# Create GEOTIFF I/O template in Python.

The folder “code” contains two files:

1. a Jupyter Notebook containing the function and the example.
2. a Python file containing the function.

Required libraries:

* NumPy
* Rasterio

The example has the purpose of creating a GeoTIFF file (correctly defining and attaching all metadata) from a generic NumPy array.

To do this, a random array was generated and from this the GeoTIFF was built.

All GeoTIFF files with their respective metadata were checked using QGIS for better visualisation.

In the below 4-steps description of the example there are two steps where, instead of using random arrays, data (but not metadata) from two existing GeoTIFF files (one multi-band and one single band) are used. This is to verify that the metadata obtained from the function is consistent.

This example is performed in four steps:

1. defining of the function

Rasterio library is used to define a function that, from a generic NumPy array, creates a GeoTIFF file by correctly attaching metadata.

The function is parametric with some optional fields and other are required.

Some parameters are passed in by the user and are required while others already have a default value but can still take another as input.

Parameters:

* Required
  + the path of GeoTIFF output file
  + data array
  + CRS
* Optional
  + Data type (default is “Byte” but also “Float32” is allowed)
  + Compression method (default is “LZW”)
  + Block x size (must be multiples of 16 and default is 64)
  + Block y size (must be multiples of 16 and default is 64)
  + Photometric (default is “RGB” but for single band file is used “Gray”)

*def geotiff\_builder(output\_path, data\_array, crs\_input, data\_type = "Byte", compression\_method = "LZW", block\_x\_size = 64, block\_y\_size = 64, pm = 'RGB')*

1. construction of a NumPy array consisting of random values, construction of the GeoTIFF file with its metadata and visualisation on QGIS.

Graphical user interface, text, application, email

Description automatically generated

After saving the GeoTIFF file obtained from the NumPy random array to disk, it is tried to open it in QGIS to check that everything went well.

Graphical user interface

Description automatically generated with medium confidence

The file is loaded correctly, without returning errors and with the chosen CRS.

Graphical user interface, application

Description automatically generated

1. GeoTIFF file creation using an array of data from an existing (multi band) GeoTIFF and verification of metadata consistency in QGIS.Graphical user interface, application

   Description automatically generated

Graphical user interface, application

Description automatically generated

1. Creation of a GeoTIFF file using an array of data from an existing (single band) GeoTIFF and verification of metadata consistency in QGIS.Graphical user interface

   Description automatically generated

Graphical user interface

Description automatically generated