

gMapIoKml

Создание скриптов на языке KML
(создание kml-файлов для отображения
маркеров, полилиний и полигонов в Google
Earth)

Rev. 1.1

Содержание

Введение	4
1. Создать файл kml	6
2. Закрыть файл kml	6
3. Преобразовать значения цвета и прозрачности в формат для kml	6
4. Задать стили графических элементов	7
5. Нарисовать графический маркер (точку)	9
6. Нарисовать полилинию	10
7. Нарисовать полигон	11
8. Пример использования gMapIoKml (скрипт gMapIoKmlTest)	12
Список литературы	16

История изменений

Версия	Дата	Описание изменений
1.0	2021-06-12	Документ создан // Иван Д.
1.1	2024-03-25	Текст отредактирован, подготовлены версии на английском и русском языках // Иван Д., Маргарита Б.

Список таблиц

<i>Table 0.1</i> Функции gMapIoKml	4
---	---

Список рисунков

<i>Figure 0.1</i> Структура библиотеки gMapIoKml	5
<i>Figure 8.1</i> Результат выполнения gMapIoKmlTest (отображение файла 112.kml в Google Earth)	13

Введение

Библиотека gMapIoKml содержит функции MatLab / GNU Octave (**Table 0.1**) для создания скриптов на языке KML [1, 2, 3], которые используются для отображения графических объектов в программе Google Earth. Функции библиотеки выполняют запись текстовых «команд» на языке KML в файл формата *.kml.

При написании библиотеки целью было не использование всех возможностей и всего синтаксиса языка KML, а выполнение необходимого минимума действий по созданию kml-файла, при минимальном времени, затраченном на изучение функций библиотеки gMapIoKml. Функции библиотеки позволяют выполнить следующую последовательность действий с kml-файлом:

- 1) создать kml-файл с прописыванием необходимых «тегов начала файла» (gMapIoKmlOpen);
- 2) задать в kml-файле стили отображения графических элементов – графического маркера, линии, полигона, всплывающего окна подсказки и др. (gMapIoKmlSetStyle);
- 3) задать в kml-файле «команды»: нарисовать графический маркер, нарисовать полилинию, нарисовать полигон (gMapIoKmlSetPoint, gMapIoKmlSetPolyline, gMapIoKmlSetPolygon);
- 4) закрыть kml-файл с прописыванием необходимых «тегов конца файла» (gMapIoKmlClose).

Функция gMapIoKmlColor используется для преобразования значений цвета и прозрачности к формату, который используется в kml. Преобразованные значения цвета и прозрачности используются при задании стилей (команда gMapIoKmlSetStyle).

Table 0.1 Функции gMapIoKml

Название функции	Описание функции
gMapIoKmlOpen	Создать файл kml
gMapIoKmlColor	Преобразовать значения цвета и прозрачности в формат, используемый в kml
gMapIoKmlSetStyle	Задать стили графических элементов
gMapIoKmlSetPoint	Нарисовать графический маркер (точку)
gMapIoKmlSetPolyline	Нарисовать полилинию
gMapIoKmlSetPolygon	Нарисовать полигон
gMapIoKmlClose	Закрыть файл kml
gMapIoKmlTest	Скрипт gMapIoKml, с примером использования функций библиотеки

При создании библиотеки использовался код функции kml_line [4].

В настоящее время функции для чтения и записи файлов формата kml включены в стандартную библиотеку Mapping Toolbox.

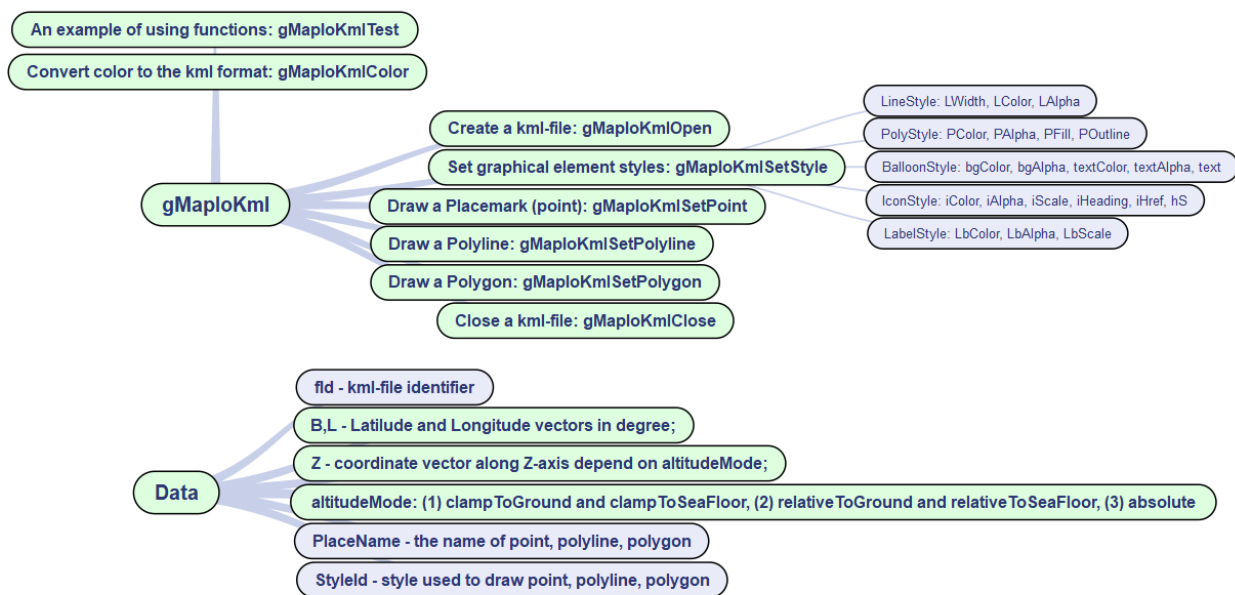


Figure 0.1 Структура библиотеки gMapIoKml

1. Создать файл kml

Функция `fId=gMapIoKmlOpen(FileName)`

Создает файл формата *.kml с именем FileName и записывает в него «теги начала kml-файла».

Параметры:

fId – идентификатор файла скрипта;

FileName – имя создаваемого KML-файла.

Пример использования функции:

```
>> fId=gMapIoKmlOpen('c:\temp\112.kml');
```

2. Закрыть файл kml

Функция `status=gMapIoKmlClose(fId)`

Записывает в kml-файл «теги конца kml-файла» и закрывает файл с идентификатором fId.

Параметры:

fId – идентификатор kml-файла;

status – переменная, принимающая значение '0', если операция закрытия файла прошла успешно, значение '-1' в противном случае.

Пример использования функции:

```
>> gMapIoKmlClose(fId);
```

3. Преобразовать значения цвета и прозрачности в формат для kml

Функция `[LColor,LAlpha]=gMapIoKmlColor(LColor,LAlpha)`

Преобразует значения цвета и прозрачности в формат, используемый в kml-файле, из иных форматов. Преобразованные значения цвета и прозрачности используются при задании стилей (команда `gMapIoKmlSetStyle`).

Параметры:

LColor – цвет, который может быть задан тремя способами: (1) последовательностью символов `bbggrr`, где `bb` – синий, `gg` – зеленый, `rr` – красный цвет, например `'00ffff'`; (2) буквенным обозначением `'r'`, `'g'`, `'b'`; (3) вектором из трех чисел от 0 до 255, кодирующим красный-зеленый-синий цвета, например `[255 255 0]`;

LAlpha – прозрачность, которая может быть задана двумя способами: (1) последовательностью символов от `'00'` в случае полной прозрачности, до значения `'ff'` в случае непрозрачности; (2) числом от 0 до 255.

Пример использования функции:

Выходные значения LColor и LAlpha, возвращаются в формате, используемом в kml-файле – как последовательность символов (например, `'00ffff'` или `'9f'`).

Пример использования функции:

```
>> [LColor,LAlpha]=gMapIoKmlColor('r',255);
```

4. Задать стили графических элементов

Функция gMapIoKmlSetStyle(fId,StyleId,varargin)

Записывает в kml-файл теги стиля для графических элементов. Фактически, функция создает стиль с именем StyleId, который содержит стиль отображения для пяти элементов: графических маркеров (IconStyle), всплывающих окон с описанием (BalloonStyle), полилиний (LineStyle), полигонов (PolyStyle) и названий объектов при отображении в 3D (LabelStyle). В коде функции неявно задаются значения colorMode='normal', gx:labelVisibility=0.

Параметры:

fId – идентификатор kml-файла;

StyleId – уникальный идентификатор стиля (строка или число); например: 'Std09', 28, 'RedStyle';

varargin – массив ячеек, при помощи которого можно задавать стиль для следующих элементов:

'LineStyle', 'BalloonStyle', 'IconStyle', 'PolyStyle', 'LabelStyle'; например: {'LineStyle',4,'r','ff'}. Если значение какого-либо из параметров (параметры для элементов стиля перечислены ниже) будет задано пустым массивом [], то kml-код соответствующий этому параметру не будет записан в kml-файл. Код цвета (LColor) и прозрачность заливки (LAlpha) должны быть заданы пустыми массивами одновременно, чтобы соответствующий им kml-код не был записан в kml-файл.

'LineStyle' определяет стиль (цвет, цветовой режим и ширину линии) отображения элементов линейной геометрии. Линейная геометрия включает в себя контуры полигонов и выдавливание (extruded “tether”) графических маркеров относительно земной поверхности (если выдавливание включено). Параметры элемента 'LineStyle':

LWidth – ширина линии в пикселях; пример: '05', 4;

LColor – цвет линии, который задается последовательностью символов bbggrt, где bb – синий, gg – зеленый, rr – красный цвет, например '00ffff';

LAlpha – прозрачность линии, которая задается последовательностью символов от '00' в случае полной прозрачности, до 'ff' в случае непрозрачности.

'PolyStyle' определяет стиль отображения полигонов, включая многоугольные и линейные (polygon extrusions, which look like the walls of buildings and line extrusions, which look like solid fences [1]). Параметры элемента 'PolyStyle':

PColor – цвет заливки полигона, который задается последовательностью символов bbggrt, где bb – синий, gg – зеленый, rr – красный цвет, например '00ffff';

PAlpha – прозрачность полигона, которая задается последовательностью символов от '00' в случае полной прозрачности, до 'ff' в случае непрозрачности;

PFill – логическая переменная, определяющая следует ли использовать заливку для полигона или нет. Пример: '1', '0';

POutline – логическая переменная, определяющая следует ли очерчивать полигон линией или нет. Для контура полигона используется текущий стиль линии, заданный элементом LineStyle; Пример: '1', '0'.

'BalloonStyle' определяет стиль отображения всплывающего окна с текстовым описанием/комментарием к графическому объекту на карте. Параметры элемента 'BalloonStyle':

bgColor – цвет фона для всплывающего окна с описанием, который задается последовательностью символов bbgrr, где bb – синий, gg – зеленый, rr – красный цвет, например '00ffff';

bgAlpha – параметр прозрачности фона для всплывающего окна, которая задается последовательностью символов от '00' в случае полной прозрачности, до 'ff' в случае непрозрачности;

textColor – цвет текста в окне с описанием, который задается последовательностью символов bbgrr, где bb – синий, gg – зеленый, rr – красный цвет, например '00ffff';

textAlpha – прозрачность текста в окне с описанием, которая задается последовательностью символов от '00' в случае полной прозрачности, до 'ff' в случае непрозрачности;

text – текст, отображаемый в окне с описанием. Пример текста:

```
<![CDATA[<b><font color="#CC0000" size="+3">${name}</b> <br> ${description}]]>
```

Подробное описание элемента 'BalloonStyle' приведено в [5].

'IconStyle' определяет стиль отображения графических маркеров для меток на карте (как на панели со списком отмеченных мест – панель Places, так и на 3D изображении земной поверхности в программе Google Earth). В коде KML: тег <Icon> определяет изображение маркера; тег <scale> определяет коэффициент масштабирования для маркера; цвет, указанный в теге <color> элемента <Icon Style>, будет смешан с цветом изображения маркера. Параметры элемента 'IconStyle':

iColor – цвет маркера, который задается последовательностью символов bbgrr, где bb – синий, gg – зеленый, rr – красный цвет, например '00ffff';

iAlpha – прозрачность маркера, которая задается последовательностью символов от '00' в случае полной прозрачности, до 'ff' в случае непрозрачности;

iScale – параметр масштаба маркера (по умолчанию равен 1);

iHeading – поворот маркера, который задается как в направлении по компасу в градусах; по умолчанию = 0 (Север); диапазон значений от 0 до +-180 градусов;

iHref – HTTP-адрес или путь к локальному файлу, используемый для загрузки маркера; например: <http://maps.google.com/mapfiles/kml/pal3/icon21.png>;

hS – вектор hS={,hS_y,hS_xunits,hS_yunits}, содержащий координаты точки на графическом маркере, которая будет «привязана» к координатам на карте (заданным тегом <Point>). Значения x и y могут быть заданы с использованием трех различных единиц измерения: (1)

пиксели с началом координат в левом нижнем углу значка ("pixels"); (2) доли маркера с началом координат в левом нижнем углу значка ("fraction"); (3) пиксели с началом координат в правом верхнем углу значка ("insetPixels"). При этом единицы измерения для x и y могут быть разными, например, x – пиксели, а y – доли изображения маркера.

hS_x – смещение вдоль оси x;

hS_y – смещение вдоль оси y;

hS_xunits – единицы измерения для hS_x ("fraction", "pixels" или "insetPixels");

hS_yunits – единицы измерения для hS_y ("fraction", "pixels" или "insetPixels").

'LabelStyle' определяет стиль названия объекта (тег <name>) при отображении в средстве 3D просмотра. Параметры элемента 'LabelStyle':

LbColor – цвет текста, который задается последовательностью символов bbggr, где bb – синий, gg – зеленый, rr – красный цвет, например '00ffff';

LbAlpha – которая задается последовательностью символов от '00' в случае полной прозрачности, до 'ff' в случае непрозрачности;

LbScale – размер текста.

Пример использования функции:

```
>> gMapIoKmlSetStyle(fld,'st001',{'LineStyle',4,'r','ff'},{'BalloonStyle','c','0f','b','ff','Area01'});
```

```
>> gMapIoKmlSetStyle(fld,'style02',{'LineStyle',4,'r','ff'},{'PolyStyle','b','5f','1','0'});
```

5. Нарисовать графический маркер (точку)

Функция gMapIoKmlSetPoint(fld,B,L,Z,altitudeMode,PlaceName,StyleId,Descript)

Записывает в файл kml теги и данные – отобразить графический маркер (точку) на карте. В коде функции неявно задается значение extrude=0.

Параметры:

fId – идентификатор kml-файла;

B – широта в градусах (одно число);

L – долгота в градусах (одно число);

Z – координата по оси Z, отображение которой будет зависеть от режима отрисовки высоты (altitudeMode); если значение Z задается пустым массивом [], то значение высоты будет 0 (устанавливается режим clampToGround).

altitudeMode – режим отрисовки высоты Z (тег <coordinates>); возможны следующие режимы:

- (1) clampToGround или clampToSeaFloor – значения высоты принимаются равными нулю,
- (2) relativeToGround или relativeToSeaFloor – относительно поверхности земли или поверхности морского дна, (3) absolute – абсолютная отметка высоты (относительно эллипсоида).

PlaceName – название графического маркера; например: 'Point002', 28;

StyleId – уникальный идентификатор стиля графического маркера (IconStyle, BalloonStyle, LabelStyle), заданный в виде строки или числа, при установке графических стилей при помощи функции gMapIoKmlSetStyle; например: 'Std09', 28, 'RedStyle';

Descript – текстовое описание/комментарий к графическому маркеру для всплывающего окна; примеры: 'seismic survey, area B005', '<![CDATA[Multichannel Seismic data

]]>'

Пример использования функции:

```
>> gMapIoKmlSetPoint(fId,65.1,141,10,2,'Line023','B1','1234567890');
```

6. Нарисовать полилинию

Функция gMapIoKmlSetPolyline(fId,B,L,Z,altitudeMode,PlaceName,LineStyleId,Descript)

Записывает в файл kml теги и данные – отобразить полилинию на карте. В коде функции неявно задаются значения extrude=0, tessellate=1.

Параметры:

fId – идентификатор kml-файла;

B – широта в градусах (одно число);

L – долгота в градусах (одно число);

Z – координата по оси Z, отображение которой будет зависеть от режима отрисовки высоты (altitudeMode); если значение Z задается пустым массивом [], то значение высоты будет 0 (устанавливается режим clampToGround).

altitudeMode – режим отрисовки высоты Z (тег <coordinates>); возможны следующие режимы:

- (1) clampToGround или clampToSeaFloor – значения высоты принимаются равными нулю,
- (2) relativeToGround или relativeToSeaFloor – относительно поверхности земли или поверхности морского дна, (3) absolute – абсолютная отметка высоты (относительно эллипсоида).

PlaceName – название полилинии; например: 'Line002', 28;

LineStyleId – уникальный идентификатор стиля линии (LineStyle, BalloonStyle, LabelStyle), заданный в виде строки или числа, при установке графических стилей при помощи функции gMapIoKmlSetStyle; например: 'Std09', 28, 'RedStyle';

Descript – текстовое описание/комментарий к полилинии для всплывающего окна; примеры: 'seismic survey, area B005', '<![CDATA[Multichannel Seismic data

]]>'

Пример использования функции:

```
>> gMapIoKmlSetPolyline(fId,[65.1 65.2 65.3],[141 141 142],[],1,'Line023','st001','');
```

7. Нарисовать полигон

Функция gMapIoKmlSetPolygon (fId,B,L,Z,altitudeMode,PlaceName,LineStyleId,Descriptor, varargin)

Записывает в файл kml теги и данные – отобразить полигон на карте. В коде функции неявно задаются значения `extrude=0`, `tessellate=1`. Полигон задается при помощи внешней границы и внутренних границ (количество внутренних границ может быть равно 0). В теге `<coordinates>`, координаты полигона должны задаваться с обходом против часовой стрелки, координаты последней точки должны быть равны координатам первой точки. Внутренние границы полигона («дырки») задаются при помощи тега `<LinearRing>`.

Когда полигон «выравнивается» (`extruded`), его границы соединяются с земной поверхностью, образуя дополнительные полигоны, что создает видимость здания или коробки. Для «выравниваемых полигонов» (`Extruded Polygons`) используется элемент `<PolyStyle>` для задания цвета, прозрачности и заливки.

Для полигонов используется «правило правой руки», которое гласит, что, если вы поместите пальцы правой руки в направлении, в котором заданы координаты (в плоскости полигона), ваш большой палец будет указывать в направлении геометрической нормали для многоугольника (в 3D-графике геометрическая нормаль используется при задании освещения и направлена в сторону «от передней грани» полигона). Поскольку программа Google Earth закрашивает только лицевую сторону полигонов, можно достичь необходимых визуальных эффектов (прорисовки и освещения) только тогда, когда координаты указаны в правильном направлении обхода (иначе полигон будет серым).

Параметры:

`fId` – идентификатор kml-файла;

`B` – широта в градусах (одно число);

`L` – долгота в градусах (одно число);

`Z` – координата по оси `Z`, отображение которой будет зависеть от режима отрисовки высоты (`altitudeMode`); если значение `Z` задается пустым массивом `[]`, то значение высоты будет 0 (устанавливается режим `clampToGround`).

`altitudeMode` – режим отрисовки высоты `Z` (тег `<coordinates>`); возможны следующие режимы:

- (1) `clampToGround` или `clampToSeaFloor` – значения высоты принимаются равными нулю,
- (2) `relativeToGround` или `relativeToSeaFloor` – относительно поверхности земли или поверхности морского дна,
- (3) `absolute` – абсолютная отметка высоты (относительно эллипсоида).

`PlaceName` – название полигона; например: 'Polygon002', 28;

`LineStyleId` – уникальный идентификатор стиля для полигона (`PolyStyle`, `LineStyle`, `BalloonStyle`, `LabelStyle`), заданный в виде строки или числа; например: 'Stl09', 28, 'RedStyle';

`Descriptor` – текстовое описание/комментарий к полигону для всплывающего окна; примеры: 'seismic survey, area B005', 'Multichannel Seismic data

]]>'</p></div><div data-bbox="138 938 241 956" data-label="Page-Footer"><p>ge0mllib.com</p></div><div data-bbox="908 938 940 955" data-label="Page-Footer"><p>11</p></div>

varargin – массив ячеек, задающий «внутренние границы» (дыры внутри полигона – *Figure 8.1*); каждая ячейка содержит матрицу из трех векторов-строк (широта, долгота и высота Z -- {B,L,Z}) или двух векторов-строк (широта и долгота -- {B,L}).

Пример использования функции:

```
>> gMapIoKmlSetPolygon(fId,[64 64.5 63.5 64],[143 140 139 143],[[],1,'Polygone01','style02','Polygone  
examlе with style02',{[64 64 63.8],[140.8 140.25 140.3],[[]],[[63.98 63.62 63.78],[140 139.30  
140.19],[[]]);
```

8. Пример использования gMapIoKml (скрипт gMapIoKmlTest)

Пример использования функций библиотеки gMapIoKml записан в скрипте gMapIoKmlTest, который входит в состав библиотеки gMapIoKml. Ниже приводятся текст скрипта, код, записанный в kml-файл этим скриптом, и *Figure 8.1* с отображением содержимого kml-файла в Google Earth.

```
>> fId=gMapIoKmlOpen('c:\temp\112.kml');  
>>  
gMapIoKmlSetStyle(fId,'style01',{ 'LineStyle',4,'r','ff'},{ 'BalloonStyle','c','0f','b','0f','<![CDATA[<b><font  
color="#CC0000" size="-3">${name}</b> <br> ${description}]]>'});  
>> gMapIoKmlSetStyle(fId,'style02',{ 'LineStyle',4,'r','ff'},{ 'PolyStyle','b','5f','1','0'});  
>> gMapIoKmlSetPoint(fId,64.85,140.34,10,2,'Line023','style01','Point examplе with style01');  
>> gMapIoKmlSetPolyline(fId,[65.1 65.2 65.5],[141 140.5 142],[[],1,'Line023','style01','Line examplе with  
style01');  
>> gMapIoKmlSetPolygon(fId,[64 64.5 63.5 64],[143 140 139 143],[[],1,'Polygone01','style02','Polygone  
examlе with style02',{[64 64 63.8],[140.8 140.25 140.3],[[]],[[63.98 63.62 63.78],[140 139.30 140.19],[[]]);  
gMapIoKmlClose(fId);
```

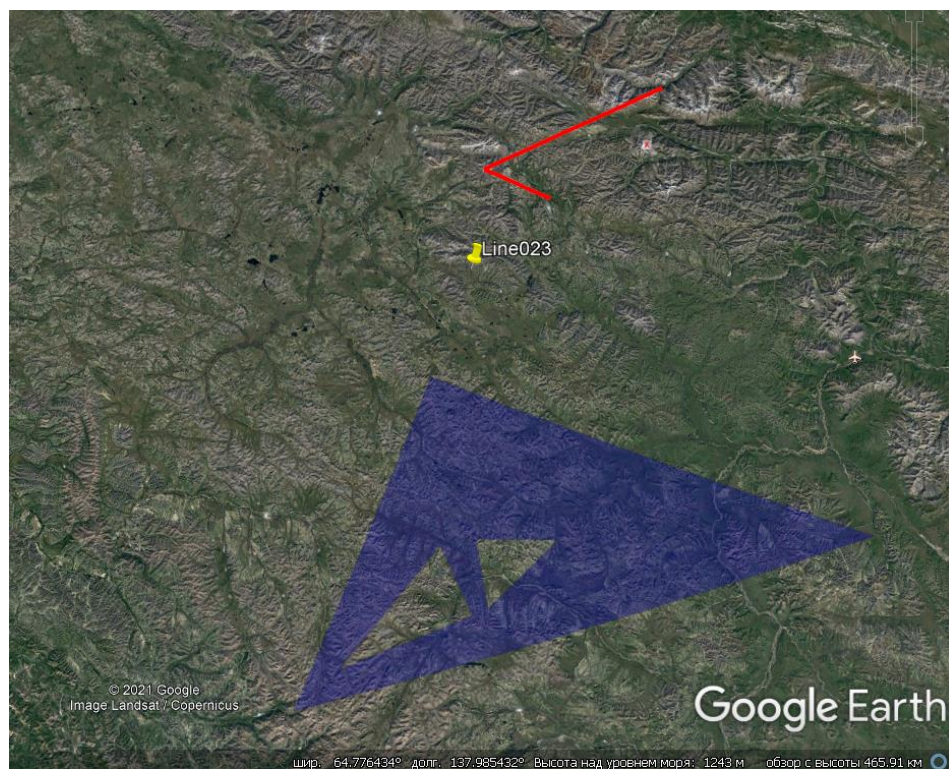


Figure 8.1 Результат выполнения gMapIoKmlTest (отображение файла 112.kml в Google Earth)

Код файла 112.kml, который был создан в результате выполнения скрипта gMapIoKmlTest.

```
<?xml version="1.0" encoding="UTF-8"?>
<kml xmlns="http://www.opengis.net/kml/2.2" xmlns:gx="http://www.google.com/kml/ext/2.2" xmlns:kml="http://www.opengis.net/kml/2.2"
xmlns:atom="http://www.w3.org/2005/Atom">
<Document>
  <name>112.kml</name>
  <Style id="style01">
    <LineStyle>
      <colorMode>normal</colorMode>
      <color>#ff0000ff</color>
      <width>4</width>
      <gx:labelVisibility>0</gx:labelVisibility>
    </LineStyle>
    <BalloonStyle>
      <bgColor>#0fffff00</bgColor>
      <textColor>#0ffff000</textColor>
      <text><![CDATA[<b><font color="#CC0000" size="-3">${name}</b> <br> ${description}]]></text>
      <displayMode>default</displayMode>
    </BalloonStyle>
  </Style>
  <Style id="style02">
    <LineStyle>
      <colorMode>normal</colorMode>
      <color>#ff0000ff</color>
      <width>4</width>
      <gx:labelVisibility>0</gx:labelVisibility>
    </LineStyle>
    <PolyStyle>
      <colorMode>normal</colorMode>
      <color>#5fff0000</color>
      <fill>1</fill>
      <outline>0</outline>
    </PolyStyle>
  </Style>
  <Placemark>
    <name>Line023</name>
    <description>Point example with style01</description>
    <styleUrl>#style01</styleUrl>
    <Point>
      <extrude>0</extrude>
      <altitudeMode>relativeToGround</altitudeMode>
      <gx:altitudeMode>relativeToSeaFloor</gx:altitudeMode>
      <coordinates>
        140.34,64.85,10
      </coordinates>
    </Point>
  </Placemark>
  <Placemark>
    <name>Line023</name>
    <description>Line examle with style01</description>
    <styleUrl>#style01</styleUrl>
    <LineString>
      <extrude>0</extrude>
      <tessellate>1</tessellate>
      <altitudeMode>clampToGround</altitudeMode>
      <gx:altitudeMode>clampToSeaFloor</gx:altitudeMode>
      <coordinates>
        141,65.1,0,140.5,65.2,0,142,65.5,0
      </coordinates>
    </LineString>
  </Placemark>
  <Placemark>
    <name>Polygone01</name>
    <description>Polygone examle with style02</description>
    <styleUrl>#style02</styleUrl>
    <Polygon>
      <extrude>0</extrude>
      <tessellate>1</tessellate>
      <altitudeMode>clampToGround</altitudeMode>
      <gx:altitudeMode>clampToSeaFloor</gx:altitudeMode>
      <outerBoundaryIs>
        <LinearRing>
          <altitudeMode>clampToGround</altitudeMode>
          <gx:altitudeMode>clampToSeaFloor</gx:altitudeMode>
          <coordinates>
            143,64,0,140,64.5,0,139,63.5,0,143,64,0
          </coordinates>
        </LinearRing>
      </outerBoundaryIs>
    </Polygon>
  </Placemark>
</Document>
</kml>
```

```

        </LinearRing>
      </outerBoundaryIs>
      <innerBoundaryIs>
        <LinearRing>
          <altitudeMode>clampToGround</altitudeMode>
          <gx:altitudeMode>clampToSeaFloor</gx:altitudeMode>
          <coordinates>
            140.8,64,0,140.25,64,0,140.3,63.8,0
          </coordinates>
        </LinearRing>
      </innerBoundaryIs>
    </innerBoundaryIs>
    <LinearRing>
      <altitudeMode>clampToGround</altitudeMode>
      <gx:altitudeMode>clampToSeaFloor</gx:altitudeMode>
      <coordinates>
        140,63.98,0,139.3,63.62,0,140.19,63.78,0
      </coordinates>
    </LinearRing>
  </innerBoundaryIs>
</Polygon>
</Placemark>
</Document>
</kml>

```

Список литературы

1. <https://developers.google.com/kml/documentation/kmlreference>
2. https://ge0mllib.com/papers/File_Formats/KML_v1.0_cookbook.pdf
3. https://ge0mllib.com/papers/File_Formats/KML_v2.1_Reference_An_OGC_Best_Practice.pdf
4. <https://www.mathworks.com/matlabcentral/fileexchange/34006-kml-line-plot>
5. <https://developers.google.com/kml/documentation/kmlreference#balloonstyle>