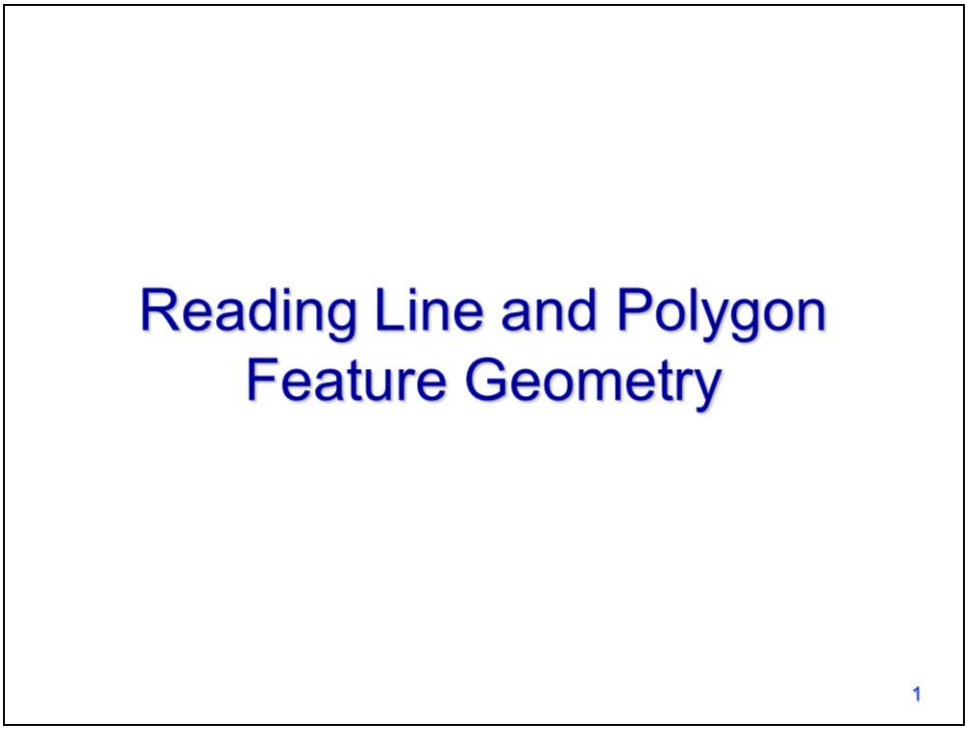
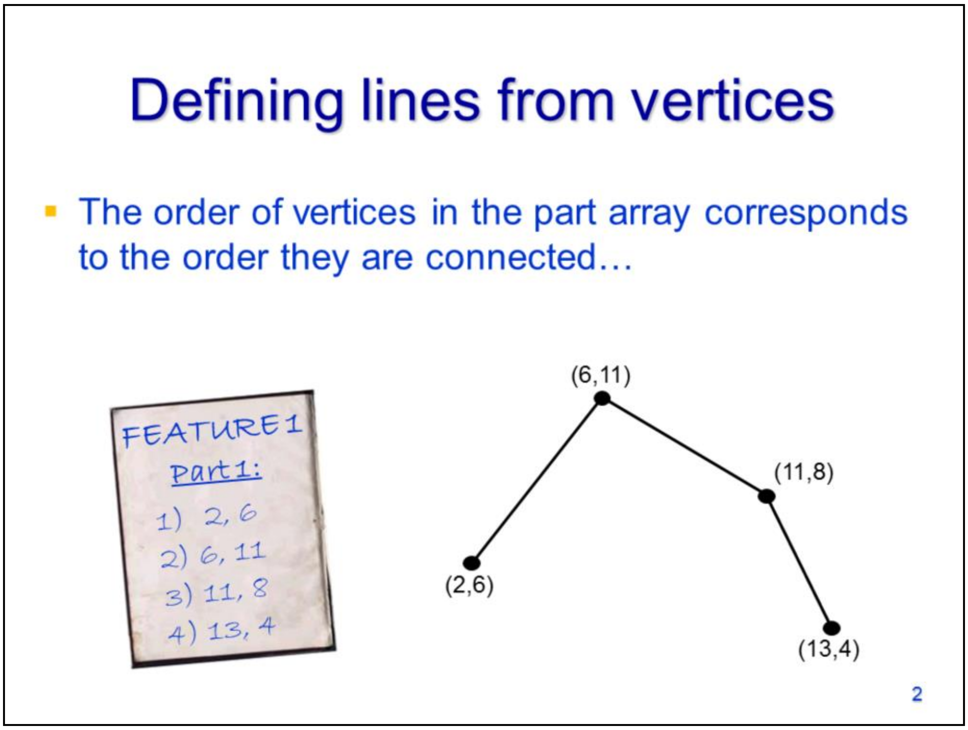


Reading Line and Polygon Feature Geometry

1

This video will discuss how to read geometry information for line and polygon features.



Defining lines from vertices

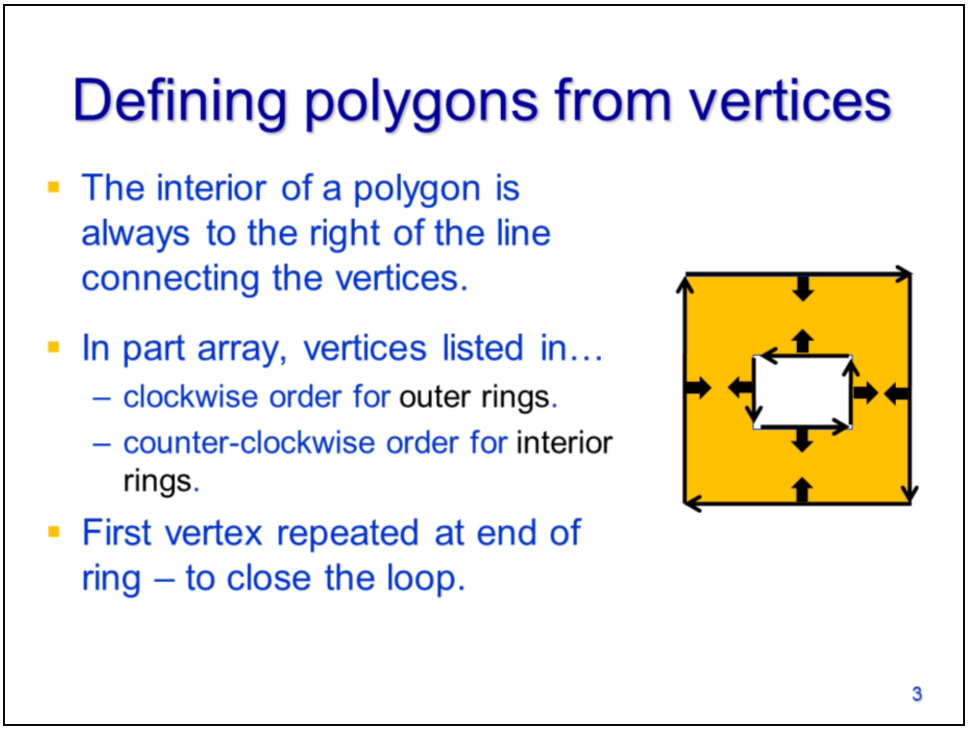
- The order of vertices in the part array corresponds to the order they are connected…

FEATURE1

Part1:

1) 2, 6
2) 6, 11
3) 11, 8
4) 13, 4

For line features, the order of the vertices in the part array corresponds to the order that they are connected.



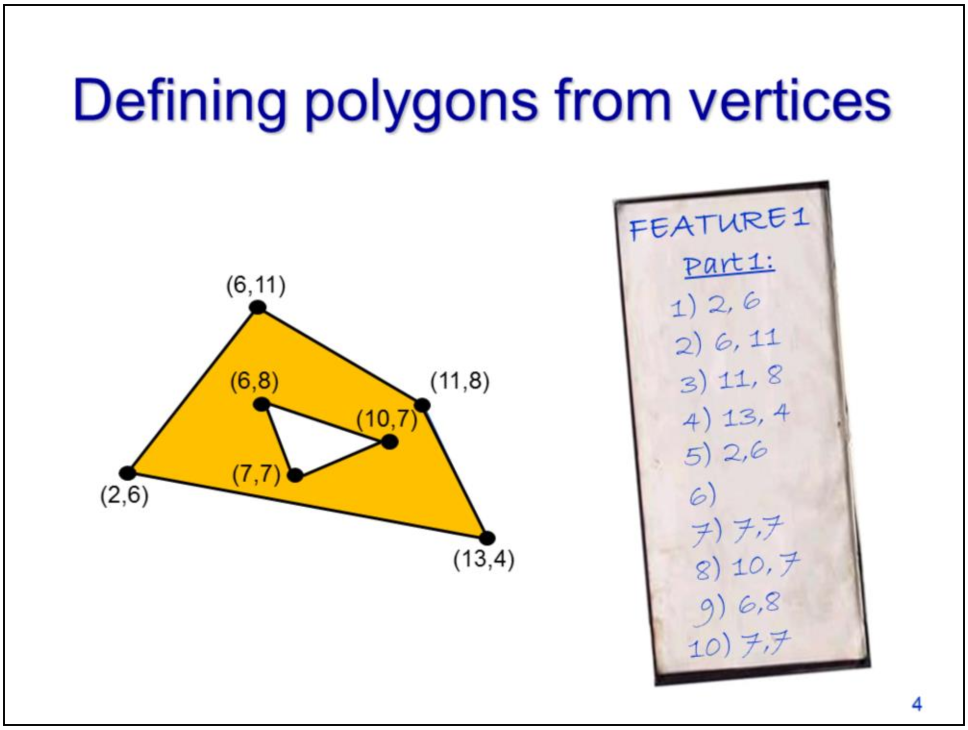
Defining polygons from vertices

- The interior of a polygon is always to the right of the line connecting the vertices.
- In part array, vertices listed in…
  - clockwise order for outer rings.
  - counter-clockwise order for interior rings.
- First vertex repeated at end of ring – to close the loop.

Polygon features delineate an area – the interior of the polygon is always located to the right of the line connecting the feature vertices.

Based on this rule, the order that vertices are listed in a part array should be clockwise for the outer ring and counter-clockwise for an inner ring. This illustration shows that with these orders, the interior of the polygon will always be to the right of the lines formed by the vertices.

To close each ring in the polygon, the first vertex is repeated at the end of the array..



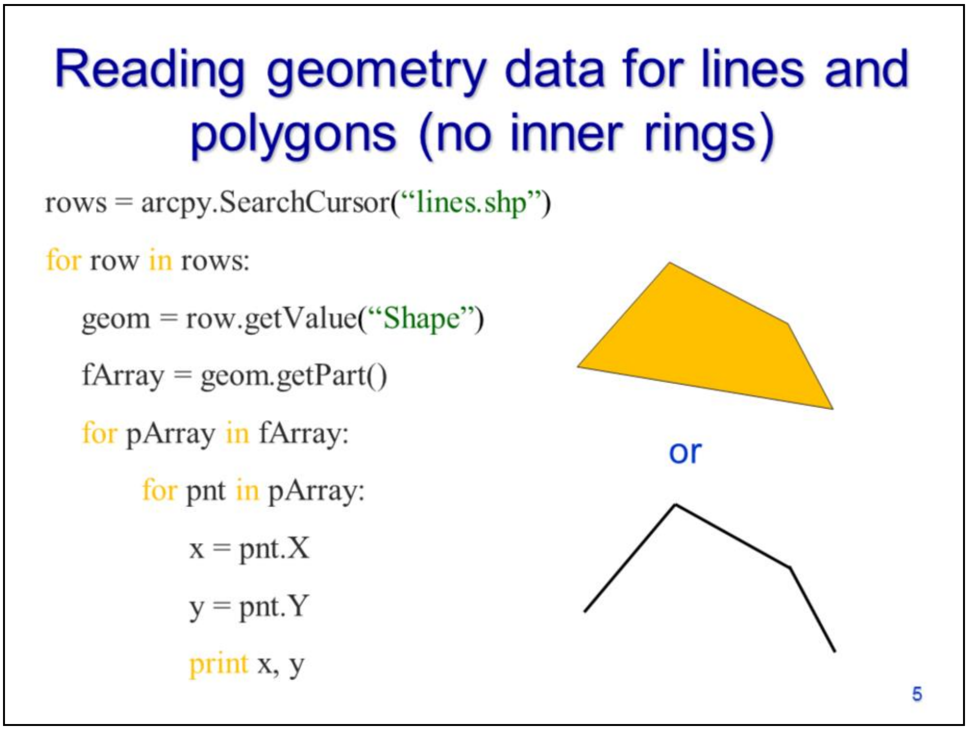
Defining polygons from vertices

FEATURE 1

Part 1:

1) 2, 6
2) 6, 11
3) 11, 8
4) 13, 4
5) 2,6
6)
7) 7,7
8) 10, 7
9) 6,8
10) 7,7

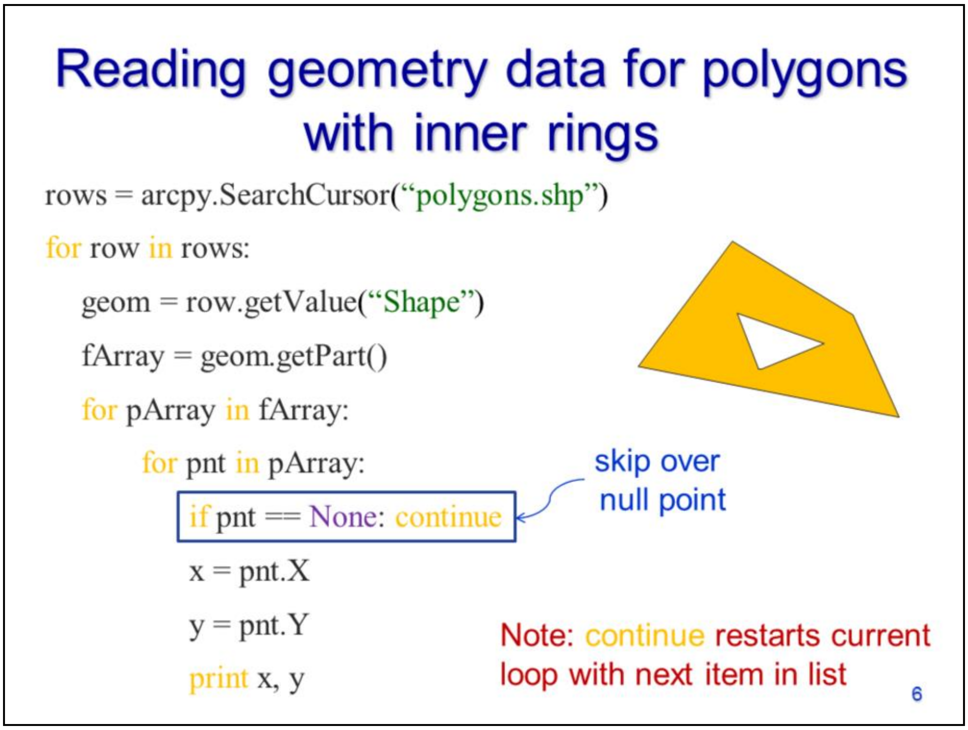
When reading vertices from a polygon’s part array, notice that the vertices of the outer ring are listed in a clockwise order. In the part array, a null point is used to separate the outer ring from (if an inner ring is present). Notice that the vertices of the inner ring are specified in a counter-clockwise order. Also note that for both the outer and inner rings, the first and last vertices in the ring are the same.



Reading geometry data for lines and polygons (no inner rings)

```
rows = arcpy.SearchCursor(“lines.shp”)
for row in rows:
    geom = row.getValue(“Shape”)
    fArray = geom.getPart()
    for pArray in fArray:
        for pnt in pArray:
            x = pnt.X
            y = pnt.Y
            print x, y
```

This slide shows an example script that reads all vertices from either a line or polygon feature. In this script example, The geometry object’s **getPart** method is used to get the feature array (fArray). A for loop is used to retrieve each part array (i.e. pArray). For each part array, a for loop is used to retrieve each point object. The X and Y coordinates are retrieved from each point object. Note that this script will not detect the inner rings for polygons, if they are present. For a polygon with an inner ring, the null point that separates the outer and inner rings will stop the for loop that iterates through the point objects in the part array.



Reading geometry data for polygons with inner rings

```
rows = arcpy.SearchCursor(“polygons.shp”)
for row in rows:
    geom = row.getValue(“Shape”)
    fArray = geom.getPart()
    for pArray in fArray:
        for pnt in pArray:
            if pnt == None: continue
            x = pnt.X
            y = pnt.Y
            print x, y
```

skip over null point

Note: continue restarts current loop with next item in list

This script example has a slight modification over the script on the previous slide which will allow the script to retrieve the vertices of inner rings if they are present in a polygon. Note that this script will also work for lines and polygons that have no inner ring.

The modification added a single line that checks if a point object, retrieved from the part array, has a null value. If the point has a null value, then the **continue** command is used to skip to the next point in the part array. Note