

Класс Graphics

System.Drawing

System.Drawing.Drawing2D

System.Drawing.Imaging

System.Drawing.Text

System.Drawing.Printing

Работа с графикой:

1. Создание объект Graphics.
2. Установка перо и кисть.
3. Отрисовка линий и фигур.

Создание объекта Graphics

1 способ: получить ссылку на объект Graphics через объект PaintEventArgs при обработке события Paint формы или элемента управления.

Это обычный способ получения ссылки на графический объект при создании кода рисования элементов управления.

```
private void Form1_Paint(object sender,  
    System.Windows.Forms.PaintEventArgs pe)  
{    Graphics g = pe.Graphics;}
```

2 способ. Вызвать метод `CreateGraphics` элемента управления или формы, чтобы получить ссылку на объект `Graphics`, соответствующий поверхности рисования этой формы или элемента управления.

Используется, если необходимо рисовать на поверхности уже существующей формы или элемента управления.

```
Graphics g;  
g = this.CreateGraphics();
```

3 способ. Создайте объект Graphics из любого объекта, унаследованного от класса Image.

Этот способ используется, когда требуется изменить существующее изображение.

```
Bitmap myBitmap = new Bitmap(@"C:\Documents  
and Settings\Joe\Pics\myPic.bmp");  
Graphics g = Graphics.FromImage(myBitmap);
```

Класс Pen

— служит для рисования линий, контуров и других геометрических объектов.

Конструкторы

	Имя	Описание
⇒	<code>Pen(Brush)</code>	Инициализирует новый экземпляр класса Pen указанным значением <code>Brush</code> .
⇒	<code>Pen(Brush, Single)</code>	Инициализирует новый экземпляр класса Pen с заданными свойствами <code>Brush</code> и <code>Width</code> .
⇒	<code>Pen(Color)</code>	Инициализирует новый экземпляр класса Pen указанным цветом.
⇒	<code>Pen(Color, Single)</code>	Инициализирует новый экземпляр класса Pen указанными свойствами <code>Color</code> и <code>Width</code> .

Свойства и методы класса Pen

- **Color**
- **DashStyle**
- **StartCap** и **EndCap**
- **Width**
- **Clone()**

```
mypen.StartCap = LineCap.RoundAnchor;  
g.DrawLine(mypen, 10, 10, 100, 10);  
mypen.StartCap = LineCap.ArrowAnchor;  
g.DrawLine(mypen, 10, 30, 100, 30);  
mypen.StartCap = LineCap.DiamondAnchor;  
g.DrawLine(mypen, 10, 50, 100, 50);  
mypen.StartCap = LineCap.Flat;  
g.DrawLine(mypen, 10, 70, 100, 70);  
mypen.StartCap = LineCap.RoundAnchor;  
g.DrawLine(mypen, 10, 90, 100, 90);  
mypen.StartCap = LineCap.SquareAnchor;  
g.DrawLine(mypen, 10, 110, 100, 110);
```




SolidBrush - класс

Определяет кисть одного цвета. Кисти используются для заливки графических фигур, таких как прямоугольники, эллипсы, круги, многоугольники и контуры.

- Color
- Clone ()

Конструкторы

	Имя	Описание
	<code>SolidBrush(Color)</code>	Инициализирует новый объект SolidBrush указанного цвета.

HatchBrush - класс

Задаёт прямоугольную кисть со стилем штриховки, основным цветом и цветом фона.

Конструкторы

	Имя	Описание
⇒	<code>HatchBrush(HatchStyle, Color)</code>	Инициализирует новый экземпляр класса <code>HatchBrush</code> с указанным перечислением <code>HatchStyle</code> и основным цветом.
⇒	<code>HatchBrush(HatchStyle, Color, Color)</code>	Инициализирует новый экземпляр класса <code>HatchBrush</code> с указанным перечислением <code>HatchStyle</code> , основным цветом и цветом фона.

- **BackgroundColor** - Получает цвет интервалов между линиями штриховки, нарисованными данным объектом HatchBrush.
- **ForegroundColor** - Получает цвет линий штриховки, нарисованных данным объектом HatchBrush.
- **HatchStyle** - Получает стиль штриховки для данного объекта HatchBrush.

Шаблон штриховки содержит два цвета: один из них определяется свойством BackgroundColor и заполняет фон, а другой используется для линий, формирующих шаблон на этом фоне, и определяется свойством ForegroundColor.

Свойство HatchStyle определяет тип шаблона кисти имеет и любое значение из перечисления HatchStyle.

В перечислении HatchStyle имеется более пятидесяти элементов.

На следующем рисунке представлен эллипс, залитый по шаблону горизонтальной штриховки.



Графические примитивы

DrawLine (параметры)

Список перегрузок

	Имя	Описание
⇒◆	<code>DrawLine(Pen, Int32, Int32, Int32, Int32)</code>	Проводит линию, соединяющую две точки, задаваемые парами координат.
⇒◆	<code>DrawLine(Pen, Point, Point)</code>	Проводит линию, соединяющую две структуры <code>Point</code> .
⇒◆	<code>DrawLine(Pen, PointF, PointF)</code>	Проводит линию, соединяющую две структуры <code>PointF</code> .
⇒◆	<code>DrawLine(Pen, Single, Single, Single, Single)</code>	Проводит линию, соединяющую две точки, задаваемые парами координат.

DrawEllipse (параметры)

Список перегрузок

	Имя	Описание
⇒	<code>DrawEllipse(Pen, Int32, Int32, Int32, Int32)</code>	Рисует эллипс, определяемый ограничивающим прямоугольником, заданным с помощью координат верхнего левого угла прямоугольника, высоты и ширины.
⇒	<code>DrawEllipse(Pen, Rectangle)</code>	Рисует эллипс, определяемый ограничивающей структурой <code>Rectangle</code> .
⇒	<code>DrawEllipse(Pen, RectangleF)</code>	Рисует эллипс, определяемый ограничивающей структурой <code>RectangleF</code> .
⇒	<code>DrawEllipse(Pen, Single, Single, Single, Single)</code>	Рисует эллипс, определяемый ограничивающим прямоугольником, заданным с помощью пары координат, ширины и высоты.

DrawRectangle (параметры)

Список перегрузок

	Имя	Описание
☰	<code>DrawRectangle(Pen, Int32, Int32, Int32, Int32)</code>	Рисует прямоугольник, определяемый парой координат, шириной и высотой.
☰	<code>DrawRectangle(Pen, Rectangle)</code>	Рисует прямоугольник, определяемый структурой <code>Rectangle</code> .
☰	<code>DrawRectangle(Pen, Single, Single, Single, Single)</code>	Рисует прямоугольник, определяемый парой координат, шириной и высотой.

FillEllipse (параметры)

Список перегрузок

	Имя	Описание
⇒	<code>FillEllipse(Brush, Int32, Int32, Int32, Int32)</code>	Заполняет внутреннюю часть эллипса, определяемого ограничивающим прямоугольником, заданным с помощью пары координат, ширины и высоты.
⇒	<code>FillEllipse(Brush, Rectangle)</code>	Заполняет внутреннюю часть эллипса, определяемого ограничивающим прямоугольником, который задан структурой <code>Rectangle</code> .
⇒	<code>FillEllipse(Brush, RectangleF)</code>	Заполняет внутреннюю часть эллипса, определяемого ограничивающим прямоугольником, который задан структурой <code>RectangleF</code> .
⇒	<code>FillEllipse(Brush, Single, Single, Single, Single)</code>	Заполняет внутреннюю часть эллипса, определяемого ограничивающим прямоугольником, заданным с помощью пары координат, ширины и высоты.

FillRectangle (параметры)

Список перегрузок

	Имя	Описание
⇒	<code>FillRectangle(Brush, Int32, Int32, Int32, Int32)</code>	Заполняет внутреннюю часть прямоугольника, который задается парой координат, шириной и высотой.
⇒	<code>FillRectangle(Brush, Rectangle)</code>	Заполняет внутреннюю часть прямоугольника, определяемого структурой <code>Rectangle</code> .
⇒	<code>FillRectangle(Brush, RectangleF)</code>	Заполняет внутреннюю часть прямоугольника, определяемого структурой <code>RectangleF</code> .
⇒	<code>FillRectangle(Brush, Single, Single, Single, Single)</code>	Заполняет внутреннюю часть прямоугольника, который задается парой координат, шириной и высотой.

DrawString (параметры)

<code>DrawString(String, Font, Brush, PointF)</code>	Создает заданную текстовую строку в указанном месте с помощью заданных объектов <code>Brush</code> и <code>Font</code> .
<code>DrawString(String, Font, Brush, PointF, StringFormat)</code>	Создает заданную текстовую строку в указанном месте с помощью заданных объектов <code>Brush</code> и <code>Font</code> , используя атрибуты форматирования заданного формата <code>StringFormat</code> .
<code>DrawString(String, Font, Brush, RectangleF)</code>	Создает заданную текстовую строку в указанном прямоугольнике с помощью заданных объектов <code>Brush</code> и <code>Font</code> .
<code>DrawString(String, Font, Brush, RectangleF, StringFormat)</code>	Создает заданную текстовую строку в указанном прямоугольнике с помощью заданных объектов <code>Brush</code> и <code>Font</code> , используя атрибуты форматирования заданного формата <code>StringFormat</code> .
<code>DrawString(String, Font, Brush, Single, Single)</code>	Создает заданную текстовую строку в указанном месте с помощью заданных объектов <code>Brush</code> и <code>Font</code> .
<code>DrawString(String, Font, Brush, Single, Single, StringFormat)</code>	Создает заданную текстовую строку в указанном месте с помощью заданных объектов <code>Brush</code> и <code>Font</code> , используя атрибуты форматирования заданного формата <code>StringFormat</code> .

