

# HEC-Commander Repository

## Open Source Notebooks:



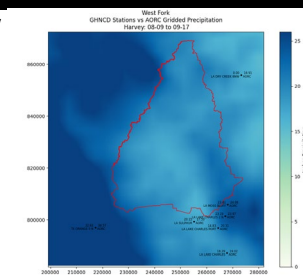
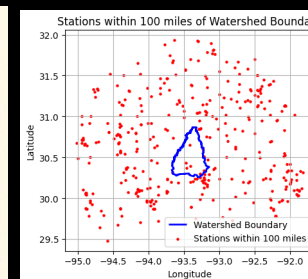
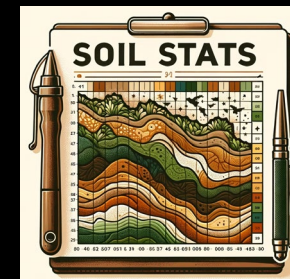
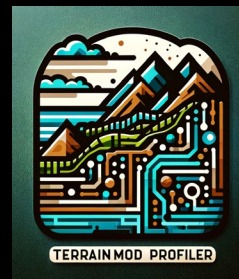
## Blogs



## ChatGPT Examples and GPT's



## Miscellaneous H&H Tools related to LWI Region 4 Efforts












HEC-Commander  
Repository (GitHub)



# AI Assistants (GPT's) for H&H Workflows

- Interact with HEC-RAS Data Formats:
  - HDF (h5py)
  - DSS (pydsstools)
- Hydromet and USGS Data
  - Access NHDPlus, USGS 3DEP, NLCD via API (hyriver)
  - Terrain Processing
  - Geospatial Data Analysis Tasks
  - Postprocessing for Mapping
- HEC-RAS Forum Assistant
  - Answers reinforced with RAS Solution Transcripts and Kleinschmidt Forum Posts










Table of GPT's and ChatGPT Examples:

Logo	Number	Name	Description	Read More Link	GPT Link
	29	LLM API Library Assistant	The LLM API Library Assistant provides tools for interfacing with various large language model APIs. It helps users integrate LLM functionalities into their applications, facilitating tasks such as text generation, summarization, and sentiment analysis.	<a href="#">Read More</a>	<a href="#">GPT Link</a>
	28	USGS API Assistant	This assistant interfaces with various USGS APIs to retrieve and analyze hydrological and geological data. It simplifies accessing datasets such as streamflow records, water quality measurements, and topographic information.	<a href="#">Read More</a>	<a href="#">GPT Link</a>
	27	Python HDF File Assistant	The Python HDF File Assistant facilitates working with HDF5 files, which are commonly used for storing large datasets. It provides functionalities for reading, writing, and querying HDF5 data.	<a href="#">Read More</a>	<a href="#">GPT Link</a>
	26	Python Raster Array Assistant	This tool assists with the manipulation and analysis of raster data using Python arrays. It supports tasks such as resampling, reclassification, and statistical analysis of raster datasets.	<a href="#">Read More</a>	<a href="#">GPT Link</a>
	25	DSS Python Assistant	The DSS Python Assistant aids in manipulating and analyzing data stored in the HEC-DSS format, providing Python tools for reading, writing, and visualizing time-series and spatial data.	<a href="#">Read More</a>	<a href="#">GPT Link</a>
	24	Shapefile Simplifier for Geospatial Processing	This assistant focuses on simplifying shapefiles, making them more manageable for geospatial processing tasks. It helps reduce file size and complexity while preserving essential spatial information.	<a href="#">Read More</a>	<a href="#">GPT Link</a>
	23	HEC-RAS Forum Assistant	The HEC-RAS Forum Assistant helps users navigate and extract valuable information from HEC-RAS forum discussions, offering solutions to common modeling problems and enhancing the efficiency of using the HEC-RAS software.	<a href="#">Read More</a>	<a href="#">GPT Link</a>
	22	Hydroclimate Data Tools with HyRiver	This tool integrates HyRiver, a suite of Python libraries for accessing and analyzing hydroclimate data, providing capabilities for retrieving and processing large-scale climate and hydrological datasets.	<a href="#">Read More</a>	<a href="#">GPT Link</a>
	21	Geospatial Python Notebook Assistant using WhiteboxTools	This assistant leverages the WhiteboxTools library within a geospatial Python notebook, offering functionalities for terrain analysis, hydrological modeling, and more. It is designed to support geospatial data processing and analysis workflows.	<a href="#">Read More</a>	<a href="#">GPT Link</a>

# HEC-RAS Python Tools

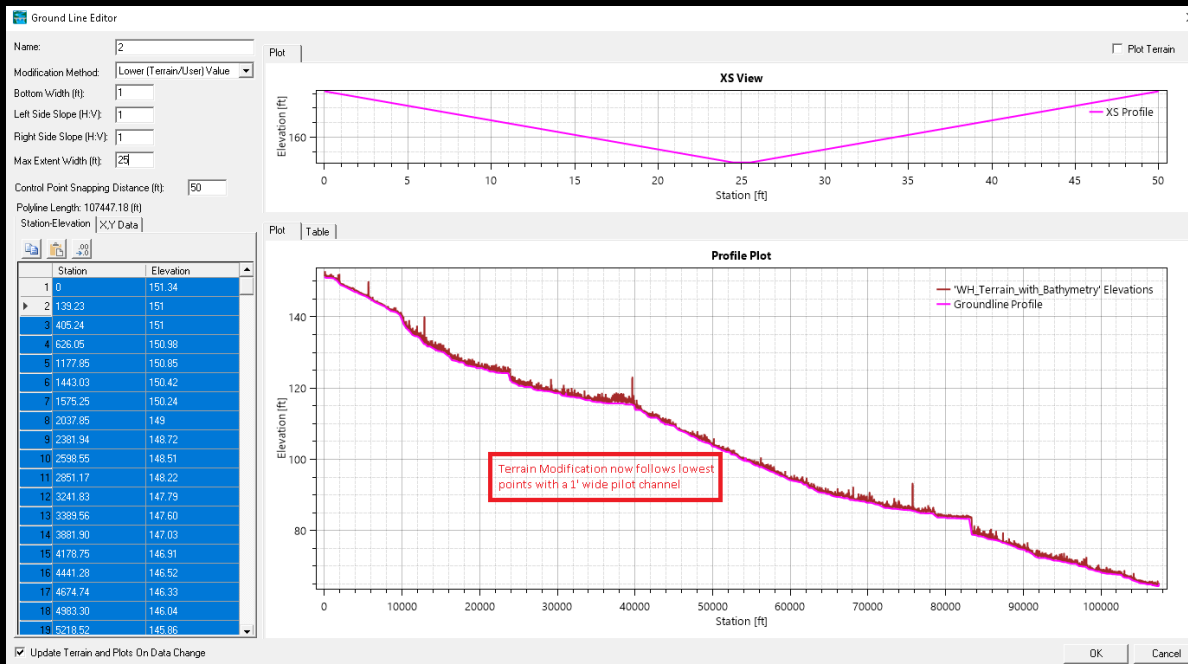
- RAS-Commander
  - HEC-RAS Calibration and Validation
  - Parallelization for Multiple Runs
- DSS-Commander
  - Visualizing DSS Results
  - Calculating Calibration Statistics
- Atlas 14 Data Variance Analysis
- Soil Stats Tool
  - May be obsolete with 6.5
- Terrain Modification Profiler

Table of GPT's and ChatGPT Examples:

Logo	Number	Name	Description	Read More Link	GPT Link
	29	LLM API Library Assistant	The LLM API Library Assistant provides tools for interfacing with various large language model APIs. It helps users integrate LLM functionalities into their applications, facilitating tasks such as text generation, summarization, and sentiment analysis.	<a href="#">Read More</a>	<a href="#">GPT Link</a>
	28	USGS API Assistant	This assistant interfaces with various USGS APIs to retrieve and analyze hydrological and geological data. It simplifies accessing datasets such as streamflow records, water quality measurements, and topographic information.	<a href="#">Read More</a>	<a href="#">GPT Link</a>
	27	Python HDF File Assistant	The Python HDF File Assistant facilitates working with HDF5 files, which are commonly used for storing large datasets. It provides functionalities for reading, writing, and querying HDF5 data.	<a href="#">Read More</a>	<a href="#">GPT Link</a>
	26	Python Raster Array Assistant	This tool assists with the manipulation and analysis of raster data using Python arrays. It supports tasks such as resampling, reclassification, and statistical analysis of raster datasets.	<a href="#">Read More</a>	<a href="#">GPT Link</a>
	25	DSS Python Assistant	The DSS Python Assistant aids in manipulating and analyzing data stored in the HEC-DSS format, providing Python tools for reading, writing, and visualizing time-series and spatial data.	<a href="#">Read More</a>	<a href="#">GPT Link</a>
	24	Shapefile Simplifier for Geospatial Processing	This assistant focuses on simplifying shapefiles, making them more manageable for geospatial processing tasks. It helps reduce file size and complexity while preserving essential spatial information.	<a href="#">Read More</a>	<a href="#">GPT Link</a>
	23	HEC-RAS Forum Assistant	The HEC-RAS Forum Assistant helps users navigate and extract valuable information from HEC-RAS forum discussions, offering solutions to common modeling problems and enhancing the efficiency of using the HEC-RAS software.	<a href="#">Read More</a>	<a href="#">GPT Link</a>
	22	Hydroclimate Data Tools with HyRiver	This tool integrates HyRiver, a suite of Python libraries for accessing and analyzing hydroclimate data, providing capabilities for retrieving and processing large-scale climate and hydrological datasets.	<a href="#">Read More</a>	<a href="#">GPT Link</a>
	21	Geospatial Python Notebook Assistant using WhiteboxTools	This assistant leverages the WhiteboxTools library within a geospatial Python notebook, offering functionalities for terrain analysis, hydrological modeling, and more. It is designed to support geospatial data processing and analysis workflows.	<a href="#">Read More</a>	<a href="#">GPT Link</a>

# Terrain Modification Profiler

- Helps create pilot channels in LIDAR-defined channels with tree cover (noisy profiles)
- Uses channel centerlines and your RAS terrain to automate generation of profiles



# AI Coding in Local Notebooks

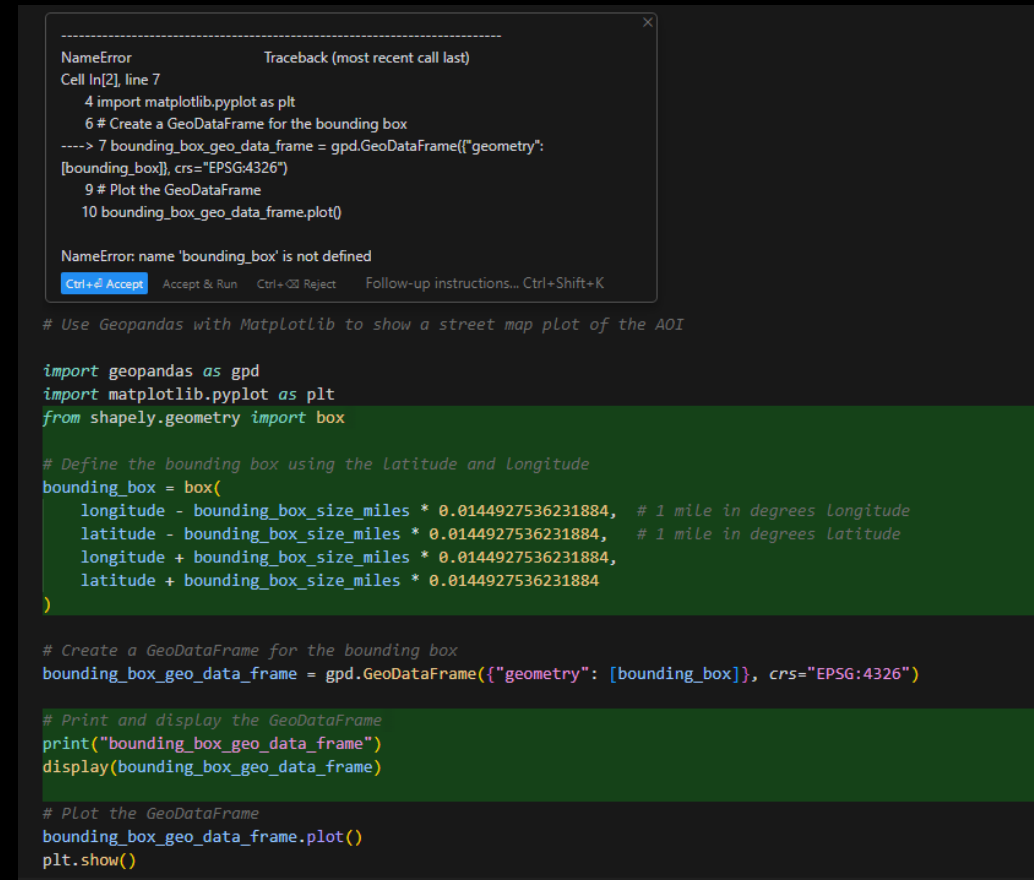
Best in Class:

Cursor, an AI-Powered VS Code fork

Most powerful way to bring your code local and continue with small iterations, debugging and improving scripts.

With recent gpt-4o-mini and Llama-3.1 models, open-source options are now available for very low cost or free.

*More in-depth AI-Assisted coding and GPT building will be covered in the Python Masterclass and future AI Applications premium webinars*



```
-----  
NameError                                Traceback (most recent call last)  
Cell In[2], line 7  
      4 import matplotlib.pyplot as plt  
      6 # Create a GeoDataFrame for the bounding box  
----> 7 bounding_box_geo_data_frame = gpd.GeoDataFrame({"geometry":  
[bounding_box], crs="EPSG:4326")  
      9 # Plot the GeoDataFrame  
     10 bounding_box_geo_data_frame.plot()  
  
NameError: name 'bounding_box' is not defined  
Ctrl+⌘ Accept  Accept & Run  Ctrl+⌘ Reject  Follow-up instructions... Ctrl+Shift+K  
  
# Use Geopandas with Matplotlib to show a street map plot of the AOI  
  
import geopandas as gpd  
import matplotlib.pyplot as plt  
from shapely.geometry import box  
  
# Define the bounding box using the latitude and longitude  
bounding_box = box(  
    longitude - bounding_box_size_miles * 0.0144927536231884, # 1 mile in degrees longitude  
    latitude - bounding_box_size_miles * 0.0144927536231884, # 1 mile in degrees latitude  
    longitude + bounding_box_size_miles * 0.0144927536231884,  
    latitude + bounding_box_size_miles * 0.0144927536231884  
)  
  
# Create a GeoDataFrame for the bounding box  
bounding_box_geo_data_frame = gpd.GeoDataFrame({"geometry": [bounding_box], crs="EPSG:4326")  
  
# Print and display the GeoDataFrame  
print("bounding_box_geo_data_frame")  
display(bounding_box_geo_data_frame)  
  
# Plot the GeoDataFrame  
bounding_box_geo_data_frame.plot()  
plt.show()
```

# Pick a model on hec-06

C:\AutoRAS\_Run

LC\_SST

- TOGGLE ALL  Delta 2020  Harvey 2017  June 2018  May 2021  SST100YR\_Choupique\_Bayou  SST005YR\_Choupique\_Bayou  SST010YR\_Choupique\_Bayou  SST025YR\_Choupique\_Bayou
- SST050YR\_Choupique\_Bayou  SST200YR\_Choupique\_Bayou  SST500YR\_Choupique\_Bayou  SST100YR\_Full  SST100YR\_South\_Ward\_3  SST100YR\_Lake\_Charles  SST100YR\_Sulphur  SST005YR\_Full
- SST005YR\_South\_Ward\_3  SST005YR\_Lake\_Charles  SST005YR\_Sulphur  SST010YR\_Full  SST010YR\_South\_Ward\_3  SST010YR\_Lake\_Charles  SST010YR\_Sulphur  SST025YR\_Full
- SST025YR\_South\_Ward\_3  SST025YR\_Lake\_Charles  SST025YR\_Sulphur  SST050YR\_Full  SST050YR\_South\_Ward\_3  SST050YR\_Lake\_Charles  SST050YR\_Sulphur  SST200YR\_Full
- SST200YR\_South\_Ward\_3  SST200YR\_Lake\_Charles  SST200YR\_Sulphur  SST500YR\_Full  SST500YR\_South\_Ward\_3  SST500YR\_Lake\_Charles  SST500YR\_Sulphur

Give your run suite a unique name

SST\_LC\_all\_runs

min # cores per run

max # cores per run

1

1

RUN 35 PLANS



Full Momentum Episode 33: The Future of ...

(00:00) Introduction, (0:35) Special Guest Introduction, (1:00) Introduce the Main Topic- The Future of leveraging...

youtu.be