

Визуал програмчлал

Ғ.ІТ203

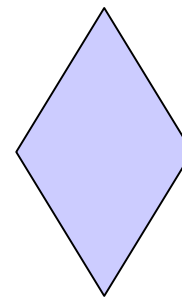
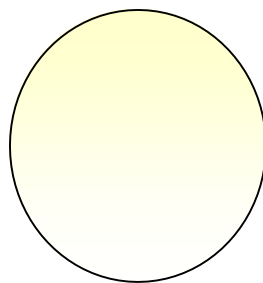
Лекц-10

Гарчиг

- График дүрслэлийг GDI + ашиглан үүсгэх
- GDI+ дэх объектуудыг судлах: Pen, Brush, Color
- GDI+ ашиглан lines, shapes, text
- GDI+ ашиглан зургийг харуулах
- GDI+ ашиглан зураг хэвлэх

GDI+ гэж юу вэ?

- Graph Device Interface (GDI+) нь WinForms-д графикийг хэрэгжүүлэх үндсэн функцийг хангадаг. GDI+ **System.Drawing.dll** дотор байрлана.
 - System.Drawing.Text
 - System.Drawing.Drawing2D
 - System.Drawing.Printing сангууд нь зургийг зурж дүүргэх, будах гэх мэт үйлдлүүдийг хийнэ



GDI+ namespaces

System.Drawing

```
graph TD; A[System.Drawing] --> B[Drawing2D]; A --> C[Imaging]; A --> D[Printing|Text];
```

Drawing2D

Imaging

Printing|Text

График класс



- Графикийн объектыг хэд хэдэн замаар үүсгэж болно:
 - OnPaint() методыг ашиглан график дүрслэлийг зурна
 - CreateGraphics() аргыг ашиглах

System.Drawing namespace.

Graphics reference үүсгэх

```
protected override void OnPaint(PaintEventArgs paintevent)
{
    Graphics graf=paintevent.Graphics;
}
```

```
private void MainForm_Paint(object sender, PaintEventArgs
paintevent)
{
    Graphics graf=paintevent.Graphics;
}
```

Graphics reference үүсгэх

```
private void PaintMe(Control testcontrol)
{
    Graphics graf=testcontrol.CreateGraphics();
    . . .
}
```

```
protected override void OnPaint(PaintEventArgs paintevent)
{
    Bitmap bmpimage=new Bitmap("Water
    Lilies.jpg"); Graphics graf =
    Graphics.FromImage (bmpimage);
    ...
}
```

Graphics object

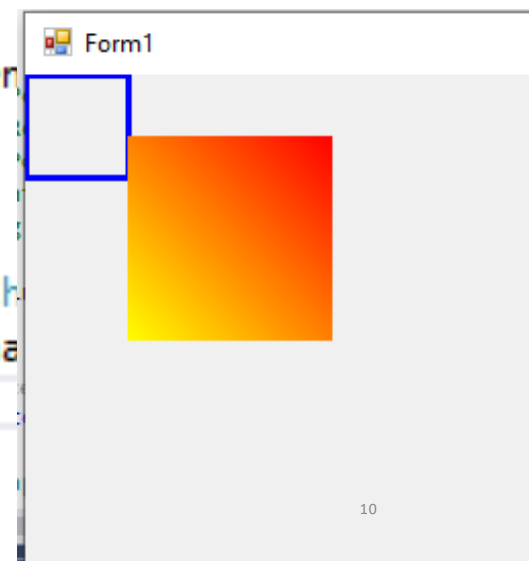
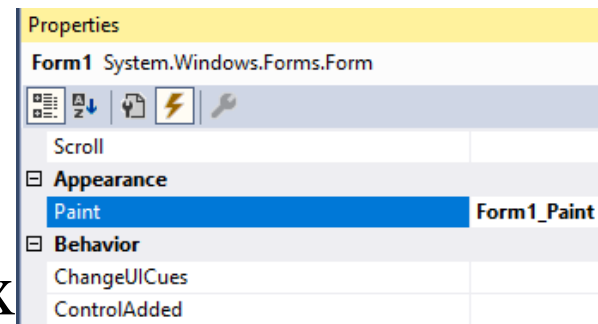
- Graphics object **зургийн гадаргууг** төлөөлнө.
- Windows төхөөрөмжийн контекст/DC/-тэй төсөөтэй.
- DC нь дэлгэцийн төхөөрөмжтэй холбогддог.

```
public Form1()
{
    InitializeComponent();
    Graphics dc = this.CreateGraphics();
    this.Show();
    Pen myPen = new Pen(Color.Blue, 3);
    dc.DrawRectangle(myPen, 0, 0, 50, 50);
}
```

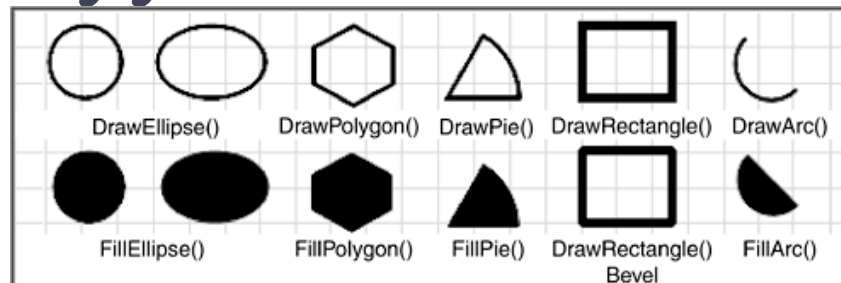

OnPaint функц

- PaintEventArgs параметрийг ашиглах
- Функц дахин тодорхойлох

```
protected override void OnPaint(PaintEventArgs paintevent)
{
    Graphics g = paintevent.Graphics;
    Rectangle rect = new Rectangle(50, 30, 100, 100);
    LinearGradientBrush lBrush = new LinearGradientBrush(
        Color.Yellow, LinearGradientMode.BackwardDiagonal,
        g.FillRectangle(lBrush, rect);
}
```



Graphics класс, методууд



	Функц	Тайлбар
1	clear()	Тухайн өнгөөр зургийн гадаргууг арилгаж, дүүрэх
2	drawarc()	Нумыг зурах
3	drawbezier()	Бейзер муруйг зурах
4	drawellipse()	Эллипсийг зурах
5	drawimage()	Зургийг зурах
6	drawline()	Шулуунийг зурах
7	drawstring()	Текст мөрийг зурах
8	drawrectangle()	Тэгш өнцөгт зурах
9	fillellipse()	Эллипсийг өнгөөр дүүргэх
10	fillrectangle()	Тэгш өнцөгтийг өнгөөр дүүргэх

Pen класс



- Дүрсийн гадаргуу хүрээг зурахад
- width, styles, fill styles, shapes заахад ашиглана.
- Тодорхойлогдоогүй ба бид үүнийг үүсгэнэ.
- **System.Drawing** нэрийн талбарт багтана.
- **Pen** объект нь **Idisposable** интерфейсээс удамшдаг учраас ашиглаж дуусаад **Dispose** методыг дуудаж байх хэрэгтэй.

```
Pen ourpen=new Pen(Color.Blue,5);
```

Дээрх код нь цэнхэр өнгийн үзэгний объектыг 5-ийн өргөнөөр зурна^{1,2}

Pen класс

- Өнгөнөөс гадна шугамын харагдах байдлыг удирдах олон тооны методуудтай.

Функц	Тайлбар
Alignment	Битүү дүрсийн хүрээ яаж зурагдахыг тодорхойлно.
Color	Дүрс, текст зурах өнгө
DashCap	Зурлагын үзүүрүүд ямар байхыг тодорхойлно. Шугамын үзүүрт сум, дугуй гэх мэт дүрсүүд байж болно.
DashOffset	Тасархай хээ шугамын эхлэлээс ямар зайтай байхыг заана
DashStyle	Зурлагын төрлүүд.
PenType	Шугам яаж будагдахыг заана. Үргэлж, уусалттай гэх мэт
StartCap EndCap	Шугамын эхлэл, төгсгөл үзүүрийн хэлбэрийг заана. Энд LineCap төрлийн утгууд зааж болно. Жишээ нь LineCap.Square.

Жишээ

```
private void OnPaint(object sender, PaintEventArgs e)
{
    Graphics g = e.Graphics ;
    Pen pn = new Pen( Color.Blue, 100 );
    Rectangle rect = new Rectangle(50, 50, 200, 100);
    g.DrawEllipse( pn, rect );
}
```



Brush класс



- Бодит хэлбэрийн өнгийг бууруулах, өсгөх байдлаар ашиглана.
- Abstract класс нь биелэгдэх боломжгүй байдаг
- System.Drawing нэрийн талбарт хамаатай.
- Brushes нь SolidBrush, LinearGradientBrush, TextureBrush зэрэг классыг ашиглан үүсгэж болно.

```
Color pink = Color.FromArgb(241, 105, 190);  
SolidBrush sldBrush = new SolidBrush(pink);  
g.FillRectangle(sldBrush, 300, 150, 70, 70);
```

Brush класс



- Бодит хэлбэрийн өнгийг бууруулах, өсгөх байдлаар ашиглана.
- Abstract класс нь биелэгдэх боломжгүй байдаг
- ***System.Drawing*** нэрийн талбарт хамаатай.
- ***Brushes*** нь ***SolidBrush***, ***LinearGradientBrush***, ***TextureBrush*** зэрэг классыг ашиглан үүсгэж болно.

```
LinearGradientBrush lgBrush = new LinearGradientBrush(new Rectangle(0, 0, 20, 20), Color.Violet, Color.LightSteelBlue, LinearGradientMode.Vertical);  
g.FillRectangle(lgBrush, 300, 220, 70, 70);
```

Brush класс



- Бодит хэлбэрийн өнгийг бууруулах, өсгөх байдлаар ашиглана.
- Abstract класс нь биелэгдэх боломжгүй байдаг
- ***System.Drawing*** нэрийн талбарт хамаатай.
- ***Brushes*** нь ***SolidBrush***, ***LinearGradientBrush***, ***TextureBrush*** зэрэг классыг ашиглан үүсгэж болно.

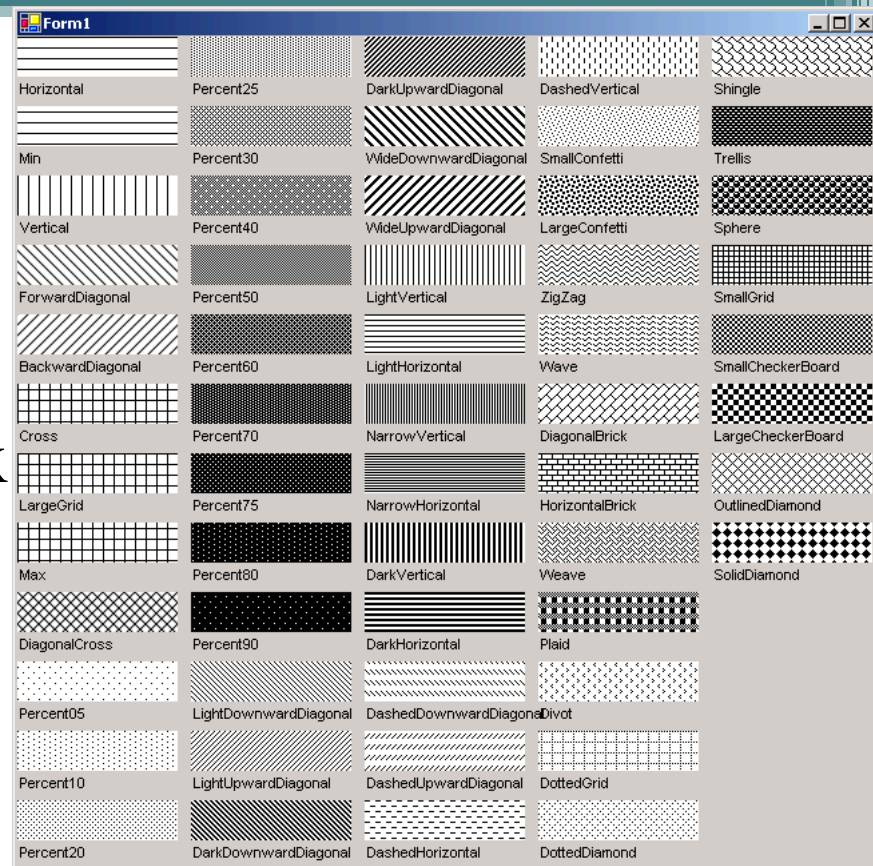
```
TextureBrush tBrush = new TextureBrush(Image.FromFile(@"Images\csharp.jpg"));  
g.FillRectangle(tBrush, 370, 290, 70, 70);
```


Brush класс



Сараалжин
шугамаар дүүргэх
багс

- HatchBrush
- hStyle
- forecolor
- backcolor



Point үүсгэх

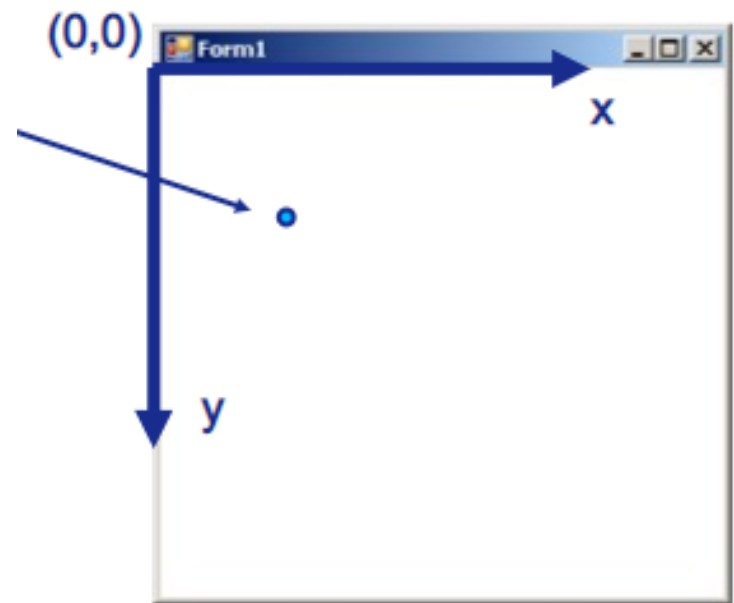
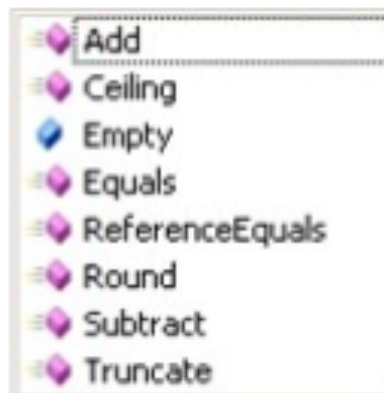
Point

- `Point p = new Point(25, 30);`

PointF

- Float arguments

Functionality



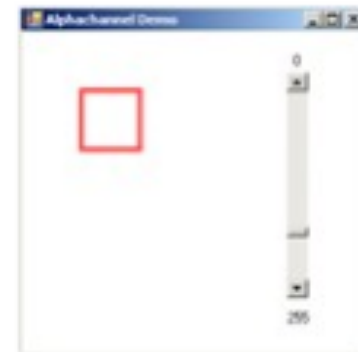
Жишээ

```
Point p1 = new Point(x,y);  
Point p2 = new Point(x+22,y+12);  
Point p3 = new Point(x+22,y+32);  
Point p4 = new Point(x,y+44);  
Point p5 = new Point(x-22,y+32);  
Point p6 = new Point(x-22,y+12);  
Point[] myPoints = {pt1, pt2, pt3, pt4, pt5, pt6};  
g.DrawPolygon(myPen, myPoints);  
g.FillPolygon(Brushes.Black, myPoints);
```

Color

Named colors

- Red, Blue, Purple, ...
- `Color c = Color.Blue;`



Custom color:

- `Color c = Color.FromArgb(255, 255, 0, 0);`

DrawLine() функц

- Графикийн DrawLine() арга нь дэлгэц дээр шулуун зурахад хэрэглэгддэг.

```
public void DrawLine(Pen, Point, Point);  
public void DrawLine(Pen, PointF, PointF);  
public void DrawLine(Pen, int, int, int, int);  
public void DrawLine(Pen, float, float, float, float);
```



DrawString() функц

- Ямар нэгэн бэлэн текст ашиглахгүйгээр дэлгэцэндээр текстийг харуулах.

```
public void DrawString(string, Font, Brush, PointF);  
public void DrawString(string, Font, Brush, RectangleF);  
public void DrawString(string, Font, Brush, PointF, StringFormat);  
public void DrawString(string, Font, Brush, RectangleF, StringFormat);  
public void DrawString(string, Font, Brush, float, float);  
public void DrawString(string, Font, Brush, float, float,  
StringFormat);
```



DrawImage()

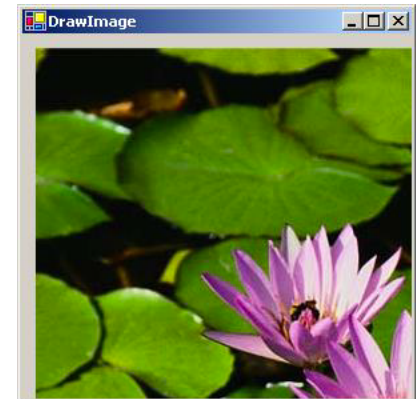
- Зургийг объект ашиглан зурахад хэрэглэгдэнэ. Ерөнхийдөө GIF, JPG, BMP зургийг зурж авдаг.



```
public void DrawImage(Image, Point)
public void DrawImage(Image, Point[])
public void DrawImage(Image, PointF)
public void DrawImage(Image, PointF[])
public void DrawImage(Image, Rectangle)
```

JPG зураг зургах

```
protected override void OnPaint(PaintEventArgs p_event)
{
    int x_coord,y_coord;
    Image testimage=Image.FromFile("Water lilies.jpg");
    Graphics graf=p_event.Graphics;
    x_coord=10; y_coord=10;
    graf.DrawImage(testimage,x_coord,y_coord);
}
```



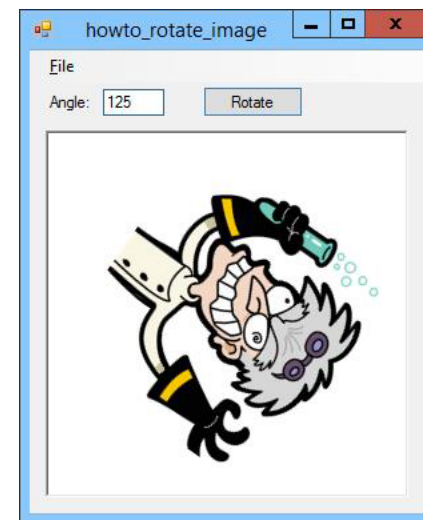
Дүрсийг зурах

```
protected override void OnPaint(PaintEventArgs  
    paintevt)  
{  
    Image img=Image.FromFile("Provider.ico");  
    Graphics graf=Graphics.FromImage(img);  
    paintevt.Graphics.DrawImage(img,50,50);  
}
```

Зураг боловсруулалт

Зураг дээр мөн эргүүлэлт, толин ойлт хийх эсвэл тодорхой цэгүүдийг солих зэргээр нарийн хувиргалтуудыг хийж болно.

- Original
- Mirrored
- Flipped
- Rotated 90



```
Bitmap bmp = Bitmap(fname);
```

```
Bmp.RotateFlip(RotateFlipType.Rotate90FlipNone);
```

GraphicsPathкласссаар дүрс зурах

Үндсэн дүрсүүдийн олонлогыг зурахад ашиглагдана.

```
Point[] ptsT = {new Point(120,20), new Point(160,20), new Point(140,50)};
Point[] ptsL = {new Point(90,50), new Point(90,90), new Point(120,50)};
Point[] ptsB = {new Point(120,120), new Point(160,120), new Point(140,90)};
Point[] ptsR = {new Point(190,90), new Point(190,50), new Point(160,70)};
Point[] ptsC = {new Point(140,50), new Point(120,70), new Point(140,90) , new Point(160,70)};
GraphicsPath myGPath = new GraphicsPath ();
myGPath.AddPolygon(ptsT );
myGPath.AddPolygon(ptsL );
myGPath.AddPolygon(ptsB );
myGPath.AddPolygon(ptsR );
myGPath.AddPolygon(ptsC );
g.DrawPath(new Pen(Color.Blue, 2), myGPath);
g.FillPath(Brushes.Gold, myGPath);
```

Дүрс дээр үзэгдэл боловсруулах

```
this.pictureBox1.MouseDown += new  
MouseEventHandler(down_Picture);
```

```
private void down_Picture(object Sender, MouseEventArgs e){  
    if(myPath.Contains(e.X, e.Y)){  
        MessageBox.Show("Төгсгөлгүй загалмай");  
    }  
}
```

Жишээ

