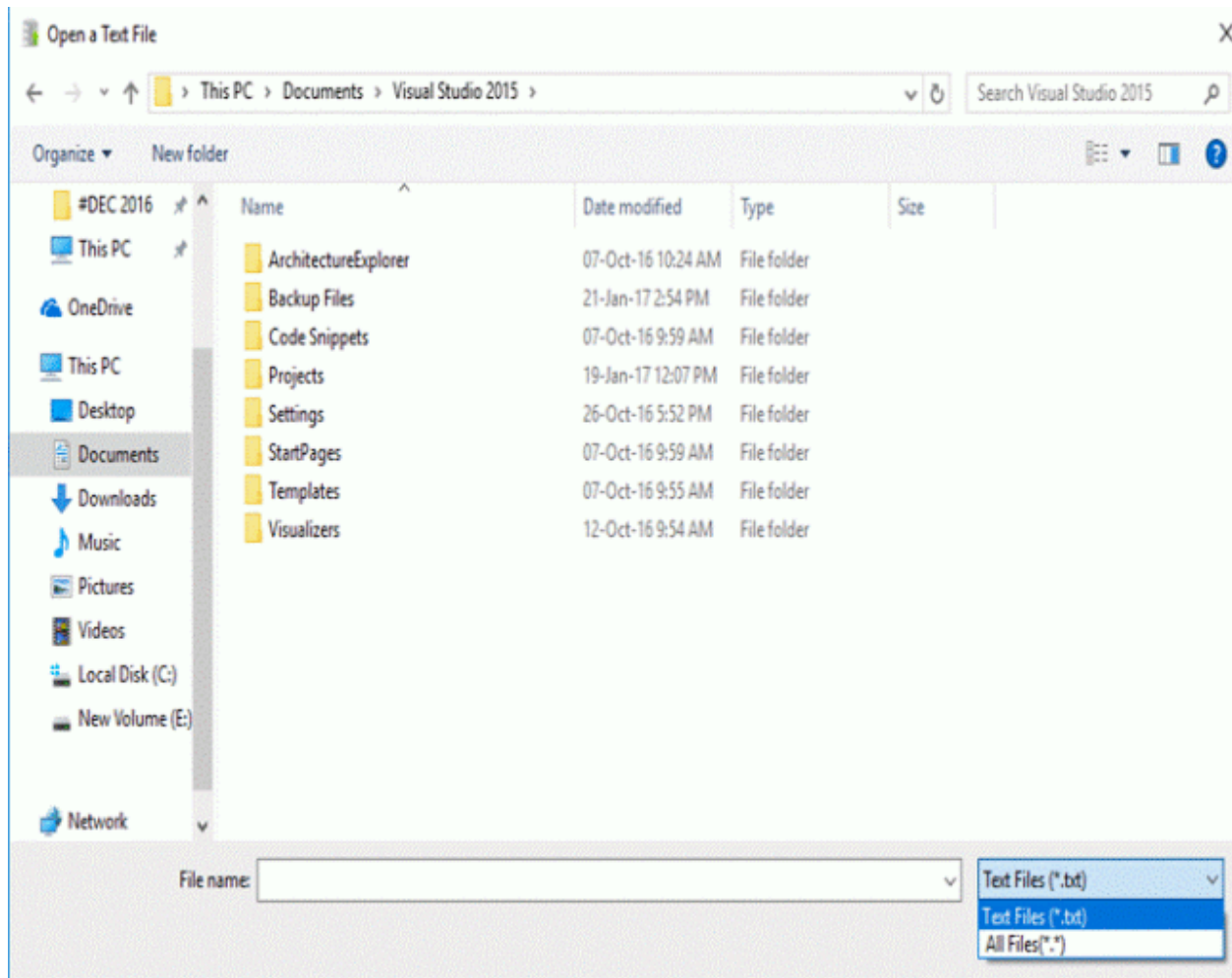
ShowDialog



# OpenFileDialog – жишээ

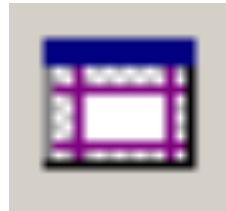
```
OpenFileDialog ofd = new OpenFileDialog();
ofd.Title = "Open a Text File";
ofd.Filter = "Text Files (*.txt) | *.txt | All Files (*.*) | *.*";
//Here you can filter which all files you wanted allow to open
DialogResult dr = ofd.ShowDialog();
if (dr == DialogResult.OK) {
    StreamReader sr = new StreamReader(ofd.FileName);
    txtEx.Text = sr.ReadToEnd();
    sr.Close();
}
```

# OpenFileDialog – жишээ



# PageSetUpDialog

AllowOrientation



PrinterSettings

AllowMargins

*Properties*

AllowPrinter

ShowNetwork

Document

ShowHelp

MinMargins

AllowPaper

PageSettings

# FontDialog

AllowScriptChange

*Properties*

AllowVectorFonts

*AllowSimulations*

AllowVerticalFonts

ScriptsOnly



ShowColor

ShowEffect

FontMustExist

MinSize

Font

MaxSize

ShowApply

*Event*

Color

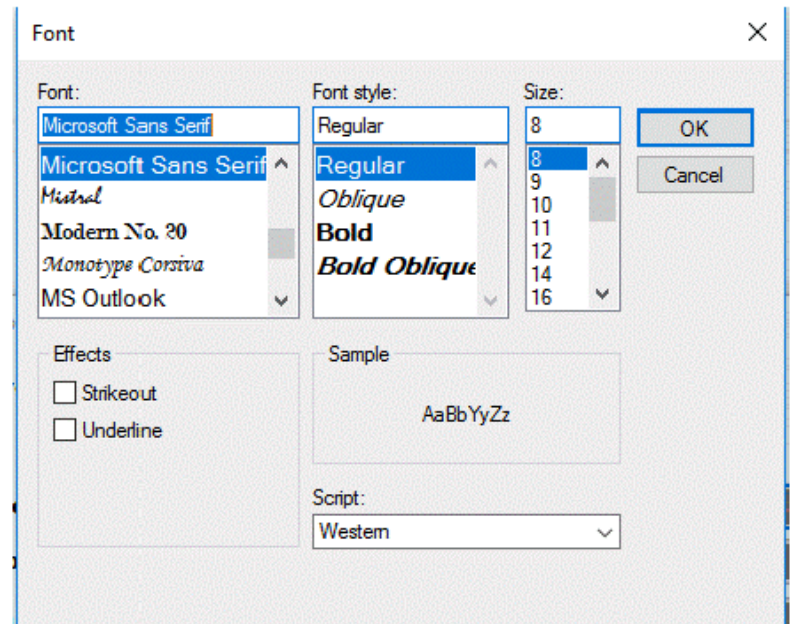
ShowHelp

*FixedPitchOnly*

Apply

# FontDialog

```
//Font Dialog  
FontDialog fdlg = new FontDialog();  
fdlg.ShowDialog();  
txtEx.Font = fdlg.Font;
```



# ColorDialog

Color



AnyColor

CustonColor

*Properties*

SolidColorOnly

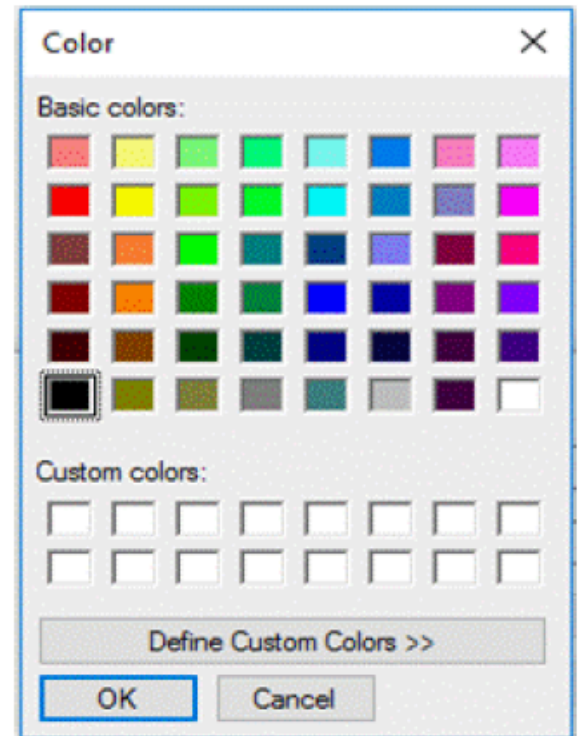
AllowFullOpen

ShowHelp

FullOpen

# ColorDialog

```
//Color Dialog  
ColorDialog cdlg = new ColorDialog();  
cdlg.ShowDialog();  
txtEx.ForeColor = cdlg.color;
```



# SaveFileDialog

*Properties*

*Method*

OverwritePrompt



OpenFile

CreatePrompt



# PrintPreviewDialog



*Property*

Document

# PrintDialog

*Properties*

AllowSelection

PrintToFile

AllowPrintToFile

AllowSomePages



*Method*

*Reset*

# MessageBox

- MessageBox

Бичигдэх хэлбэр:

```
MessageBox.Show (" [Message] " ) ;
```

- MessageBox.Show методоор энэ харилцах цонхыг харуулдаг.

# MessageBox –ны бичигдэх дүрэм

## Бичигдэх хэлбэр:

```
Public static DialogResult Show(string msg);
```

```
Public static DialogResult Show(string msg, string caption);
```

```
public static DialogResult Show(string msg, string caption, MessageBoxButtons buttons);
```

```
public static DialogResult Show(string msg, string caption, MessageBoxButtons buttons, MessageBoxIcon icon);
```

# MessageBox

- MessageBox харицах цонхонд харагдах хэлбэрүүд
  - ◆ MessageBoxButtons
  - ◆ MessageBoxIcon
  - ◆ MessageBoxDefaultButton
  - ◆ MessageBoxOptions

# MessageBoxButtons

Member Name
AbortRetryIgnore
OK
OKCancel
RetryCancel
YesNo
YesNoCancel









# MessageBoxButtons

```
string message = "Do you want to close this window?";  
string title = "Close Window";  
MessageBoxButtons buttons = MessageBoxButtons.YesNo;  
DialogResult result = MessageBox.Show(message, title, buttons);  
if (result == DialogResult.Yes) {  
    this.Close();  
} else {  
    // Do something  
}
```



# MessageBoxIcon

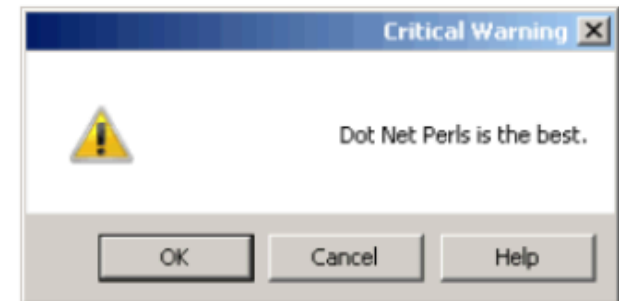
Member Name	Member Name
Asterisk	Question
Error	Stop
Exclamation	Warning
Hand	
Information	
None	

Icon	Name
	Hand
	Question
	Exclamation
	Asterisk
	Stop
	Error
	Warning
	Information



# MessageBoxIcon

```
private void Form1_FormClosing(object sender, FormClosingEventArgs e)
{
    const string message = "Are you sure that you would like to close the form?";
    const string caption = "Form Closing";
    var result = MessageBox.Show(message, caption, MessageBoxButtons.YesNo,
        MessageBoxIcon.Exclamation);
    // If the no button was pressed ...
    if (result == DialogResult.No) {
        // cancel the closure of the form. e.Cancel = true;
    }
}
```



# MessageBoxDefaultButton

Member Name
Button1
Button2
Button3

# MessageBoxOptions

Member Name
DefaultDesktopOnly
RightAlign
RtlReadingService
Notification

# MessageBox-Жишээ

```
OpenFileDialog ofd = new OpenFileDialog();  
if (ofd.ShowDialog() == DialogResult.OK)  
{  
    richTextBox1.LoadFile(ofd.FileName);  
} else {  
    MessageBox.Show("File songoogui", "FileInfo",  
        MessageBoxButtons.OK,  
        MessageBoxIcon.Information);  
}
```

# Form-ууд дээр хийх эffфектүүд

- Transparent forms
- Control docking
- Control Anchoring

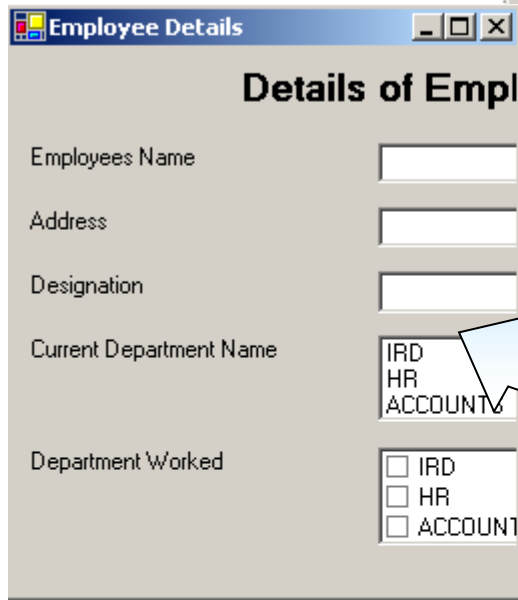
# Transparent Forms

Opacity

```
this.Opacity = 0.83;
```

# Control Anchoring

**Control Anchored**



Employee Details

**Details of Employees**

Employees Name

Address

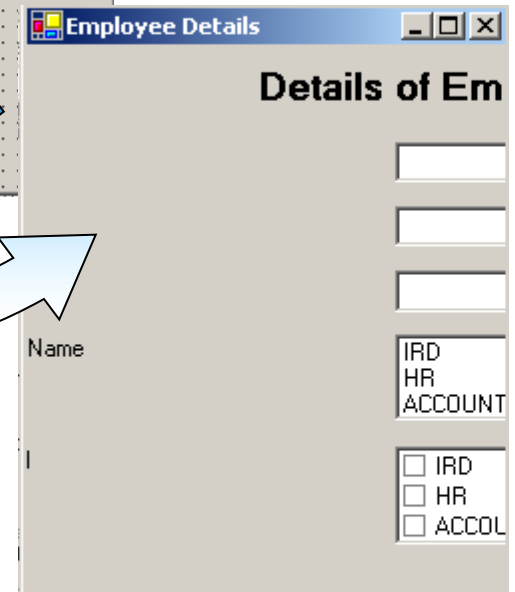
Designation

Current Department Name   
HR  
ACCOUNTS

Department Worked ☐ IRD  
☐ HR  
☐ ACCOUNTS

This form illustrates control anchoring. The controls are anchored to the form's layout, meaning they maintain their relative positions even when the form is resized. The form is titled "Employee Details" and "Details of Employees". It contains fields for "Employees Name", "Address", "Designation", "Current Department Name", and "Department Worked". The "Current Department Name" field is a list box with "IRD", "HR", and "ACCOUNTS" as options. The "Department Worked" field is a group box containing three radio buttons labeled "IRD", "HR", and "ACCOUNTS".

**Control not Anchored**



Employee Details

**Details of Em**

Name

Name   
HR  
ACCOUNT

☐ IRD  
☐ HR  
☐ ACCOL

This form illustrates control not anchoring. The controls are not anchored to the form's layout, meaning they do not maintain their relative positions when the form is resized. The form is titled "Employee Details" and "Details of Em". It contains a "Name" field and several other fields. The "Name" field is a list box with "IRD", "HR", and "ACCOUNT" as options. The "Name" field is a group box containing three radio buttons labeled "IRD", "HR", and "ACCOL".

# Control Docking (контролын байрлал)

Тохиргоо	Тайлбар
Bottom	Агуулагдаж байгаа контролынхоо доод хэсэгт байрлах
Fill	Агуулагдаж байгаа контролынхоо бүрэн дүүргэх
Left	Агуулагдаж байгаа контролынхоо зүүн хэсэгт байрлах
None	Анхны хэлбэрээрээ байрлах
Right	Агуулагдаж байгаа контролынхоо баруун хэсэгт байрлах
Top	Агуулагдаж байгаа контролынхоо дээд хэсэгт байрлах

```
this.DockPadding.Top = 10;
```