**🧭 GIS Programming – Class Notes & Announcements**

**📅 Date:** April 8, 2025  
**👨‍🏫 Instructor:** Mr. Erik Bushland  
**🕔 Class Time:** 5:00 – 7:20 PM

**🗒️ Class Notes – April 8, 2025**

**✅ Administrative Reminders**

* **Zeros will be entered** for any missing assignments **up through the midterm** by Thursday.
  + This is to give a more accurate picture of your actual grade.
  + Zeros **will not** be entered yet for work assigned **after the midterm** (unless very late).
* **April 17 is the last day to drop a class.**
  + If you're far behind or unable to catch up, consider whether dropping is better for your GPA.

Example: If you’ve only done the midterm and scored a 90%, your grade may show an A—but if you’re missing everything else, the zeros will bring it down.

**🌐 Professional Development Resource**

* **SCAUG (South Central Arc User Group)** annual meeting is next week in Frisco.
  + Co-founded by instructor Scott Sires.
  + Covers GIS professionals in **TX, OK, LA, AR**, and possibly **NM**.
  + While attendance is pricey ($350–$400 for non-students), the website occasionally posts **job listings**.
  + Visit their site occasionally if you're seeking jobs or internships in the region.

**📅 Course Progress and Week Reference**

* This is **Week 12** (counting Spring Break as a week).
* **Current focus:** Finishing **Exercise 8** (Manipulating Spatial Data).
* **Next focus:** Starting **Chapter 9 – Working with Geometries**.

**📁 Portfolio Projects – Important Info**

* The course only officially sets aside **one week** for your three required **portfolio items**.
* To avoid being overwhelmed, the **portfolio requirements** will be made **available early (Week 16 unlocked today)**.
  + So you can begin working on your own time.
* Otherwise, you'd only get two class periods (Tues/Thurs) to work on it.
* **Due Date:** One week after portfolio week begins — same as normal assignment schedule.
* **Final Exam:**
  + Will be available the **Friday before finals week (May 9)**.
  + Due by **May 15**.
  + Will be **same format as the midterm**.

**📅 Portfolio Time Estimate Adjustment**

* Originally, students were expected to spend **6+ hours** per portfolio item across 2 weeks.
* Since the timeline has changed to **just 1 week**, the professor plans to **adjust that expectation** down (approx. **3+ hours per item** minimum now).
* Class time allocated = approx. **4 hours and 40 minutes total** across that week.

**❓ Suggested Questions to Ask (Based on Today’s Class)**

Here are a few you can ask the professor during class, office hours, or by email/MS Teams — especially relevant to your goals:

**🧠 Class/Career Understanding**

1. **“Can I use job listings or real-world problems from SCAUG as inspiration for one of my portfolio items?”**
2. **“Are there examples of past portfolio projects you’d recommend reviewing for ideas?”**
3. **“If I want to present my portfolio as part of a grad school or job application, what sections or tools should I emphasize?”**
4. **“Would you be willing to give feedback on a draft version of a portfolio item before the final submission?”**

**💼 Academic/Career Application**

1. **“Can my project focus on public health or disaster mapping since I volunteer with the Red Cross and am studying health data science?”**
2. **“Are we allowed to use external data sources like CDC or health GIS datasets for our portfolio?”**
3. **“How much Python customization should be shown in a portfolio piece to demonstrate proficiency?”**
4. **“Could one of my existing or previous class projects be adapted into a portfolio item if I revise it using ArcPy?”**

**📌 Portfolio Importance**

* A local GIS manager **reposted a job** because **none of the applicants had a portfolio**.
* Portfolios are **highly valued in the GIS job market**—they help you **stand out**.
* Your professor shared from personal experience:
  + He was the only one to bring a **physical portfolio** to job interviews.
  + That helped him rank as a top candidate (second place in two out of three interviews).
  + Even if you don’t remember *everything*, showing you’ve done the work is powerful.
  + The portfolio **proves working knowledge** and initiative.

**🎯 Portfolio Assignment Details**

* **Week 16** is your designated portfolio work week.
* You must complete **3 portfolio items**, each worth **100 points** (300 total).
* Portfolio = 17% of your **total course grade (1800 points)**.

**📁 What Counts as a Portfolio Item?**

You have **creative flexibility** to:

* Adapt any lesson, tutorial, or assignment.
* Turn it into a **unique product** that shows your:
  + Skills
  + Knowledge
  + Abilities
* Options include:
  + Written documents explaining a **methodology or process**
  + A **web map**, **web app**, or **dashboard**
  + Any **virtual product** or **GIS tool**

**🧠 How to Think About Portfolio Work**

* Consider your **syllabus and learning outcomes**—choose projects that reflect key concepts.
* Imagine your project from a **hiring manager’s perspective**:
  + If they’re engaged, impressed, and asking questions about your work—it’s a success.
* The work doesn’t have to be perfect—just **demonstrate what you’ve learned and can do**.

**❓ Suggested Questions to Ask (For Clarity & Career Goals)**

Here are some questions you can ask Professor Bushland (in class, office hours, or via Teams/email):

**📘 Academic/Project Guidance**

1. **“Would you be open to reviewing a rough draft of one of my portfolio items for feedback?”**
2. **“Can I combine GIS with other disciplines (like public health, Red Cross disaster relief, or AI) in my portfolio projects?”**
3. **“Is it okay if my project builds on or improves a previous assignment from class or another course?”**
4. **“Can I use non-class datasets, like open data from FEMA or CDC, for my portfolio items?”**

**💼 Job Prep Focus**

1. **“What types of portfolio items have helped your past students land internships or jobs?”**
2. **“Would creating a web app with ArcGIS Experience Builder stand out to hiring managers more than a static map?”**
3. **“Should I include a short write-up with each portfolio item to explain context and purpose for future employers?”**
4. **“Are there any free hosting tools or platforms you recommend for showcasing GIS portfolio work online?”**

**🎯 GIS Portfolio Concept & Purpose**

* The **goal** of a GIS course portfolio is to create something **original** that:
  + Applies techniques you’ve learned in class
  + Demonstrates your **skills and creativity**
  + Can be taken to **job interviews**
  + Serves as a **talking point** to discuss for 2–3 minutes

💡 *“The portfolio should not just be visual. It should showcase your thought process and practical skill.”*

**⚠️ What *Not* To Do**

* **Don’t just include class assignments or workbook results.**
  + These are basic and not impressive to employers.
  + They don’t show initiative or creativity.
* You *can* revisit an assignment **only if**:
  + You improve it
  + You add new data, run a deeper analysis, or apply new methods

**💡 What *To* Do (Tips for Enhancing Portfolio Projects)**

1. **Apply something learned later** in the semester to an earlier project.
   * Example: Add **buffering** or **visual weight** to maps.
   * Re-run spatial analysis for better insights.
2. **Localize your project** for relevance.
   * Tailor an assignment to a real-world issue in your area (e.g. disaster response in Dallas).
   * This shows creativity **and** local knowledge.
3. **Create an app, web map, or dashboard** that lets people interact with your results.
   * This adds technical depth and polish.

**🧩 Professor’s Portfolio Example – Realistic Scope**

* He emphasized that **you can include incomplete projects**, as long as you:
  + Explain your goals
  + Describe what didn’t work
  + Reflect on what you learned
* He said: *“That would go over well in an interview.”*
  + Example: “This project was a bit more than I could do in the time I had, but I wanted to demonstrate the skills I was learning.”

**📍 Long-Term Access to ArcGIS Pro Organizational Account**

* Students *typically retain access* to the school’s GIS software/org unless they abuse it (e.g., using it for unrelated job work).
* That means you may have continued access to **ArcGIS Pro** and **Model Builder** **after this semester**.

**❓ Suggested Questions to Ask (Based on This Segment)**

These will help you refine your portfolio and connect it to job/internship goals:

**🛠️ Technical/Project Development**

1. **“Would tailoring a health map to Hope Supply Co. or the Red Cross count as localizing a project?”**
2. **“What are good ways to add visual weight to a map to make it more interview-worthy?”**
3. **“Can I add an interactive element like pop-up windows or time sliders to enhance a past assignment?”**
4. **“Could I re-run an earlier spatial analysis using different methods and explain why the new approach is better?”**

**💬 Interview/Portfolio Strategy**

1. **“In interviews, what kinds of portfolio stories or talking points do employers seem to find most impressive?”**
2. **“Should I include my rationale or workflow decisions in a write-up for each portfolio item?”**
3. **“If I include a dashboard, should I also provide screenshots or a video walkthrough in case the app isn’t live?”**

**Portfolio Submission Requirements**

**✅ What You Must Turn In**

For this course, you are required to submit the following **via eCampus**:

**🕒 Time Spent:**

* **Minimum 6+ hours** of work per portfolio item

**📂 Project Components:**

* **3 or more .py files** and/or **Model Builder** projects
* **Screenshots or exports** of your **Model Builder layouts**
* **Shared ArcGIS Pro project**:
  + Must include a clear **folder organization structure**

**📝 Word Document (Explanation of Your Project):**

Include:

* **What your project is** and **why you chose it**
* **Any difficulties** you encountered
* **How you solved** those issues

⚠️ Reminder: You must **upload both the document and the actual project files** — not just the write-up.

**🔄 Flexible Portfolio Formats**

* You can mix:
  + All Python
  + All Model Builder
  + Or combinations (e.g., 1 Python + 1 MB + 1 write-up)
* Emphasized: **It doesn't have to be all code**—reflection and documentation matter!

**❓ Suggested Questions to Ask (Portfolio Logistics & Strategy)**

**📁 File Prep & Structure**

1. **“Should we organize the ArcGIS Pro project folders in a specific way before sharing?”**
2. **“Are there size limits or naming conventions we should follow when uploading .py and Model Builder files?”**
3. **“Can we submit additional documentation like a PDF walkthrough or short video along with the Word doc?”**

**💡 Strategy for Success**

1. **“If we include Model Builder layouts, how detailed should they be to stand out?”**
2. **“Would showing how I solved a technical issue in code or ArcGIS be impressive in an interview?”**
3. **“Can we combine Python scripting and Model Builder in one portfolio item if they’re part of the same workflow?”**

**📍 Where to Find Data for Portfolio Projects**

Here are the recommended sources for **local and public data**:

**🌍 Living Atlas (ArcGIS)**

* Built into ArcGIS Pro.
* Access by:
  + **Add Data → Living Atlas** under Portal folder.
  + Or via the **Catalog Pane → Living Atlas tab** (last button across the top).
* Includes a wide range of **environmental, demographic, health, land use, etc.** layers.

**📊 Real-World Data Example**

* He referenced Dallas GIS data to calculate:
  + **Tax value by district**
  + **Acres by council district**
* Encouraged scripting localized projects (e.g., agricultural zoning, inner-city gaps, or council boundary analysis).

**📌 Local & Regional GIS Open Data Portals**

You can use data from the following:

* **Dallas**: [https://gis.dallascityhall.com](https://gis.dallascityhall.com/)  
  (Includes police beats, parks, schools, etc.)
* **Arlington**: <https://opendata.arlingtontx.gov/>
* **Garland**: [https://data-garland.opendata.arcgis.com](https://data-garland.opendata.arcgis.com/)
* **Richardson**: [https://opendata-richardson.opendata.arcgis.com](https://opendata-richardson.opendata.arcgis.com/)
* **Texas statewide data**: [https://data.texas.gov](https://data.texas.gov/)  
  (Includes elevation, floodplain, demographic, and health-related data)
* **U.S. Census**: <https://www.census.gov/data.html>

**❓ Suggested Questions to Ask (Data Sources & Application)**

**🔍 Data Access & Ethics**

1. **“Can we combine datasets from different cities (e.g., Dallas + Arlington) if we want to compare regions?”**
2. **“Are there any licensing or attribution requirements when using Living Atlas or city datasets in our portfolios?”**
3. **“Are we allowed to use national datasets (like CDC or FEMA) even if they weren’t listed here?”**

**💡 Data Strategy for Projects**

1. **“Could I map social vulnerability or disaster risk using a combo of Census and floodplain data?”**
2. **“Would it be valuable to build a health-focused dashboard using Census health insurance or poverty data by ZIP code?”**
3. **“Are there datasets in the Living Atlas that focus on public health, emergency response, or COVID-19 we could still use?”**

**💡 Local GIS Portfolio Ideas (DFW Area)**

**1. Disaster Response & Preparedness Map**

* **Data Sources**: Dallas Open Data (fire stations, shelters), FEMA flood zones, Living Atlas hazard data.
* **Tools**: ArcGIS Pro, Model Builder for evacuation modeling.
* **Outcome**: A web map or dashboard that helps visualize vulnerable areas and emergency service zones.

**2. Poverty & Health Access Analysis**

* **Data Sources**: U.S. Census, Dallas and Arlington datasets, Living Atlas for health insurance rates.
* **Tools**: Python scripts + ArcGIS for mapping access to clinics or hospitals by income or race.
* **Outcome**: A static map + short policy brief or blog-style write-up you can show hiring managers.

**3. Green Space Equity in Dallas**

* **Data Sources**: Dallas parks and schools data, Census demographics.
* **Tools**: Spatial join + buffers + symbology.
* **Outcome**: A map showing which neighborhoods lack access to parks or playgrounds.

**4. Transportation & Commute Pattern Analysis**

* **Data Sources**: Local traffic or DART routes + Census commute data.
* **Outcome**: A visual study of transportation deserts or areas that could benefit from new infrastructure.

**🧸 Hope Supply Co.-Relevant GIS Portfolio Ideas**

Several of those project ideas can be **strategically aligned with Hope Supply Co.’s mission**. Since they support **children and families in need** by providing essentials like diapers, food, and hygiene supplies, your GIS work could highlight **where the greatest needs exist** and how to **optimize delivery or outreach**.

**1. Child Poverty & Resource Access Map**

**Purpose**: Identify neighborhoods with the highest concentration of children in poverty and limited access to aid.

* **Data Sources**:
  + Census data: child poverty, households with children, SNAP benefits
  + Dallas open data: food pantries, shelters, public health clinics, etc.
* **Tool**: ArcGIS Pro + Model Builder or Python to automate data filtering and map generation.
* **Portfolio Angle**: Demonstrates spatial analysis, automation, and nonprofit impact planning.

**2. Diaper Need Hotspot Analysis**

**Purpose**: Visualize which zip codes or census tracts have high diaper need and limited access to supplies.

* **Data Sources**:
  + Census: households with infants/toddlers, poverty status
  + Local data: maternal health clinics, shelters
* **Tool**: Geoprocessing with buffers and choropleth mapping
* **Portfolio Angle**: Great for showcasing community-level decision support using Python/Model Builder.

**3. Delivery Route Optimization for Supply Drop-offs**

**Purpose**: Help optimize routes for supply deliveries (diapers, hygiene kits, food) using proximity and travel-time analysis.

* **Data Sources**:
  + Facility locations (Hope Supply Co., partner orgs)
  + Local road networks from Living Atlas or OpenStreetMap
* **Tool**: ArcGIS Network Analyst or script-based distance calculations
* **Portfolio Angle**: Shows logistics planning, valuable for nonprofits and government jobs.

**4. Resource Gap Map: Schools vs. Support Services**

**Purpose**: Map underserved schools or daycares that are far from supply pickup locations or outreach events.

* **Data Sources**:
  + Schools/daycares: Dallas/Arlington GIS portals
  + Supply partners: Hope Supply Co. or open data on pantries/shelters
* **Tool**: Spatial join + drive-time analysis
* **Portfolio Angle**: Demonstrates awareness of equity, useful in public health, education, and social service roles.

**❓ Suggested Questions to Ask Your Professor**

1. **“Would a map showing underserved families in Dallas be a strong portfolio item if I connect it to real local nonprofits?”**
2. **“Can I include a real-world organization like Hope Supply Co. in the write-up even if I don’t use their internal data?”**
3. **“Could a Model Builder tool that calculates high-need areas based on Census data and clinic/shelter access be an impressive automation project?”**

**🧩 Integrated GIS + Data Analytics Portfolio Ideas**

**1. Interactive Dashboard: Child Poverty, Services, and Supply Needs**

**Skills used**: ArcGIS, SQL, data joins, dashboard creation  
**Data Analytics tie-in**:

* Store cleaned Census + local service location data in a SQLite or PostgreSQL database.
* Query child poverty rates and link to shelter/clinic distances.
* Visualize in ArcGIS Dashboard with filters (by zip code, race, income level).

**2. Database-Driven Resource Allocation Model**

**Skills used**: RDBMS + spatial joins + ArcPy or Model Builder  
**Project idea**:

* Create a database with tables for:
  + Supply partner orgs
  + Monthly donation levels
  + Number of families served
* Link that to spatial data and create a map that shows “over-served” vs. “under-served” regions based on calculated need vs. supply.

**3. SQL-Backed Geodatabase: Hope Supply Co. Distribution Tracker**

**Skills used**: SQL queries, geodatabase design, ArcGIS  
**Project idea**:

* Design a schema to track Hope Supply Co. deliveries and recipient locations.
* Run SQL queries to show trends over time (e.g., where donations increased or dropped).
* Generate maps from the query results showing time-based patterns.

**❓ Suggested Questions for Your Professors**

* GIS Prof: **“Can I store and query data from a database for a spatial analysis project in my portfolio?”**
* Database Prof: **“Can I use our SQL/R skills to support an external portfolio project that involves spatial data?”**
* Both: **“Would it be valuable to create a relational database with spatial joins for a nonprofit-focused delivery map?”**

**📘 Chapter 9 Overview – Working with Geometries**

**Topics Covered**:

* Geometry objects let you read/write spatial features like **points, lines, and polygons**.
* Each geometry is built from **vertices**—pairs of x, y coordinates.
* In ArcPy, use:
  + SHAPE@ to get full geometry
  + SHAPE@XY, SHAPE@LENGTH, etc. to get specific properties (faster!)

**Key Classes & Tokens**:

* arcpy.Geometry – generic geometry object (point, line, polygon)
* arcpy.Point – single x,y coordinate
* arcpy.PointGeometry – Point object *with* spatial reference
* arcpy.Array – to store multiple Point objects and build Polyline or Polygon
* Multipart features (e.g. Hawaii = multiple islands = one polygon with parts)

**🛠️ Professor's Personal Project Example**

* Created a script that calculated **acres per park type** (e.g., dog parks, golf courses).
* Combined **Python + Model Builder** to do the work:
  + Python calculated acreage
  + Model Builder handled some data prep
  + Included a **write-up explaining** why he split the workflow and any issues he faced

💬 **Takeaway**: You *can mix and match* Python scripts, Model Builder, and write-ups in your portfolio.

**📂 Portfolio Guidance Recap**

* You can do:
  + 3 Python scripts, OR
  + 3 Model Builder projects, OR
  + A combination (e.g., 2 scripts + 1 write-up)
* The **write-up can reflect on a failed or unfinished attempt**, as long as you explain your process.
* **Real-world orgs (like Hope Supply Co.) are okay** to reference! Just avoid sensitive or personal data.

💡 Projects showing your **problem-solving process** are just as impressive as fully complete ones.

**📍 Examples of Local Data You Can Use**

* **City of Dallas GIS Hub**: Council boundaries, tax parcels, land use zones
* Possible project ideas:
  + Script that maps **acres per district**
  + Script that shows **tax value per zone**
  + Filter parcels by land use type (e.g., agricultural only)

You *can* also use **global datasets**, but the professor warns not to bite off too much—keep it focused!

**💡 Skills from This Chapter You Can Use in Your Portfolio**

* Use SHAPE@XY and SHAPE@LENGTH to:
  + Print centroid coordinates
  + Sum total lengths of polyline features
  + Measure areas or boundaries
* Use Point + Array + Polyline/Polygon to:
  + Build custom geometries from scratch
  + Compare geometries (e.g., .crosses())

**❓ Suggested Questions to Ask (Chapter 9 + Portfolio Strategy)**

**💬 Clarifying Geometries**

1. **“In what kind of projects is it better to use SHAPE@ over SHAPE@XY or SHAPE@LENGTH?”**
2. **“Can I build a multipolygon feature (like service zones) from raw CSV coordinate data?”**
3. **“Can you give an example of when to use a PointGeometry versus just a Point object?”**

**🧠 Project Strategy**

1. **“If I analyze tax parcels by district and visualize total area zoned for agriculture, is that a strong use of geometry tokens?”**
2. **“Could I script a proximity analysis from each council district to the nearest Hope Supply Co. partner using Point and Polyline classes?”**
3. **“If I write one script and one Model Builder tool that both analyze geometries, would that show enough technical variety for my portfolio?”**

**🔑 Key Concepts Covered**

**🧱 Geometry Objects in ArcPy**

* Geometry objects = shapes/features like **points**, **polylines**, and **polygons**.
* Access full geometry with SHAPE@ or specific properties with tokens like SHAPE@XY.

**🔁 Reading Geometry from Feature Classes**

* Use SearchCursor() to loop through geometry features.
* Use .getPart(0) for **single-part** features.
* For **multipart features**, loop through each part (like islands of Hawaii).
* isMultipart tells if a feature has multiple parts.
* .partCount returns how many parts exist in a geometry.

**🕳️ Polygons with Holes**

* A polygon can include:
  + **Exterior ring** (boundary – clockwise)
  + **Interior rings** (holes – counterclockwise)
* Use **null Point objects** to identify the start of an interior ring when looping through geometry.

**🛠️ Writing Geometry**

* You can create new features using:
  + InsertCursor() with Point() or Polygon()
  + Or CopyFeatures\_management() to write in-memory geometry to a feature class
* Coordinates can come from:
  + Python lists
  + External text files (e.g., points.txt)
  + Or dynamically generated scripts

**🌐 Setting Spatial Reference**

* Use SpatialReference() to define the coordinate system (e.g., WGS84 / EPSG:4326)
* Essential for ensuring geometry aligns properly across datasets
* You can set spatial references on:
  + Feature classes
  + Geometry objects
  + Cursors (for on-the-fly projection)

**⚙️ Using Geometry in Geoprocessing Tools**

* Geometry objects can be directly used in tools like Buffer\_analysis()
* Avoids creating temporary feature classes
* Improves speed and reduces clutter

**🔄 Reading Multipart Features**

* A **multipart feature** = multiple geometries represented as a single record.
  + 🗺️ Example: State of Hawaii (multiple islands = 1 record)
* Use getPart() and nested loops to:
  + Iterate over parts
  + Iterate over points in each part
* For **points**, these are called **multipoint** features.

**🌀 Working with Polygons with Holes**

* A polygon with holes contains:
  + **Exterior ring** (clockwise)
  + **Interior ring(s)** (counterclockwise)
* Rings are made of closed paths (from = to)
* Holes are indicated in scripts using **null Point objects**
* Script adjusts for null points to separate outer and inner rings.

🔍 Professor's Dallas example: Dallas surrounds cities like Highland Park and Cockrell Hill — those are **interior rings** from a GIS perspective, even though they're not part of Dallas proper.

**✍️ Writing Geometries**

* To **write new geometries** into a feature class:
  1. Create geometry (Point, PointGeometry, Array, etc.)
  2. Use InsertCursor or CopyFeatures
* For **point data**:
  1. You can use InsertCursor directly with Point
  2. Can use SHAPE@XY token to insert faster
* For **polygon/polyline**:
  1. Need to build using Array of Point objects
  2. Use Polygon() or Polyline() constructors
* CreateFeatureclass() lets you make a new empty feature class if needed

**🔁 Alternatives:**

* Instead of a cursor, use:
* arcpy.CopyFeatures\_management(geometry\_object, "output\_name")

This is faster for bulk feature creation but doesn't let you add attributes.

**🧠 What You Can Apply to Your Portfolio Project**

* **Use InsertCursor + geometry tokens** to write customized feature classes (e.g., map diaper drop-off zones for Hope Supply Co. using buffered polygons).
* **Analyze geometry structure** of Dallas-area zones to identify service gaps or equity issues.
* Use CopyFeatures() for **performance-focused batch geometry outputs** — great if you're creating visualizations for presentations.
* **Detect multipart or hollow polygons** in local GIS datasets to clean or preprocess data before mapping or analysis.

**❓ Suggested Questions to Ask in Class**

**🧩 Geometry Writing & Strategy**

1. **“If I want to show diaper distribution zones with inner ‘no-service’ areas, would a polygon with holes be the best structure?”**
2. **“Is there a way to automate detecting whether a geometry is multipart or has interior rings?”**
3. **“What’s the best practice for choosing between InsertCursor and CopyFeatures() for building local maps?”**

**💼 Portfolio + Real-World Context**

1. **“Could I use a Hope Supply Co. drop-off site dataset and visualize underserved areas based on service rings?”**
2. **“If I do a zone-by-zone tax map and add attributes like avg. household income, how do I sync that with geometry tokens?”**
3. **“Could you review a draft write-up where I explain a failed attempt to write new geometries and how I adapted?”**

**🔄 Creating Geometry Objects Without Cursors**

* You can use **arcpy.Point()** and **arcpy.Array()** to build geometry manually.
* Example:
* point = arcpy.Point()
* array = arcpy.Array()
* coordinates = [[x1, y1], [x2, y2], ...]
* for coord in coordinates:
* point.X = coord[0]
* point.Y = coord[1]
* array.add(point)
* polygon = arcpy.Polygon(array, 2277) # 2277 is the spatial reference

**✍️ Writing Geometry to Feature Class**

Two main methods:

1. **Insert Cursor** (adds features row by row):
2. with arcpy.da.InsertCursor(fc, ["SHAPE@"]) as cursor:
3. cursor.insertRow([polygon])
4. **CopyFeatures Tool** (copies geometry objects directly):
5. arcpy.CopyFeatures\_management(polygon, fc)

**📂 Reading Coordinates from a Text File**

* Uses fileinput to read from a .txt file:
* import fileinput
* for line in fileinput.input(textfile):
* point.ID, point.X, point.Y = line.split()
* array.add(point)
* polygon = arcpy.Polygon(array)
* Be sure the coordinates form a **closed loop** (first and last point must be the same) when creating polygons.

**📐 Setting the Spatial Reference**

* Can be done when creating the geometry object (e.g., arcpy.Polygon(array, 2277))
* Or when creating the feature class or cursor:
* sr = arcpy.SpatialReference("path/to/file.prj")
* with arcpy.da.SearchCursor(fc, ["SHAPE@"], "", sr) as cursor:
* ...

**📊 Exporting Coordinates to a Text File**

* Use a search cursor and write():
* output = open("result.txt", "w")
* for row in cursor:
* point = row[0]
* output.write(f"{point.X} {point.Y}\n")
* output.close()

**⚙️ Using Geometry Objects with Geoprocessing Tools**

* Geometry objects like PointGeometry can be directly used in tools like Buffer:
* point = arcpy.Point(x, y)
* pointgeo = arcpy.PointGeometry(point, sr)
* arcpy.Buffer\_analysis([pointgeo], "output.shp", "10 METERS")
* Or extract geometry from existing features:
* geolist = arcpy.CopyFeatures\_management(fc, arcpy.Geometry())
* for geom in geolist:
* print(geom.length)

**🧠 Key Takeaways ("Points to Remember")**

* Points, polylines, and polygons are geometry objects.
* Use SHAPE@ or SHAPE@XY for geometry access in cursors.
* Multipart and polygon-with-holes geometries need special handling.
* Geometry objects can be created in memory (efficient!) or written directly.
* Spatial reference matters for accurate results.
* You can streamline workflows by skipping the creation of temporary feature classes.

**🔑 Key Terms Quick Definitions**

| **Term** | **Definition** |
| --- | --- |
| **exterior ring** | The outer boundary of a polygon, drawn in a **clockwise** direction. |
| **format code** | A string like {0:.2f} that formats numbers in print output (e.g., 2 decimal places). |
| **geometry object** | An object representing a shape (Point, Polyline, Polygon) in memory. |
| **geometry token** | A shortcut like SHAPE@XY to access parts of a geometry (like X,Y only). |
| **interior ring** | A hole in a polygon, defined by a **counterclockwise** boundary. |
| **multipart feature** | A single feature made up of multiple separate shapes (like Hawaii's islands). |
| **multipoint feature** | A single feature that includes multiple points. |
| **null point** | A placeholder (empty point) used to separate interior rings from the exterior ring. |
| **path** | A series of connected vertices that define part of a line or ring. |
| **polygon with holes** | A polygon that contains one or more interior rings (holes). |
| **ring** | A closed path forming part of a polygon’s boundary. |
| **single-part feature** | A feature made up of only one shape. |
| **vertex** | A coordinate pair (x,y) marking a point on a line or polygon boundary. |

**📘 Review Questions – Sample Answers**

1. **When do you use the full geometry object (SHAPE@) vs a geometry token (e.g., SHAPE@XY)?**
   * Use SHAPE@ when you need access to the **entire geometry** (e.g., vertices, length, area).
   * Use tokens like SHAPE@XY when you only need **specific properties** like x and y coordinates — it's faster and uses less memory.
2. **Difference between Point and PointGeometry?**
   * Point is a simple object holding x,y (and optionally z/m) coordinates.
   * PointGeometry wraps a Point and includes spatial reference info. You need PointGeometry when you're creating geometry for use in geoprocessing tools or spatial operations.
3. **Steps to read geometry and print vertex coordinates:**
   * Use a SearchCursor with SHAPE@
   * Loop over each row and access .getPart() or .getPart(0)
   * Loop through points in the part and print .X, .Y (with optional formatting like {0:.2f})
4. **How do multipart features and polygons with holes change your workflow?**
   * Multipart: You need to loop through **each part** of the geometry (for part in row[1]:)
   * Polygons with holes: You check for **null points** inside a ring to detect interior rings.
5. **Steps to create a new geometry from a coordinate list:**
   * Create Point and Array objects
   * Loop over the list, assign x/y to Point, add to Array
   * Create Polygon or Polyline using the Array
   * Use InsertCursor or CopyFeatures to write to a feature class
6. **What does the Array class do in ArcPy, and when do you use it?**
   * It stores a collection of Point objects.
   * You use it when building complex geometries like lines or polygons from multiple vertices.

**🎯 Career & Academic Connections**

* 📌 **Data Science + GIS**: Learning to programmatically read/write geometry will help you build spatial analysis pipelines—essential in **public health**, **urban planning**, and **disaster response**.
* 🤖 **AI + Automation**: You can automate the creation of training datasets for ML models (e.g., classifying satellite imagery or disease hotspots).
* 🏥 **Health GIS**: Mapping health data (e.g., maternal mortality clusters) with accurate spatial reference is critical for **epidemiological analysis** and **health equity initiatives**.
* 🧪 **Research**: These scripting skills will help you create reproducible spatial data workflows for academic projects, theses, or Ph.D. work.
* 📊 **Portfolio Boost**: These tools and scripts are perfect for showing off GIS proficiency in your **job portfolio** or **grad school applications**.

**🔍 Concept Clarification & Deeper Understanding**

1. **When working with polygons with holes, how can we validate that the ring directions (clockwise vs counterclockwise) are correctly assigned?**
2. **Can you explain a real-world example where multipart features are essential for analysis, beyond the Hawaii example?**
3. **What are the pros and cons of using CopyFeatures vs InsertCursor when creating new features, especially for large datasets?**

**💡 Practical Use Cases & Advice**

1. **How often do you deal with geometry objects directly in professional GIS work versus relying on pre-built tools in ArcGIS Pro or QGIS?**
2. **When should we use geometry tokens like SHAPE@XY versus the full SHAPE@ geometry object in performance-heavy scripts?**
3. **What’s the best way to handle large feature classes where printing or analyzing all vertices might be inefficient or crash the script?**

**🐛 Common Mistakes in Geometry Scripts**

* The **most common error?** Simple **typos** and **not using error messages effectively**.
* Tip: **Proofread your code**, and use error logs/tracebacks to identify mistakes quickly.

**🛠️ Techniques and Troubleshooting**

1. **If a polygon doesn’t render properly in ArcGIS, how would you diagnose whether the issue is with the ring direction, null points, or multipart structure?**
2. **What’s a common mistake students make when writing geometry-based scripts, and how can we avoid it?**
3. **How can we automate ring direction correction if we have imported data from a source that doesn’t follow the clockwise/counterclockwise convention?**

**🌍 Career and Skills Advice**

1. **Which skills in working with geometry objects and cursors do employers value most in entry-level GIS analysts?**
2. **Are there specific types of GIS projects where using geometry objects directly is preferred over using feature classes or shapefiles?**
3. **Can knowledge of geometry scripting help in non-traditional GIS roles, like remote sensing or urban planning?**

**📣 Class Announcements & Tips**

* ✅ Always double-check your **spatial reference** to avoid alignment issues.
* 🐌 If dealing with large datasets, consider limiting output using counters or if conditions.
* 📂 Create reusable scripts to read or write geometry using lists or external files (like points.txt).
* 💬 Real-world example: City of Dallas’ **jurisdiction quirks** (e.g., Cockrell Hill & Highland Park are "islands" not actually part of Dallas).

**📅 Portfolio Time Estimate Adjustment**

* Originally, students were expected to spend **6+ hours** per portfolio item across 2 weeks.
* Since the timeline has changed to **just 1 week**, the professor plans to **adjust that expectation** down (approx. **3+ hours per item** minimum now).
* Class time allocated = approx. **4 hours and 40 minutes total** across that week.

**🧩 Professor’s Portfolio Example – Realistic Scope**

* He emphasized that **you can include incomplete projects**, as long as you:
  + Explain your goals
  + Describe what didn’t work
  + Reflect on what you learned
* He said: *“That would go over well in an interview.”*
  + Example: “This project was a bit more than I could do in the time I had, but I wanted to demonstrate the skills I was learning.”

**📊 Real-World Data Example**

* He referenced Dallas GIS data to calculate:
  + **Tax value by district**
  + **Acres by council district**
* Encouraged scripting localized projects (e.g., agricultural zoning, inner-city gaps, or council boundary analysis).

**🔄 Flexible Portfolio Formats**

* You can mix:
  + All Python
  + All Model Builder
  + Or combinations (e.g., 1 Python + 1 MB + 1 write-up)
* Emphasized: **It doesn't have to be all code**—reflection and documentation matter!

**📍 Long-Term Access to ArcGIS Pro Organizational Account**

* Students *typically retain access* to the school’s GIS software/org unless they abuse it (e.g., using it for unrelated job work).
* That means you may have continued access to **ArcGIS Pro** and **Model Builder** **after this semester**.

**🐛 Common Mistakes in Geometry Scripts**

* The **most common error?** Simple **typos** and **not using error messages effectively**.
* Tip: **Proofread your code**, and use error logs/tracebacks to identify mistakes quickly.