

PyQGIS rainfall interpolation and map export

Goal

Script that takes an input csv file with rainfall measurements at point locations, interpolates a rainfall surface, then exports a map in png format including boundary layers and Google Satellite imagery.

Requirements

- QGIS 3.xx

Inputs

PROJECT_NAME	The name of the project. This will appear in the legend title and will be used to name output files
INPUT_RAINFALL_CSV_PATH	The filepath to the rainfall csv
XFIELD	The name of the field in the input rainfall csv that has X coordinates for the data
YFIELD	The name of the field in the input rainfall csv that has Y coordinates for the data
INTERPOLATION_DATA_FIELD_NAME	The name of the field in the input rainfall csv that has the rainfall data that will be used to create interpolated raster
BOUNDARY_LAYERS_PATHS	List of file paths for boundary layers to include in the final map. If you just have one boundary, you must still wrap it in a list.
OUTPUT_FOLDER	Directory to place permanent output files (interpolated rainfall raster, final map)
QGIS_PREFIX_PATH	File path to our qgis install. Specifically, this is the path to the \apps\qgis folder within your qgis install. On my system (windows, it's C:\Program Files\QGIS 3.4\apps\qgis). Search 'pyqgis 3 prefix path linux' to get an idea where this should point for a linux os.
TEMP_FOLDER	Location to place temporary files. These will be cleaned up after each run
BOUNDARY_STYLE_FILE	Path to a .qml style file for the boundary layers
RAINFALL_POINTS_STYLE_FILE	Path to a .qml style file for the rainfall measurement points
COLOUR_PALLETE	List of hex colour values used to style the interpolated rainfall raster.
PROJECT_EPSG	EPSG code for coordinate system of project – this MUST match the input csv coordinates. In practice, this will likely always be 4326, unless your input csv data was recorded in a different crs
RESOLUTION	The resolution in meters of the produced rainfall interpolation raster
INTERPOLATION_PAD_DISTANCE	The distance in meters to pad the given extent by when running the interpolation
INTERPOLATION_EXTENT_LAYER	The layer extent to use when deciding where to interpolate data to. ONLY VALID VALUES ARE 'boundary' or 'points'. Boundary means use the union of all boundary layer extents, points means use the csv rainfall measurement points.
EXPORT_DPI	Resolution of final png map

Outputs

Rainfall Raster	Raster in .tif format, coordinate system WGS84 showing interpolated rainfall values derived from input csv file.
Final Map	Final output map in .png format showing Google Satellite base, semi-transparent rainfall raster, rainfall points from csv layer, and boundary layers. Includes legend showing rainfall in mm.

Usage

Script is currently just meant to be run from a python interpreter or from command line without arguments setup. This means that input parameters must be edited within the .py file as opposed to being fed through command line arguments. This can be changed as needed on your side, however I wasn't sure how you planned on integrating this so I thought it would be best to start simple.

I've tried to break the parameters up into ones that will be commonly changed for each run (ie the file path to input data, project name, field names in input csv, etc) and parameters that you'll likely just set onces (ie your QGIS prefix path, temp folder location, paths to style files, etc). Hopefully this makes is easier to use.

Example output

