

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РФ



Федеральное государственное бюджетное образовательное
учреждение высшего профессионального образования
«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

Методические указания к лабораторной работе №12

«Классы Path и Geometry»

по дисциплине «Технологии разработки пользовательских интерфейсов»

Вичугов В.Н., доцент каф. АиКС

Томск 2012

Класс Path («путь») является наследником от абстрактного класса Shape и позволяет объединять в себе различные геометрические фигуры (линии, прямоугольники, эллипсы, дуги, кривые линии).

Свойство Data класса Path содержит объект-наследник класса Geometry, который определяет геометрическую фигуру. Объект Geometry определяет фигуру, а объект Path рисует её с использованием заданной кисти Stroke и фона Fill.

В свойство Data можно записать объект одного из следующих классов-наследников абстрактного класса Geometry:

LineGeometry – прямая линия (объект аналогичен фигуре Line);

RectangleGeometry – прямоугольник (объект аналогичен фигуре Rectangle);

EllipseGeometry – эллипс (объект аналогичен фигуре Ellipse);

GeometryGroup – набор объектов класса Geometry;

CombinedGeometry – объединение из двух объектов класса Geometry;

PathGeometry – сложную фигуру, состоящую из отрезков прямой, эллиптических дуг, кривых Безье;

StreamGeometry – упрощенный вариант задания сложных фигур с помощью мини-языка описания геометрии; объект StreamGeometry не может быть изменен в процессе выполнения программы.

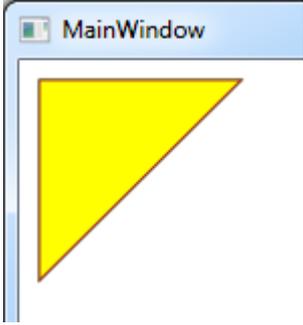
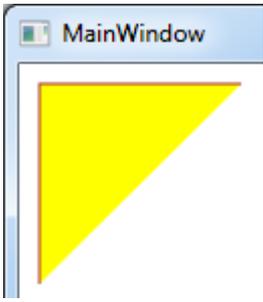
Если в свойстве Data записан текст (например, Data="M10,10 L180,10 180,50 10,80 z"), то на основе этого текста создается объект StreamGeometry, который записывается в свойство Data.

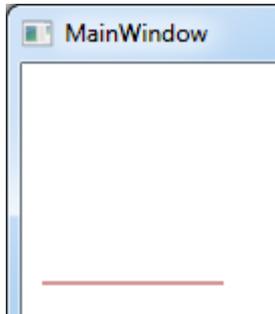
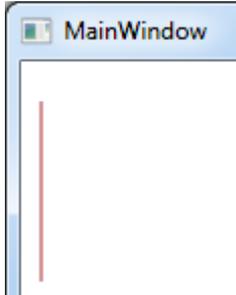
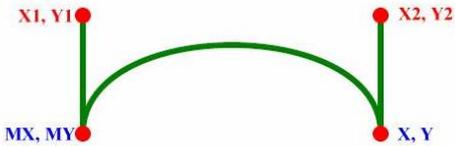
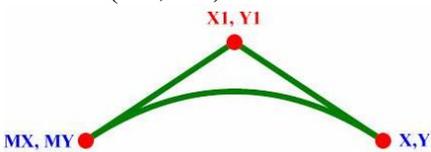
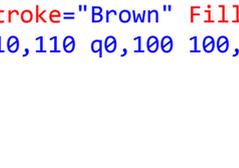
Классы PathGeometry и StreamGeometry

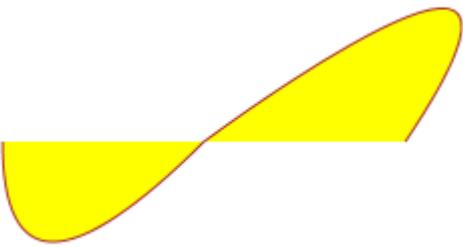
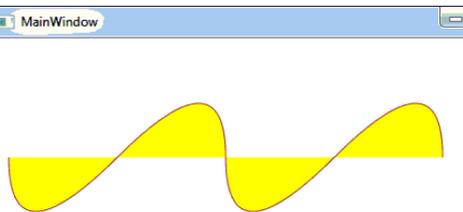
Объект класса PathGeometry может содержать один или несколько объектов класса PathFigure, каждый из которых может содержать один или несколько сегментов. Также данные сегменты могут быть описаны на мини-языке описания геометрии в объекте StreamGeometry.

Виды сегментов:

Описание	Мини-язык	PathGeometry
Начальная точка новой фигуры	M X, Y m dX, dY (Move) X, Y – абсолютные координаты, dX, dY – смещение относительно последней точки; Пример: <pre><Path Data="M10,20 ... M3.5,10 ..." /></pre>	Свойство StartPoint класса PathGeometry в формате "X,Y" (только абсолютные координаты) Пример: <pre><Path> <Path.Data> <PathGeometry> <PathFigure StartPoint="10,20"> ... </PathFigure> ... <PathFigure StartPoint="3.5,10"> ... </PathFigure> </PathGeometry> </Path.Data> </Path></pre>
Завершение фигуры. К фигуре	Z	Отрезок, соединяющий

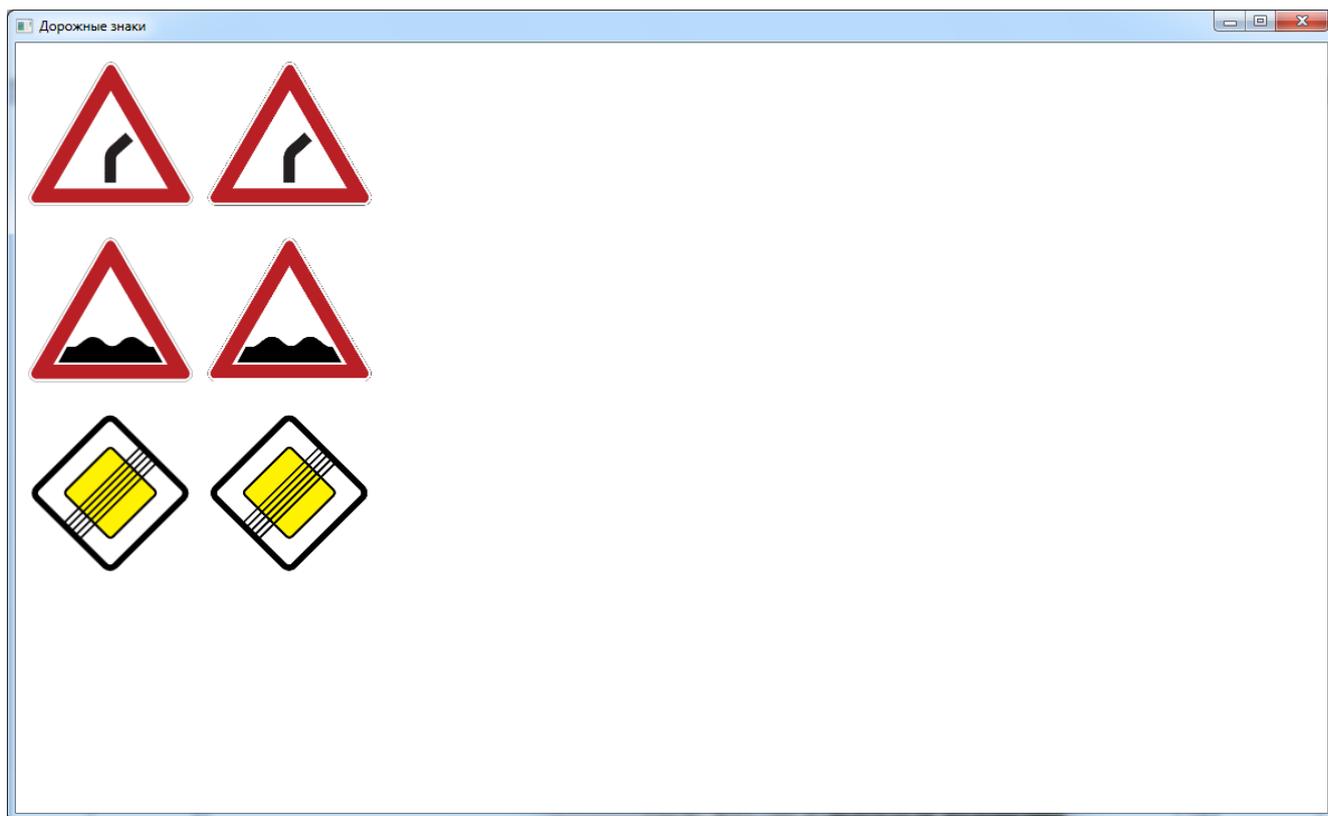
<p>добавляется отрезок, соединяющий текущую точку с начальной.</p>	<p>Z</p> <p>(Close)</p> <p>Пример: <code><Path Data="M10,20 ... Z M3.5,10 ..." /></code></p> <p>Первая фигура завершена, вторая – нет.</p>	<p>текущую точку с начальной, добавляется в том случае, если свойство IsClosed=True</p> <p>Пример: <code><PathFigure StartPoint="10,20" IsClosed="True"></code> <code>...</code> <code></PathFigure></code> <code><PathFigure StartPoint="3.5,10"></code> <code>...</code> <code></PathFigure></code></p> <p>Первая фигура завершена, вторая – нет.</p>
<p>Отрезок от текущей точки до указанной точки. Для всех сегментов новая точка становится текущей.</p>	<p>L X, Y l dx, dy</p> <p>(Line)</p> <p>Пример завершенной фигуры: <code><Path Stroke="Brown" Fill="Yellow" Data="M10,110 L10,10 L110,10 Z" /></code></p>  <p>Повторяющиеся буквы L можно не повторять. Пример незавершенной фигуры: <code><Path Stroke="Brown" Fill="Yellow" Data="M10,110 L10,10 110,10" /></code></p>  <p>Относительные координаты: <code><Path Stroke="Brown" Fill="Yellow" Data="M10,110 l0,-100 100,0" /></code></p>	<p><LineSegment Point="X,Y" /></p> <p>Пример: <code><PathFigure StartPoint="10,110" IsClosed="True"></code> <code><LineSegment Point="10,10" /></code> <code></PathFigure></code> <code><LineSegment Point="110,10" /></code> <code></PathFigure></code></p>
<p>Горизонтальный отрезок от текущей точки до указанной координаты.</p>	<p>H X h dx</p> <p>(Horizontal line)</p> <p>Пример: <code><Path Stroke="Brown" Fill="Yellow"</code></p>	

	<pre>Data="M10,110 H100" /></pre>  <pre><Path Stroke="Brown" Fill="Yellow" Data="M10,110 h90" /></pre>	
<p>Вертикальный отрезок от текущей точки до указанной координаты.</p>	<p>V Y v dY</p> <p>(Vertical line) Пример: <pre><Path Stroke="Brown" Fill="Yellow" Data="M10,110 V10" /></pre></p>  <pre><Path Stroke="Brown" Fill="Yellow" Data="M10,110 v-90" /></pre>	
<p>Кубическая кривая Безье от текущей точки (MX, MY) до заданной (X, Y) с начальной контрольной точкой (X1, Y1) и конечной (X2, Y2)</p> 	<p>C X1, Y1 X2, Y2 X, Y c dx1, dy1 dx2, dy2 dx, dy</p> <p>(Cubic Bezier Curve) Пример: <pre><Path Stroke="Brown" Fill="Yellow" Data="M10,110 c0,50 100,-50 100,0" /></pre></p> 	<pre><BezierSegment Point1="X1,Y1" Point2="X2,Y2" Point3="X,Y"/></pre> <p>Пример: <pre><PathFigure StartPoint="10,110"></pre> <pre><BezierSegment Point1="10,160" Point2="110,60" Point3="110,110"/></pre> <pre></PathFigure></pre> </p>
<p>Квадратичная кривая Безье от текущей точки (MX, MY) до заданной (X, Y) с контрольной точкой (X1, Y1)</p> 	<p>Q X1, Y1 X, Y q dx1, dy1 dx, dy</p> <p>(Quadratic Bezier Curve) Пример: <pre><Path Stroke="Brown" Fill="Yellow" Data="M10,110 q0,100 100,0" /></pre></p> 	<pre><QuadraticBezierSegment Point1="X1,Y1" Point2="X,Y"/></pre> <p>Пример: <pre><PathFigure StartPoint="10,110"></pre> <pre><QuadraticBezierSegment Point1="10,210" Point2="110, 110"/></pre> </p>

		<code></PathFigure></code>
<p>Кубическая кривая Безье от текущей точки (MX, MY) до заданной (X,Y) с конечной контрольной точкой (X1, Y1). Начальная контрольная точка представляет собой отражение контрольной точки предыдущей команды.</p>	<p>S X1, Y1 X, Y</p> <p>s dx1, dy1 dx, dy</p> <p>(Smooth Cubic Bezier Curve) Пример: <code><Path Stroke="Brown" Fill="Yellow" Data="M10,110 q0,100 100,0 s200,-150 100,0" /></code></p> 	
<p>Квадратичная кривая Безье от текущей точки (MX, MY) до заданной (X,Y). Контрольная точка представляет собой отражение контрольной точки предыдущей команды.</p> 	<p>T X, Y</p> <p>t dx, dy</p> <p>(Smooth Quadratic Bezier Curve) Пример: <code><Path Stroke="Brown" Fill="Yellow" Data="M10,110 q0,100 100,0 t100,0 100,0 100,0" /></code></p> <p>Повторяющиеся буквы (в данном случае t) можно не указывать.</p> 	

Задание 1

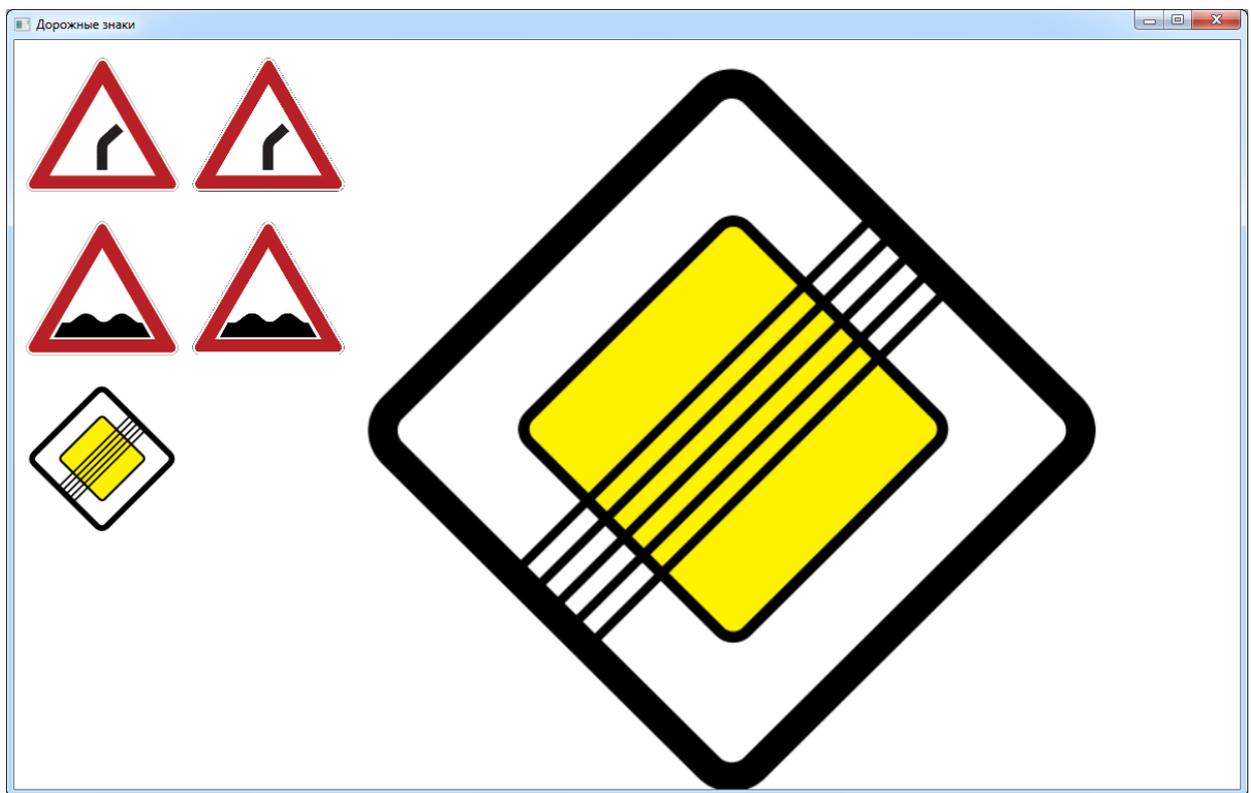
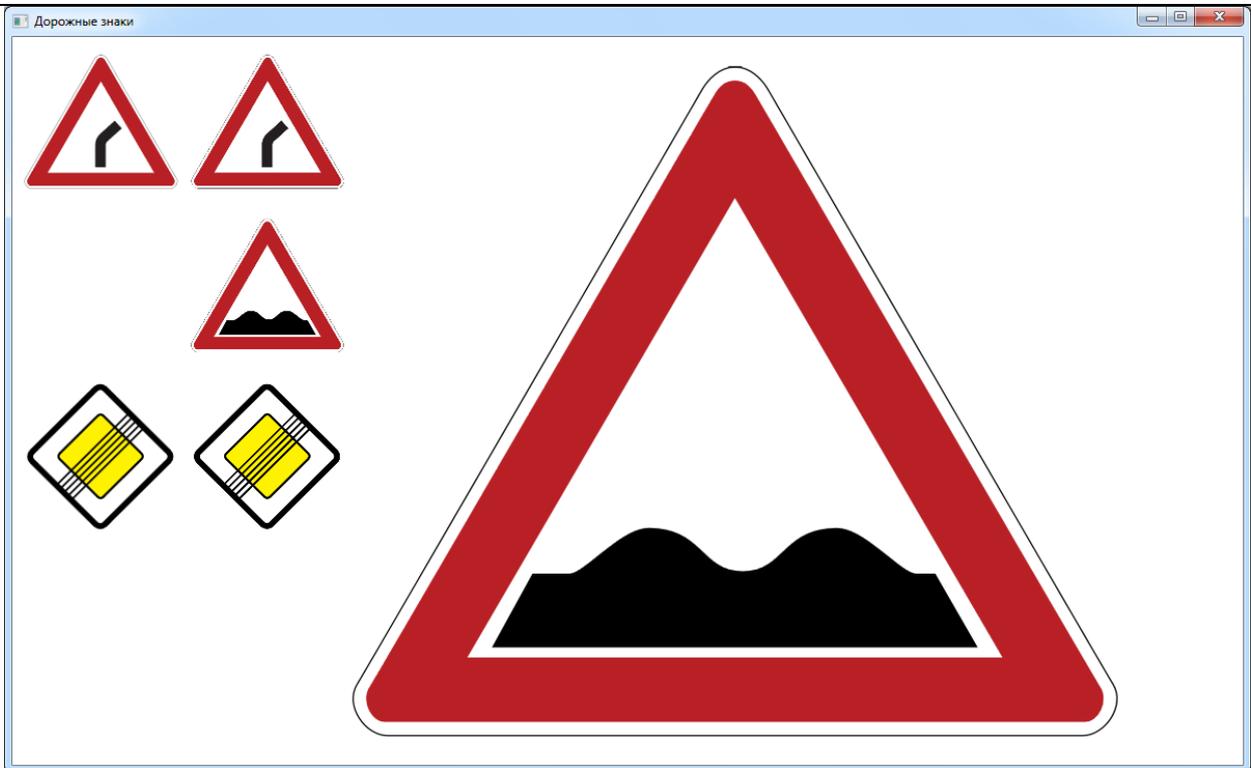
Разработайте WPF-приложение «Дорожные знаки» со следующим интерфейсом:



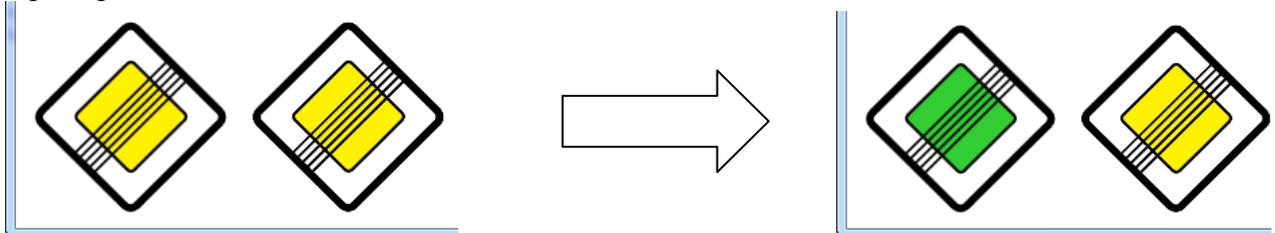
По запросу «Дорожные знаки Российской Федерации» найдите в Википедии соответствующую страницу и выберите на ней три **сложных** знака, **отличных от изображенных в примере**. Обязательно, чтобы при нажатии на знак открывался файл SVG.

В первом столбце знаки изображаются с помощью объектов Path, во втором – с помощью картинок в формате PNG. Конвертацию SVG-файла в XAML- и PNG-файлы можно произвести с помощью программы Inkscape.

При нажатии на любой знак он плавно увеличивается и перемещается в правую часть окна. Через две секунды он плавно возвращается в исходное состояние:



При наведении курсора мыши на какой-то один знак плавно меняется цвет какой-либо части этого знака. Пример:



Подсказки:

Для плавного изменения цвета используйте объект `ColorAnimation` и свойство `Fill.(SolidColorBrush.Color)`