**ЗАДАНИЕ 28**

Задание 1. Изучить и выполнить примеры Лабораторной работы №10 по книге Программирование на С# Демин, Дорофеев. Выполните задания. 1. Нарисуйте 5 различных фигур треугольник, эллипс, закрашенный круг, закрашенный прямоугольник, сектор.

Листинг программы:

namespace zad1

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

private void Form1\_Paint(object sender, PaintEventArgs e)

{

Graphics g = e.Graphics;

Point[] trianglePoints = { new Point(10, 50), new Point(50, 10), new Point(90, 50) };

g.DrawPolygon(Pens.Black, trianglePoints);

Rectangle ellipseRect = new Rectangle(110, 10, 80, 60);

g.DrawEllipse(Pens.Black, ellipseRect);

Rectangle filledCircleRect = new Rectangle(220, 10, 60, 60);

g.FillEllipse(Brushes.Blue, filledCircleRect);

Rectangle filledRectangleRect = new Rectangle(320, 10, 80, 60);

g.FillRectangle(Brushes.Green, filledRectangleRect);

Rectangle sectorRect = new Rectangle(440, 10, 80, 60);

g.FillPie(Brushes.Red, sectorRect, 0, 90);

Rectangle targetRect = new Rectangle(10, 210, 80, 80);

g.DrawEllipse(Pens.Black, targetRect);

g.DrawEllipse(Pens.Black, targetRect.X + 20, targetRect.Y + 20, 40, 40);

g.DrawEllipse(Pens.Black, targetRect.X + 40, targetRect.Y + 40, 0, 0);

int squareSize = 20;

for (int i = 0; i < 5; i++)

{

Rectangle squareRect = new Rectangle(150 + i \* squareSize, 210 + i \* squareSize, squareSize, squareSize);

g.FillRectangle(Brushes.White, squareRect);

g.DrawRectangle(Pens.Black, squareRect);

}

int gridSize = 10;

for (int row = 0; row < 8; row++)

{

for (int col = 0; col < 8; col++)

{

Brush brush = (row + col) % 2 == 0 ? Brushes.Black : Brushes.White;

Rectangle cellRect = new Rectangle(300 + col \* gridSize, 210 + row \* gridSize, gridSize, gridSize);

g.FillRectangle(brush, cellRect);

}

}

}

private void Form1\_Load(object sender, EventArgs e)

{

}

}

}

Анализ результатов:

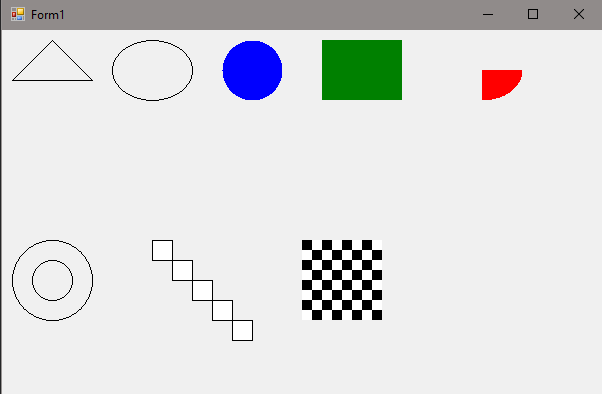


Рисунок 28.1 – Результат работы программы

Задание 2. Выполните задание 1 в приложении WPF используя объект DrawingContext(смотрите пример).

Листинг программы:

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows;

using System.Windows.Controls;

using System.Windows.Data;

using System.Windows.Documents;

using System.Windows.Input;

using System.Windows.Media;

using System.Windows.Media.Imaging;

using System.Windows.Navigation;

using System.Windows.Shapes;

namespace zad2

{

/// <summary>

/// Логика взаимодействия для MainWindow.xaml

/// </summary>

public partial class MainWindow : Window

{

public MainWindow()

{

InitializeComponent();

}

}

public class TestDrawing : Control

{

protected override void OnRender(DrawingContext drawingContext)

{

base.OnRender(drawingContext);

Point p1 = new Point(50, 50);

Point p2 = new Point(100, 100);

Point p3 = new Point(150, 50);

Pen trianglePen = new Pen(Brushes.Blue, 2);

drawingContext.DrawLine(trianglePen, p1, p2);

drawingContext.DrawLine(trianglePen, p2, p3);

drawingContext.DrawLine(trianglePen, p3, p1);

Rect ellipseRect = new Rect(new Point(200, 50), new Size(100, 50));

Pen ellipsePen = new Pen(Brushes.Red, 2);

drawingContext.DrawEllipse(null, ellipsePen, new Point(250, 75), 50, 25);

Rect filledCircleRect = new Rect(new Point(350, 50), new Size(50, 50));

Brush filledCircleBrush = new SolidColorBrush(Color.FromRgb(0, 255, 0));

drawingContext.DrawEllipse(filledCircleBrush, null, new Point(375, 75), 25, 25);

Rect filledRectangleRect = new Rect(new Point(450, 50), new Size(100, 50));

Brush filledRectangleBrush = new SolidColorBrush(Color.FromRgb(255, 0, 0));

drawingContext.DrawRectangle(filledRectangleBrush, null, filledRectangleRect);

Point center = new Point(600, 75);

double radius = 50;

double startAngle = 30;

double endAngle = 120;

StreamGeometry sectorGeometry = new StreamGeometry();

using (StreamGeometryContext context = sectorGeometry.Open())

{

context.BeginFigure(center + new Vector(radius, 0), true, true);

context.ArcTo(center, new Size(radius, radius), 0, false, SweepDirection.Clockwise, true, true);

context.LineTo(center, true, true);

}

Pen sectorPen = new Pen(Brushes.Purple, 2);

drawingContext.DrawGeometry(null, sectorPen, sectorGeometry);

Rect square1 = new Rect(new Point(50, 150), new Size(50, 50));

Rect square2 = new Rect(new Point(75, 175), new Size(50, 50));

Rect square3 = new Rect(new Point(100, 200), new Size(50, 50));

Rect square4 = new Rect(new Point(125, 225), new Size(50, 50));

Rect square5 = new Rect(new Point(150, 250), new Size(50, 50));

Pen squarePen = new Pen(Brushes.Black, 2);

drawingContext.DrawRectangle(null, squarePen, square1);

drawingContext.DrawRectangle(null, squarePen, square2);

drawingContext.DrawRectangle(null, squarePen, square3);

drawingContext.DrawRectangle(null, squarePen, square4);

drawingContext.DrawRectangle(null, squarePen, square5);

for (int i = 0; i < 8; i++)

{

for (int j = 0; j < 8; j++)

{

Rect cellRect = new Rect(new Point(300 + i \* 50, 150 + j \* 50), new Size(50, 50));

Brush cellBrush = (i + j) % 2 == 0 ? Brushes.White : Brushes.Black;

drawingContext.DrawRectangle(cellBrush, null, cellRect);

}

}

}

}

}

Анализ результатов:

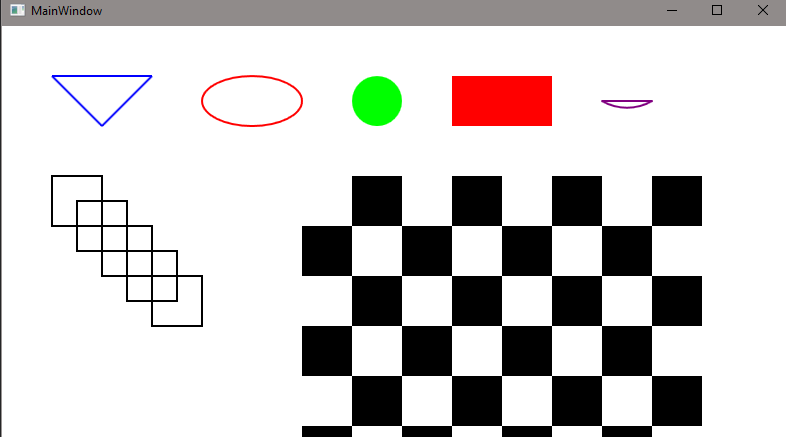


Рисунок 28.2 – Результат работы программы