

Assignment 3 – GGS590 GeoAI – Functions, testing, and profilingDue February 18th 2026

This assignment aims to reinforce the core ideas from this week’s class, including:

- Refactoring script-based geospatial workflows into reusable functions, identifying inputs, outputs, and side effects in GeoAI pipelines.
- Designing and implementing unit tests for geospatial functions using assert, verifying correctness of geometry, CRS, and dataset structure.
- Constructing integration tests that validate multi-step spatial workflows, ensuring that combined operations produce coherent and non-empty results.
- Applying spatial validation checks to assess whether outputs are plausible and consistent with domain knowledge.
- Critically evaluating the reliability of automated geospatial workflows, distinguishing between code that merely runs and code that produces trustworthy results.

To complete this assignment, you will build a small geospatial workflow twice (with both versions producing equivalent analytical results):

1. As a sequential script (e.g., to produce the basic analysis in a linear/sequential way).
2. Then refactored into functions with added unit + integration tests to demonstrate you can trust the workflow.

You will use a boundary dataset (e.g., states, counties, etc.), as well as a feature dataset (such as from Natural Earth for places, rivers, lakes, urban areas etc.).

Depending on your data type, you then either need to carry out:

- A points-in-polygon analysis (e.g., for a research question asking “how are cities distributed across states/counties”) or
- A line/polygon-in-boundaries analysis (e.g., for a research question asking “how much river length / urban area exists per state/county”).

You can choose the topic, the necessary data, research question, and appropriate processing pipeline. Just make sure to organize and clearly number your analysis in appropriate sections (e.g., for the 5 parts).

Outputs

Ideally, you will submit your notebook rendered as a PDF that contains:

1. Script-based workflow (Part 1, 20 points)
2. Function-based refactor (Part 2, 20 points)
3. Unit tests (Part 3, 20 points)
4. Integration test(s) (Part 4, 20 points)
5. Written up outputs, with at least one map and a description of how the results answer the research question (Part 5, 20 points).

The learning emphasis here is on showing you can piece together an analytical workflow to answer a research question, restructure it into functions, and write some appropriate tests to show it works correctly.

Submission format

Please write your analysis in a Google Colab notebook and then use the print function to save it to a .pdf file for submission on Canvas. You can also submit a Word document rendered to PDF, if you desire (e.g., if you have formatting issues with images/maps). Without submitting your code files like this, you will receive a 50-point penalty to your overall grade (as you need to provide both the code and the answers).

The Mason Honor code applies. Please also remember to follow the Mason AI use policy.