**28 Работа с графикой. Трехмерная графика.**

Задание 1. Нарисуйте 5 различных фигур треугольник, эллипс, закрашенный круг, закрашенный прямоугольник, сектор.

Листинг программы:

public class Shapes : FrameworkElement

{

SolidColorBrush brush = new SolidColorBrush(Colors.Black);

private Pen mainPen = new Pen(Brushes.Black, 1.0);

protected override void OnRender(DrawingContext drawingContext)

{

drawingContext.DrawEllipse(null, mainPen, new Point(50, 50), 40, 40);

drawingContext.DrawEllipse(null, mainPen, new Point(50, 50), 30, 30);

drawingContext.DrawEllipse(null, mainPen, new Point(50, 50), 20, 20);

drawingContext.DrawRectangle(null, mainPen, new Rect(100, 30, 40, 60));

drawingContext.DrawRectangle(null, mainPen, new Rect(120, 70, 40, 60));

drawingContext.DrawRectangle(null, mainPen, new Rect(140, 110, 40, 60));

drawingContext.DrawRectangle(null, mainPen, new Rect(160, 150, 40, 60));

drawingContext.DrawRectangle(null, mainPen, new Rect(180, 190, 40, 60));

for (int i = 1; i <= 8; i++)

{

for (int j = 1; j <= 8; j++)

{

if (((i % 2 == 0) && (j % 2 != 0)) || ((i % 2 != 0) && (j % 2 == 0)))

drawingContext.DrawRectangle(brush, mainPen, new Rect(20 \* (j + 10), 20 \* i, 20, 20));

else

drawingContext.DrawRectangle(null, mainPen, new Rect(20 \* (j + 10), 20 \* i, 20, 20));

}

}

drawingContext.DrawEllipse(brush, mainPen, new Point(50, 150), 20, 20);

drawingContext.DrawRectangle(brush, mainPen, new Rect(50, 200, 40, 60));

drawingContext.DrawEllipse(null, mainPen, new Point(50, 300), 20, 20);

drawingContext.DrawLine(mainPen, new Point(300, 350), new Point(350, 300));

drawingContext.DrawLine(mainPen, new Point(350, 300), new Point(350, 400));

drawingContext.DrawLine(mainPen, new Point(350, 400), new Point(300, 350));

base.OnRender(drawingContext);

Таблица 1.1 – Выходные и входные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
|  |  |

Анализ результатов:

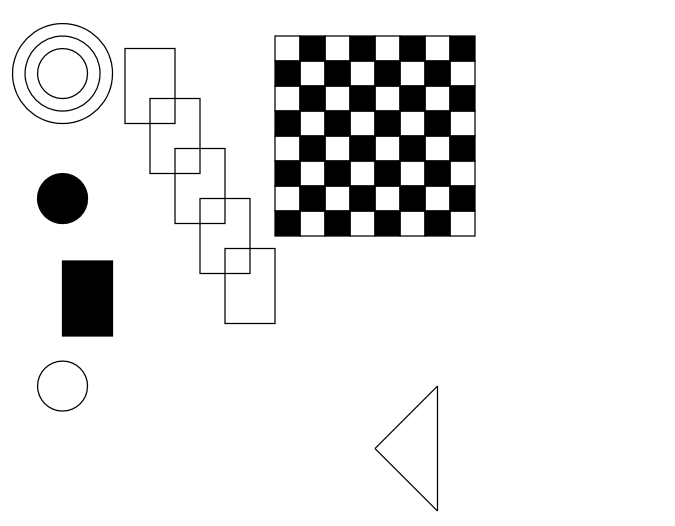


Рисунок 1.1 – Результат работы программы

Источник: собственная разработка

Задание 2. Нарисовать 3 кнопки: Треугольную, круглую, в форме пирамиды.

Листинг программы:

<Button Content="" HorizontalAlignment="Left" Height="100" Margin="236,252,0,0" VerticalAlignment="Top" Width="100" Background="#FF040404">

<Button.Clip>

<EllipseGeometry Center="50,50" RadiusX="50" RadiusY="50"></EllipseGeometry>

</Button.Clip>

</Button>

<Button Content="" HorizontalAlignment="Left" Height="85" Margin="459,254,0,0" VerticalAlignment="Top" Width="88" RenderTransformOrigin="0.5,0.5" Background="#FF6E8888">

<Button.RenderTransform>

<TransformGroup>

<ScaleTransform/>

<SkewTransform/>

<RotateTransform Angle="91.145"/>

<TranslateTransform/>

</TransformGroup>

</Button.RenderTransform>

<Button.Clip>

<PathGeometry>

<PathFigure IsClosed="True" StartPoint="10 10">

<LineSegment Point="10 100"/>

<LineSegment Point="100 100"/>

</PathFigure>

</PathGeometry>

</Button.Clip>

</Button>

<Button Content="" HorizontalAlignment="Center" Height="100" Margin="0,238,0,0" VerticalAlignment="Top" Width="41" RenderTransformOrigin="0.5,0.5" Background="#FF316263">

<Button.RenderTransform>

<TransformGroup>

<ScaleTransform/>

<SkewTransform/>

<RotateTransform Angle="-0.402"/>

<TranslateTransform/>

</TransformGroup>

</Button.RenderTransform>

<Button.Clip>

<PathGeometry>

<PathFigure IsClosed="True" StartPoint="10 10">

<LineSegment Point="-1 100"/>

<LineSegment Point="80 200"/>

</PathFigure>

</PathGeometry>

</Button.Clip>

</Button>

Таблица 1.2 – Выходные и входные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
|  |  |

Анализ результатов:

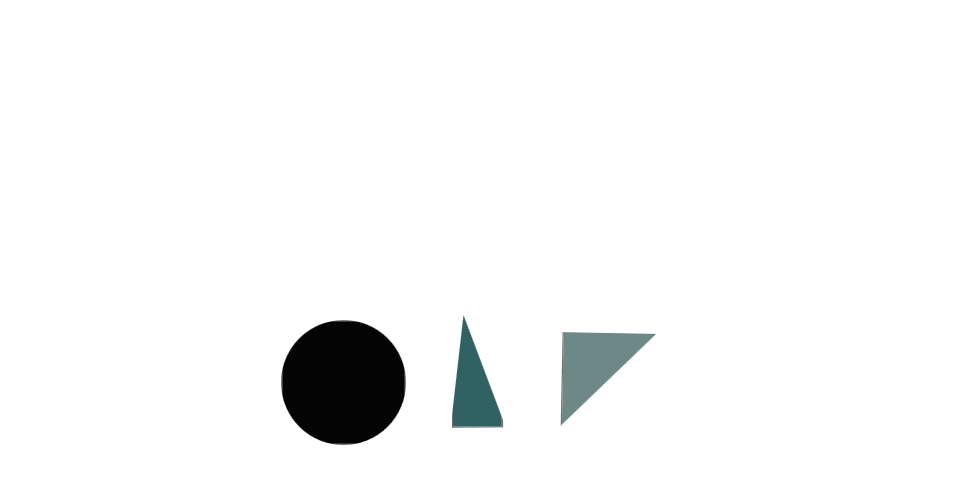


Рисунок 1.2 – Результат работы программы

Источник: собственная разработка