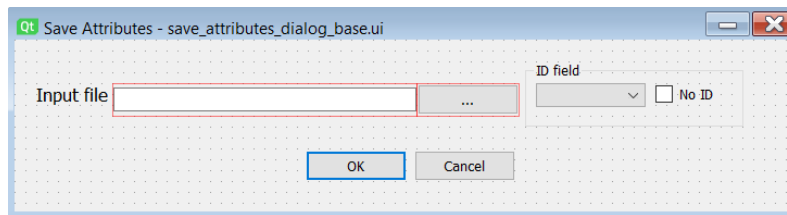


GMT 456 – GIS Programming 2020 – 21 Fall Midterm Project Guidelines

Fork the course's [GitHub page](#). You may start investigating the QGIS plugin under the `save_attributes/` folder.

The GUI of the plugin is as follows:



At the moment, this plugin adds two attributes (X and Y) to a point shp file and populates them by inserting the relevant values corresponding to each feature.

The objective of the midterm project is to update the plugin such that the following requirements are satisfied:

- If a **point** shp file is provided, it will identify the distances between all features (points). The plugin will then create a **new line shp file**, which provides the details of the two special lines (i.e. the min/max distance between the two closest/distant points, where the distance is calculated as the shortest distance between two points).
 - Template of the resulting line shape file looks like:

<i>Optional</i>		
id_start	id_end	Length
id1_shortest	id2_shortest	distance between two closest pairs
id2_longest	id2_longest	distance between two most distant pairs

- If the user provided an id field corresponding to each point feature, then the start and end points of these two special lines are recorded as shown in the first two columns.
 - If there is no such id field, the user will check the box 'No ID', and then only the length of the lines are recorded. That's why the first two fields (id_start and id_end) are optional.
- If a **polyline** shapefile is provided, it will add two attributes (fields) to the shp file.
 - The first attribute is **length**, which records the length of the polyline.
 - The second attribute is **shortest_length**, which first identifies the start and end points of a polyline, and records the shortest distance between these two points.
- If a **polygon** shapefile is provided, provide a gentle warning to the user stating that the plugin cannot operate on polygons.

You need to consider the following adjustments to increase the quality of your plugin:

1) Add a control statement that checks whether a new field that you define is already present? If the new field(s) are present, there is no need to add once more. This will be useful when debugging the code multiple times, and prevent situations like this in which we add the same attributes (i.e. x and y) multiple times:

	id	name	x	y	x_1	y_1	x_2	y_2
1	2722573678	101. Lojman	NULL	NULL	NULL	NULL	NULL	NULL
2	2722573679	MSSF	NULL	NULL	NULL	NULL	NULL	NULL
3	2722573681	Doğu Kampüs ...	NULL	NULL	NULL	NULL	NULL	NULL
4	2722573682	103. Lojman	NULL	NULL	NULL	NULL	NULL	NULL

2) At the moment, the plugin adds the X and Y fields regardless of the input geometry. Make sure that the new fields that you will be adding are specific to the input geometry.

3) Investigate ways in which to increase the quality of the GUI layout. At the moment, the **horizontal layout** is not correctly structured, so that the “browse” button takes more space than it should. Can you also identify other aspects which increases the quality of the GUI?

Input file