**Create a Point**

from osgeo import ogr

point = ogr.Geometry(ogr.wkbPoint)

point.AddPoint(1198054.34, 648493.09)

print point.ExportToWkt()

**Create a LineString**

from osgeo import ogr

line = ogr.Geometry(ogr.wkbLineString)

line.AddPoint(1116651.439379124, 637392.6969887456)

line.AddPoint(1188804.0108498496, 652655.7409537067)

line.AddPoint(1226730.3625203592, 634155.0816022386)

line.AddPoint(1281307.30760719, 636467.6640211721)

print line.ExportToWkt()

**Create a Polygon**

from osgeo import ogr

# Create ring

ring = ogr.Geometry(ogr.wkbLinearRing)

ring.AddPoint(1179091.1646903288, 712782.8838459781)

ring.AddPoint(1161053.0218226474, 667456.2684348812)

ring.AddPoint(1214704.933941905, 641092.8288590391)

ring.AddPoint(1228580.428455506, 682719.3123998424)

ring.AddPoint(1218405.0658121984, 721108.1805541387)

ring.AddPoint(1179091.1646903288, 712782.8838459781)

# Create polygon

poly = ogr.Geometry(ogr.wkbPolygon)

poly.AddGeometry(ring)

print poly.ExportToWkt()

**Create a Polygon with holes**

from osgeo import ogr

# Create outer ring

outRing = ogr.Geometry(ogr.wkbLinearRing)

outRing.AddPoint(1154115.274565847, 686419.4442701361)

outRing.AddPoint(1154115.274565847, 653118.2574374934)

outRing.AddPoint(1165678.1866605144, 653118.2574374934)

outRing.AddPoint(1165678.1866605144, 686419.4442701361)

outRing.AddPoint(1154115.274565847, 686419.4442701361)

# Create inner ring

innerRing = ogr.Geometry(ogr.wkbLinearRing)

innerRing.AddPoint(1149490.1097279799, 691044.6091080031)

innerRing.AddPoint(1149490.1097279799, 648030.5761158396)

innerRing.AddPoint(1191579.1097525698, 648030.5761158396)

innerRing.AddPoint(1191579.1097525698, 691044.6091080031)

innerRing.AddPoint(1149490.1097279799, 691044.6091080031)

# Create polygon

poly = ogr.Geometry(ogr.wkbPolygon)

poly.AddGeometry(outRing)

poly.AddGeometry(innerRing)

print poly.ExportToWkt()

**Create a MultiPoint**

from osgeo import ogr

multipoint = ogr.Geometry(ogr.wkbMultiPoint)

point1 = ogr.Geometry(ogr.wkbPoint)

point1.AddPoint(1251243.7361610543, 598078.7958668759)

multipoint.AddGeometry(point1)

point2 = ogr.Geometry(ogr.wkbPoint)

point2.AddPoint(1240605.8570339603, 601778.9277371694)

multipoint.AddGeometry(point2)

point3 = ogr.Geometry(ogr.wkbPoint)

point3.AddPoint(1250318.7031934808, 606404.0925750365)

multipoint.AddGeometry(point3)

print multipoint.ExportToWkt()

**Create a MultiLineString**

from osgeo import ogr

multiline = ogr.Geometry(ogr.wkbMultiLineString)

line1 = ogr.Geometry(ogr.wkbLineString)line1.AddPoint(1214242.4174581182,617041.9717021306)line1.AddPoint(1234593.142744733, 629529.9167643716)multiline.AddGeometry(line1)

line1 = ogr.Geometry(ogr.wkbLineString)line1.AddPoint(1184641.3624957693,626754.8178616514)line1.AddPoint(1219792.6152635587, 606866.6090588232)multiline.AddGeometry(line1)

print multiline.ExportToWkt()

**Create a MultiPolygon**

from osgeo import ogr

multipolygon = ogr.Geometry(ogr.wkbMultiPolygon)

# Create ring #1ring1 = ogr.Geometry(ogr.wkbLinearRing)ring1.AddPoint(1204067.0548148106,634617.5980860253)ring1.AddPoint(1204067.0548148106,620742.1035724243)ring1.AddPoint(1215167.4504256917,620742.1035724243)ring1.AddPoint(1215167.4504256917,634617.5980860253)ring1.AddPoint(1204067.0548148106, 634617.5980860253)

# Create polygon #1poly1 =ogr.Geometry(ogr.wkbPolygon)poly1.AddGeometry(ring1)multipolygon.AddGeometry(poly1)

# Create ring #2ring2 = ogr.Geometry(ogr.wkbLinearRing)ring2.AddPoint(1179553.6811741155,647105.5431482664)ring2.AddPoint(1179553.6811741155,626292.3013778647)ring2.AddPoint(1194354.20865529,626292.3013778647)ring2.AddPoint(1194354.20865529,647105.5431482664)ring2.AddPoint(1179553.6811741155, 647105.5431482664)

# Create polygon #2poly2 =ogr.Geometry(ogr.wkbPolygon)poly2.AddGeometry(ring2)multipolygon.AddGeometry(poly2)

print multipolygon.ExportToWkt()

**Create a GeometryCollection**

from osgeo import ogr

# Create a geometry collectiongeomcol = ogr.Geometry(ogr.wkbGeometryCollection)

# Add a pointpoint = ogr.Geometry(ogr.wkbPoint)point.AddPoint(-122.23, 47.09)geomcol.AddGeometry(point)

# Add a lineline = ogr.Geometry(ogr.wkbLineString)line.AddPoint(-122.60, 47.14)line.AddPoint(-122.48,47.23)geomcol.AddGeometry(line)

print geomcol.ExportToWkt()

**Create Geometry from WKT**

from osgeo import ogr

wkt = "POINT (1120351.5712494177 741921.4223245403)"point = ogr.CreateGeometryFromWkt(wkt)print"%d,%d" % (point.GetX(), point.GetY())

**Create Geometry from GeoJSON**

from osgeo import ogr

geojson = """{"type":"Point","coordinates":[108420.33,753808.59]}"""point =ogr.CreateGeometryFromJson(geojson)print "%d,%d" % (point.GetX(), point.GetY())

**Create Geometry from GML**

from osgeo import ogr

gml = """<gml:Point xmlns:gml="http://www.opengis.net/gml"><gml:coordinates>108420.33,753808.59</gml:coordinates></gml:Point>"""point =ogr.CreateGeometryFromGML(gml)print "%d,%d" % (point.GetX(), point.GetY())

**Create Geometry from WKB**

from osgeo import ogrfrom base64 import b64decode

wkb = b64decode("AIAAAAFBMkfmVwo9cUEjylouFHrhAAAAAAAAAAA=")point =ogr.CreateGeometryFromWkb(wkb)print "%d,%d" % (point.GetX(), point.GetY())

**Count Points in a Geometry**

from osgeo import ogr

wkt = "LINESTRING (1181866.263593049 615654.4222507705, 1205917.1207499576 623979.7189589312, 1227192.8790041457 643405.4112779726, 1224880.2965852122 665143.6860159477)"geom =ogr.CreateGeometryFromWkt(wkt)print "Geometry has %i points" % (geom.GetPointCount())

**Count Geometries in a Geometry**

from osgeo import ogr

wkt = "MULTIPOINT (1181866.263593049 615654.4222507705, 1205917.1207499576 623979.7189589312, 1227192.8790041457 643405.4112779726, 1224880.2965852122 665143.6860159477)"geom =ogr.CreateGeometryFromWkt(wkt)print "Geometry has %i geometries" % (geom.GetGeometryCount())

**Iterate over Geometries in a Geometry**

from osgeo import ogr

wkt = "MULTIPOINT (1181866.263593049 615654.4222507705, 1205917.1207499576 623979.7189589312, 1227192.8790041457 643405.4112779726, 1224880.2965852122 665143.6860159477)"geom =ogr.CreateGeometryFromWkt(wkt)for i in range(0, geom.GetGeometryCount()):

g = geom.GetGeometryRef(i)

print "%i). %s" %(i, g.ExportToWkt())

**Iterate over Points in a Geometry**

from osgeo import ogr

wkt = "LINESTRING (1181866.263593049 615654.4222507705, 1205917.1207499576 623979.7189589312, 1227192.8790041457 643405.4112779726, 1224880.2965852122 665143.6860159477)"geom =ogr.CreateGeometryFromWkt(wkt)for i in range(0, geom.GetPointCount()):

# GetPoint returns a tuple not a Geometry

pt = geom.GetPoint(i)

print "%i). POINT (%d %d)" %(i, pt[0], pt[1])

**Buffer a Geometry**

from osgeo import ogr

wkt = "POINT (1198054.34 648493.09)"pt = ogr.CreateGeometryFromWkt(wkt)bufferDistance = 500poly =pt.Buffer(bufferDistance)print "%s buffered by %d is %s" % (pt.ExportToWkt(), bufferDistance,poly.ExportToWkt())

**Calculate Envelope of a Geometry**

from osgeo import ogr

wkt = "LINESTRING (1181866.263593049 615654.4222507705, 1205917.1207499576 623979.7189589312, 1227192.8790041457 643405.4112779726, 1224880.2965852122 665143.6860159477)"geom =ogr.CreateGeometryFromWkt(wkt)# Get Envelope returns a tuple (minX, maxX, minY, maxY)env =geom.GetEnvelope()print "minX: %d, minY: %d, maxX: %d, maxY: %d" %(env[0],env[2],env[1],env[3])

**Calculate the Area of a Geometry**

from osgeo import ogr

wkt = "POLYGON ((1162440.5712740074 672081.4332727483, 1162440.5712740074 647105.5431482664, 1195279.2416228633 647105.5431482664, 1195279.2416228633 672081.4332727483, 1162440.5712740074 672081.4332727483))"poly = ogr.CreateGeometryFromWkt(wkt)print "Area = %d" %poly.GetArea()

**Calculate the Length of a Geometry**

from osgeo import ogr

wkt = "LINESTRING (1181866.263593049 615654.4222507705, 1205917.1207499576 623979.7189589312, 1227192.8790041457 643405.4112779726, 1224880.2965852122 665143.6860159477)"geom =ogr.CreateGeometryFromWkt(wkt)print "Length = %d" % geom.Length()

**Get the geometry type (as a string) from a Geometry**

from osgeo import ogr

wkts = [

"POINT (1198054.34 648493.09)",

"LINESTRING (1181866.263593049 615654.4222507705, 1205917.1207499576 623979.7189589312, 1227192.8790041457 643405.4112779726, 1224880.2965852122 665143.6860159477)",

"POLYGON ((1162440.5712740074 672081.4332727483, 1162440.5712740074 647105.5431482664, 1195279.2416228633 647105.5431482664, 1195279.2416228633 672081.4332727483, 1162440.5712740074 672081.4332727483))"]

for wkt in wkts:

geom = ogr.CreateGeometryFromWkt(wkt)

print geom.GetGeometryName()

**Calculate intersection between two Geometries**

from osgeo import ogr

wkt1 = "POLYGON ((1208064.271243039 624154.6783778917, 1208064.271243039 601260.9785661874, 1231345.9998651114 601260.9785661874, 1231345.9998651114 624154.6783778917, 1208064.271243039 624154.6783778917))"wkt2 = "POLYGON ((1199915.6662253144 633079.3410163528, 1199915.6662253144 614453.958118695, 1219317.1067437078 614453.958118695, 1219317.1067437078 633079.3410163528, 1199915.6662253144 633079.3410163528)))"

poly1 = ogr.CreateGeometryFromWkt(wkt1)poly2 = ogr.CreateGeometryFromWkt(wkt2)

intersection = poly1.Intersection(poly2)

print intersection.ExportToWkt()

**Calculate union between two Geometries**

from osgeo import ogr

wkt1 = "POLYGON ((1208064.271243039 624154.6783778917, 1208064.271243039 601260.9785661874, 1231345.9998651114 601260.9785661874, 1231345.9998651114 624154.6783778917, 1208064.271243039 624154.6783778917))"wkt2 = "POLYGON ((1199915.6662253144 633079.3410163528, 1199915.6662253144 614453.958118695, 1219317.1067437078 614453.958118695, 1219317.1067437078 633079.3410163528, 1199915.6662253144 633079.3410163528)))"

poly1 = ogr.CreateGeometryFromWkt(wkt1)poly2 = ogr.CreateGeometryFromWkt(wkt2)

union = poly1.Union(poly2)

print poly1print poly2print union.ExportToWkt()

**Write Geometry to GeoJSON**

There are two options to create a GeoJSON from a geometry.

You can either create a new GeoJSON file or simply export the geometry to Json and print it. Both options are explained below.

from osgeo import ogr

# Create test polygonring = ogr.Geometry(ogr.wkbLinearRing)ring.AddPoint(1179091.1646903288,712782.8838459781)ring.AddPoint(1161053.0218226474,667456.2684348812)ring.AddPoint(1214704.933941905,641092.8288590391)ring.AddPoint(1228580.428455506,682719.3123998424)ring.AddPoint(1218405.0658121984,721108.1805541387)ring.AddPoint(1179091.1646903288, 712782.8838459781)poly =ogr.Geometry(ogr.wkbPolygon)poly.AddGeometry(ring)

# Create the output DriveroutDriver = ogr.GetDriverByName('GeoJSON')

# Create the output GeoJSONoutDataSource = outDriver.CreateDataSource('test.geojson')outLayer =outDataSource.CreateLayer('test.geojson', geom\_type=ogr.wkbPolygon )

# Get the output Layer's Feature DefinitionfeatureDefn = outLayer.GetLayerDefn()

# create a new featureoutFeature = ogr.Feature(featureDefn)

# Set new geometryoutFeature.SetGeometry(poly)

# Add new feature to output LayeroutLayer.CreateFeature(outFeature)

# destroy the featureoutFeature.Destroy

# Close DataSourcesoutDataSource.Destroy()

from osgeo import ogr

# Create test polygonring = ogr.Geometry(ogr.wkbLinearRing)ring.AddPoint(1179091.1646903288,712782.8838459781)ring.AddPoint(1161053.0218226474,667456.2684348812)ring.AddPoint(1214704.933941905,641092.8288590391)ring.AddPoint(1228580.428455506,682719.3123998424)ring.AddPoint(1218405.0658121984,721108.1805541387)ring.AddPoint(1179091.1646903288, 712782.8838459781)poly =ogr.Geometry(ogr.wkbPolygon)poly.AddGeometry(ring)

geojson = poly.ExportToJSON()print geojson

**Write Geometry to WKT**

from osgeo import ogr

# Create test polygonring = ogr.Geometry(ogr.wkbLinearRing)ring.AddPoint(1179091.1646903288,712782.8838459781)ring.AddPoint(1161053.0218226474,667456.2684348812)ring.AddPoint(1214704.933941905,641092.8288590391)ring.AddPoint(1228580.428455506,682719.3123998424)ring.AddPoint(1218405.0658121984,721108.1805541387)ring.AddPoint(1179091.1646903288, 712782.8838459781)geom\_poly =ogr.Geometry(ogr.wkbPolygon)geom\_poly.AddGeometry(ring)

# Export geometry to WKTwkt = geom\_poly.ExportToWkt()print wkt

**Write Geometry to KML**

from osgeo import ogr

# Create test polygonring = ogr.Geometry(ogr.wkbLinearRing)ring.AddPoint(1179091.1646903288,712782.8838459781)ring.AddPoint(1161053.0218226474,667456.2684348812)ring.AddPoint(1214704.933941905,641092.8288590391)ring.AddPoint(1228580.428455506,682719.3123998424)ring.AddPoint(1218405.0658121984,721108.1805541387)ring.AddPoint(1179091.1646903288, 712782.8838459781)geom\_poly =ogr.Geometry(ogr.wkbPolygon)geom\_poly.AddGeometry(ring)

kml = geom\_poly.ExportToKML()print kml

**Write Geometry to WKB**

from osgeo import ogr

# Create test polygonring = ogr.Geometry(ogr.wkbLinearRing)ring.AddPoint(1179091.1646903288,712782.8838459781)ring.AddPoint(1161053.0218226474,667456.2684348812)ring.AddPoint(1214704.933941905,641092.8288590391)ring.AddPoint(1228580.428455506,682719.3123998424)ring.AddPoint(1218405.0658121984,721108.1805541387)ring.AddPoint(1179091.1646903288, 712782.8838459781)geom\_poly =ogr.Geometry(ogr.wkbPolygon)geom\_poly.AddGeometry(ring)

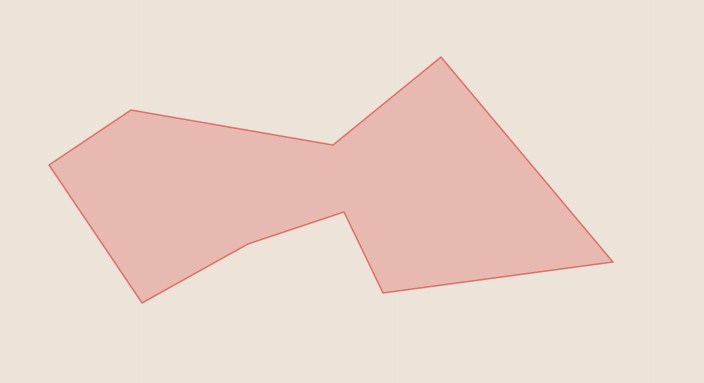
# Export geometry to WKTwkb = geom\_poly.ExportToWkb()print wkb

**Quarter polygon and create centroids**

This recipe quarters a polygon and creates the centroid of the four quarters

import ogr

# Given a test polygonpoly\_Wkt= "POLYGON((-107.42631019589980212 40.11971708125970082,-107.42455436683293613 40.12061219666851741,-107.42020981542387403 40.12004414402532859,-107.41789122063043749 40.12149008687303819,-107.41419947746419439 40.11811617239460048,-107.41915181585792993 40.11761695654455906,-107.41998470913324581 40.11894245264452508,-107.42203317637793702 40.1184088144647788,-107.42430674991324224 40.1174448122981957,-107.42430674991324224 40.1174448122981957,-107.42631019589980212 40.11971708125970082))"geom\_poly = ogr.CreateGeometryFromWkt(poly\_Wkt)



# Create 4 square polygonsgeom\_poly\_envelope = geom\_poly.GetEnvelope()minX =geom\_poly\_envelope[0]minY = geom\_poly\_envelope[2]maxX = geom\_poly\_envelope[1]maxY =geom\_poly\_envelope[3]

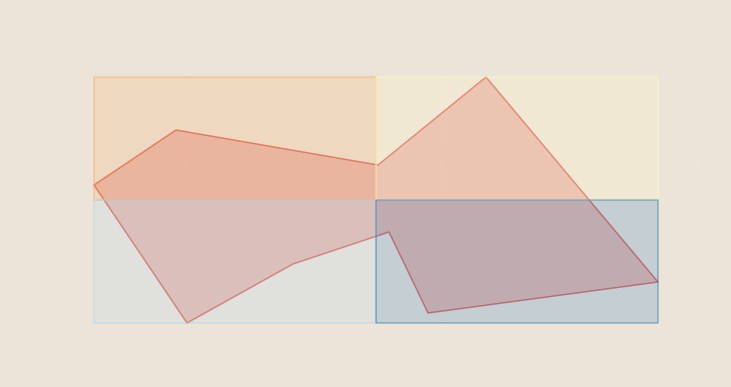
'''coord0----coord1----coord2|           |           |coord3----coord4----coord5|           |           |coord6----coord7----coord8'''coord0 = minX, maxYcoord1 = minX+(maxX-minX)/2, maxYcoord2 = maxX, maxYcoord3 = minX,minY+(maxY-minY)/2coord4 = minX+(maxX-minX)/2, minY+(maxY-minY)/2coord5 = maxX,minY+(maxY-minY)/2coord6 = minX, minYcoord7 = minX+(maxX-minX)/2, minYcoord8 = maxX, minY

ringTopLeft =ogr.Geometry(ogr.wkbLinearRing)ringTopLeft.AddPoint\_2D(\*coord0)ringTopLeft.AddPoint\_2D(\*coord1)ringTopLeft.AddPoint\_2D(\*coord4)ringTopLeft.AddPoint\_2D(\*coord3)ringTopLeft.AddPoint\_2D(\*coord0)polyTopLeft= ogr.Geometry(ogr.wkbPolygon)polyTopLeft.AddGeometry(ringTopLeft)

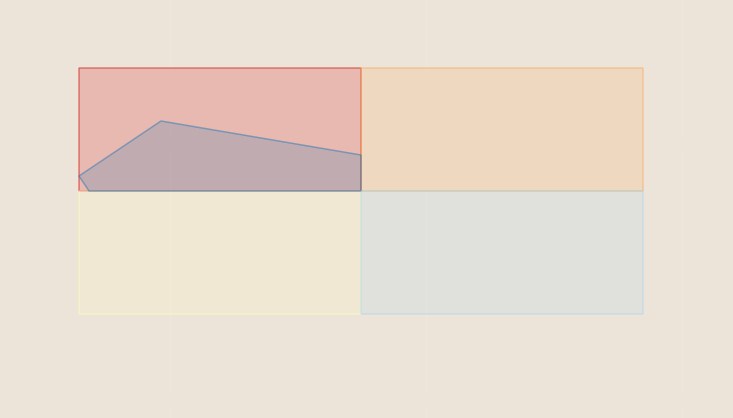
ringTopRight =ogr.Geometry(ogr.wkbLinearRing)ringTopRight.AddPoint\_2D(\*coord1)ringTopRight.AddPoint\_2D(\*coord2)ringTopRight.AddPoint\_2D(\*coord5)ringTopRight.AddPoint\_2D(\*coord4)ringTopRight.AddPoint\_2D(\*coord1)polyTopRight= ogr.Geometry(ogr.wkbPolygon)polyTopRight.AddGeometry(ringTopRight)

ringBottomLeft =ogr.Geometry(ogr.wkbLinearRing)ringBottomLeft.AddPoint\_2D(\*coord3)ringBottomLeft.AddPoint\_2D(\*coord4)ringBottomLeft.AddPoint\_2D(\*coord7)ringBottomLeft.AddPoint\_2D(\*coord6)ringBottomLeft.AddPoint\_2D(\*coord3)polyBottomLeft= ogr.Geometry(ogr.wkbPolygon)polyBottomLeft.AddGeometry(ringBottomLeft)

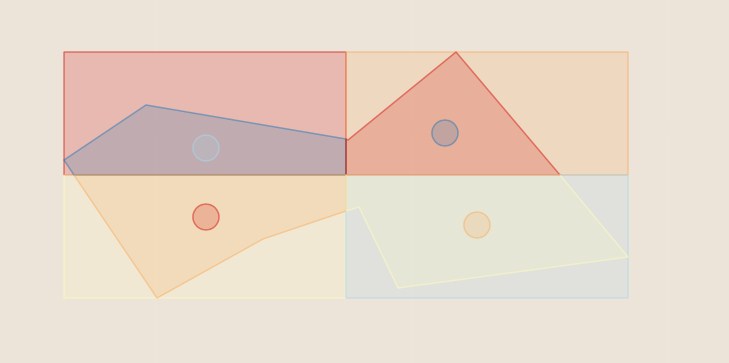
ringBottomRight =ogr.Geometry(ogr.wkbLinearRing)ringBottomRight.AddPoint\_2D(\*coord4)ringBottomRight.AddPoint\_2D(\*coord5)ringBottomRight.AddPoint\_2D(\*coord8)ringBottomRight.AddPoint\_2D(\*coord7)ringBottomRight.AddPoint\_2D(\*coord4)polyBottomRight= ogr.Geometry(ogr.wkbPolygon)polyBottomRight.AddGeometry(ringBottomRight)



# Intersect 4 squares polygons with test polygonquaterPolyTopLeft =polyTopLeft.Intersection(geom\_poly)quaterPolyTopRight =polyTopRight.Intersection(geom\_poly)quaterPolyBottomLeft =polyBottomLeft.Intersection(geom\_poly)quaterPolyBottomRight = polyBottomRight.Intersection(geom\_poly)



# Create centroids of each intersected polygoncentroidTopLeft = quaterPolyTopLeft.Centroid()centroidTopRight= quaterPolyTopRight.Centroid()centroidBottomLeft = quaterPolyBottomLeft.Centroid()centroidBottomRight =quaterPolyBottomRight.Centroid()



来源： <<http://pcjericks.github.io/py-gdalogr-cookbook/geometry.html>>