



---

## QGIS Training Modules



---

### Module 0 – Introduction to GIS and QGIS



# Introduction to GIS and QGIS

Some of this text was adapted by Saylor Academy under a Creative Commons Attribution-NonCommercial-ShareAlike 3.0 License without attribution as requested by the work's original creator or licensor.



## Introduction

In this module we introduce the concept of thinking spatially, and how we associate data and information with a location. We introduce the basic concepts of spatial data and the different spatial data models and how this data can be used and stored as well as introducing basic geometric operations. In the second part of this module, we present a tutorial on the basics of working with spatial data in QGIS, setting up a project, loading data into QGIS, visualizing data, using attribute tables and finally saving and exporting your data.

## Data Folder

Please, create your own working space on your local drive and save the data and instructions for this module from `\\lowfilecds\Function\GIS\Training\QGIS\Module0_Introduction_to_QGIS\`. The data folder contains the following, required for the tutorial section of this module:

- world\_boundaries.shp

For the tutorial section of this module, please skip to [here](#).

# Spatial Thinking

At no other time in the history of the world has it been easier to create or acquire a map of nearly anything. Maps and mapping technology are physically and virtually everywhere. Though the modes and means of making and distributing maps have been revolutionized with recent advances in computing, the art and science of map making date back centuries. This is because humans are inherently spatial organisms, and to live in the world, we must first somehow relate to it.

## Mental Maps

Mental or cognitive maps are psychological tools that we all use every day. As the name suggests, mental maps are maps of our environment that are stored in our brain. We rely on our mental maps to get from one place to another, to plan our daily activities, or to understand and situate events that we hear about from our friends, family, or the news. Mental maps also reflect the amount and extent of geographic knowledge and spatial awareness that we possess.



**[EXERCISE]** To illustrate this point, pretend that a friend is visiting you from out of town for the first time. Using a blank sheet of paper, take five to ten minutes to draw a map from memory of your hometown that will help your friend get around.

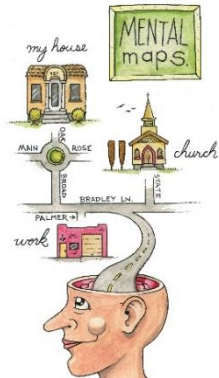


Figure 1. Mental map illustration

What did you choose to draw on your map? Is your house or where you work on the map? What about streets, restaurants, malls, museums, or other points of interest? How did you draw objects on your map? Did you use symbols, lines, and shapes? Are places labelled? Why did you choose to include certain places and features on your map but not others? What limitations did you encounter when making your map?

This simple exercise is instructive for several reasons. First, it illustrates what you know about where you live. Your simple map is a rough approximation of your local geographic knowledge and a mental map. Second, it highlights the way in which you relate to your local environment. What you choose to include and exclude on your map provides insights about what places you think are important and how you move through your place of residence. Third, if we were to compare your mental map to

someone else's from the same place, certain similarities emerge that shed light upon how we as humans tend to think spatially and organize geographical information in our minds. Fourth, this exercise reveals something about your artistic, creative, and cartographic abilities. In this respect, not only are mental maps unique, but also the way in which such maps are drawn or represented on the page is unique too.

Imagine you are going to an international conference and someone from another country ask you to draw where Cefas is located. Probably you aren't going to draw in detail the train station and the roads to reach Cefas building, instead your mental map would picture the location of Lowestoft and then you would describe Lowestoft by giving some attributes about the location (e.g., population, beach, building description, etc). Without knowing it, you have created the basic geographic information system, this means a **location and a set of attributes related to this location**.



Figure 2. Location of cities.

Mental maps are awesome, think now that you are asked to draw again the location of Cefas by a person living in Norwich or a person from Lowestoft. The references you would use would be completely different and your mental maps give more or less detail depending on what knowledge you expect from the town where Cefas building is located. This means you structure and filter your knowledge and information based on the intended audience.

# What is a geometry

The process of moving from the “real world” to the world of maps is referred to as map abstraction.

Within the realm of maps, the world is made up of various features or entities. Such entities include but are not restricted to fire hydrants, caves, roads, rivers, lakes, hills, valleys, or oceans. Moreover, such features have a form, and more precisely, a geometric form. For instance, fire hydrants and geysers are considered point-like features; rivers and streams are linear features; and lakes, countries, and forests are areal features.



Figure 3. Oulton Broad from above

## Geometry representation in cartesian system

A Cartesian coordinate system in two dimensions (also called a rectangular coordinate system or an orthogonal coordinate system) is defined by an ordered pair of perpendicular lines (axes), a single unit of length for both axes, and an orientation for each axis.

The Cartesian coordinates (also called rectangular coordinates) of a point are a pair of numbers (in two-dimensions) that specified signed distances from the coordinate axis.

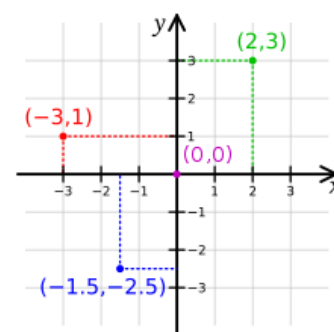


Figure 4. Cartesian Grid

In a similar way, we can think about a cartesian coordinate system representing the earth surface in a 2-dimensional plane. In this 2D geographic coordinate system the Y-axis represents the latitudes and the X-axis the longitudes. These coordinates represent the angular distance in decimal degrees geographical units. The latitudes are representing the angular distance from the Equator parallel (origin of the latitudes) and are limited up to 90 degrees North (positive values) and 90 degrees South (negative values), coincidence with location of the earth poles. The longitudes are representing the angular distance from Greenwich Meridian (origin of the longitudes) and are limited between 180 degrees to the east (positive values) and 180 degrees to the west (negative values).



Using this geographic coordinate system, we just need two values to represent a position (a point) in the sphere, the pair of coordinates representing longitude (x) and latitude (y). Examples:

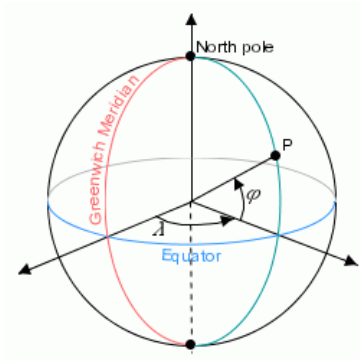


Figure 5. Geographic coordinate system

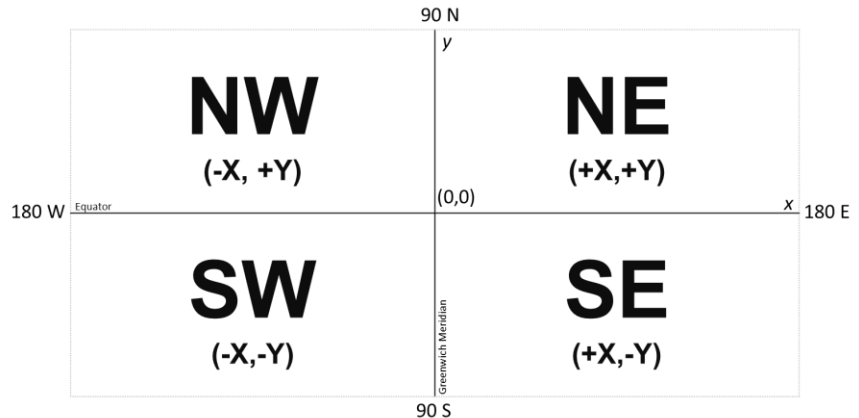


Figure 6. Representation of coordinates on cartesian grid

- Point in the North-East quadrant: (125, 45) = 125 degrees East and 45 degrees North
- Point in the North-West quadrant: (-125, 45) = 125 degrees West and 45 degrees North
- Point in the South-East quadrant: (125, -45) = 125 degrees East and 45 degrees South
- Point in the South-West quadrant: (-125, -45) = 125 degrees West and 45 degrees South
- 

Understanding how we can reference a unique position (a point) in the plane representing the terrestrial sphere, we can introduce the concept of spatial vector data. The main unit of the vector data is the point, and the other vector data types are lines and polygons. Lines consist of more than one point (vertex) and a path (line) that links these vertices. Polygons are formed by more than 2 points (vertex) and paths (edges) that link these vertices, in a polygon the last vertex is located in the same position than the first, creating a closed-shape feature.

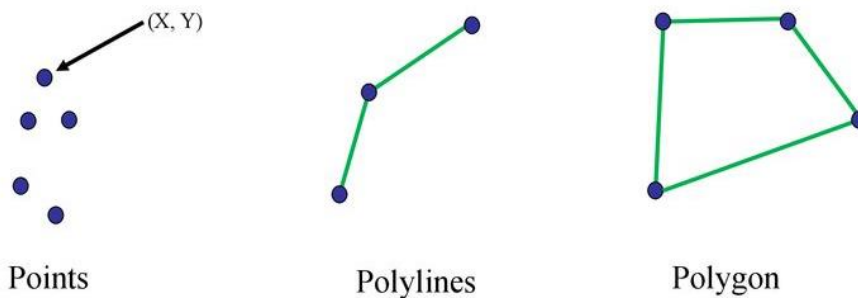
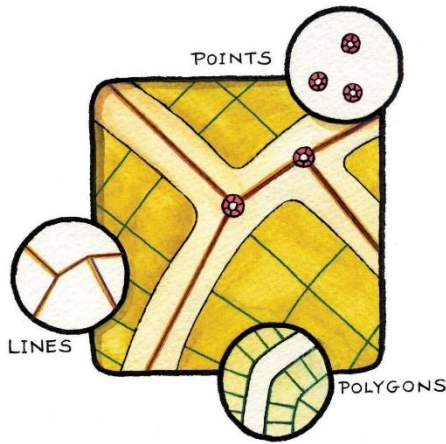


Figure 7. GIS representation of points, polylines, and polygons

# Vector Spatial Data Model

There are three fundamental vector types that exist in geographic information systems (GISs): **points, lines, and polygons**.



**Points** are zero-dimensional objects that contain only a single coordinate pair. Points are typically used to model singular, discrete features such as buildings, wells, power poles, sample locations, and so forth. Points have only the property of location.

**Lines** are one-dimensional features composed of multiple, explicitly connected points. Lines are used to represent linear features such as roads, streams, faults, boundaries, and so forth. Lines have the *property of length*. Lines that directly connect two nodes are sometimes referred to as chains, edges, segments, or arcs.

Figure 8. Points, Lines and Polygons

**Polygons** are two-dimensional features created by multiple lines that loop back to create a “closed” feature. In the case of polygons, the first coordinate pair (point) on the first line segment is the same as the last coordinate pair on the last line segment. Polygons are used to represent features such as city boundaries, geologic formations, lakes, soil associations, vegetation communities, and so forth. Polygons have the *properties of area and perimeter*. Polygons are also called areas.

Other types of point features include the node and the vertex. Specifically, a point is a stand-alone feature, while a node is a topological junction representing a common X, Y coordinate pair between intersecting lines and/or polygons. Vertices are defined as each bend along a line or polygon feature that is not the intersection of lines or polygons.

## Data attributes related to geographic information

The “Geographic Information System” refers to the computational systems to manage and analyse geographic data. A table is the most common format in computational science to store ordered information, where the rows are representing unique records, and columns contain information related to each record. As an initial example we could think of a table consisting, for example, of three rows and one column where the unique records are city names (Table 1).

Table 1. Non-spatial table

city
London
Buenos Aires
Beijing
Sydney

This table includes a minimum amount information just about the city names, however, we could scale this table both in number of rows, if we add new cities, or adding new columns with new information related to the existing cities in the table (Table 2).

In a table each column is commonly limited to a single unique data type (text, numeric, double, Boolean, etc). This means that if, for example, the column “City” is defined as a text data type any new record included in this column will be treated as text, even if we include a number, this would be treated as text, therefore we won’t be able to perform, for example, mathematical operations using the information in this column.

Following the previous cities table example, we are adding two new columns providing information about the city’s population and number of employed people. These two new columns are defined as numeric data type columns.

*Table 2. Extended non-spatial table*

city	population	jobs
London	8,962,000	6,148,000
Buenos Aires	2,891,082	1,254,321
Beijing	21,542,000	12,218,643
Sydney	5,312,163	2,272,722

Finally, we want to include in the table information related to geographic location of the city. Therefore, we need to add the pair of coordinates that reference the city’s location in the earth spheroid. Since the cities can be represented in a map as a discrete location, we are going to include the columns of “longitude” and “latitude” defining a point geometry pair of coordinates (Table 3).

*Table 3. Table with longitude and latitude coordinates.*

city	population	jobs	Longitude	latitude
London	8,962,000	6,148,000	0°7’39”W	51°30’26”N
Buenos Aires	2,891,082	1,254,321	58°22’54”W	34°36’12”S
Beijing	21,542,000	12,218,643	116°23’51”E	39°54’24”N
Sydney	5,312,163	2,272,722	151°12’34”E	33°51’54”S

The geographical coordinates can be provided in several formats, most of which are interpretable by the GIS software. However, independently of which coordinate format we choose, it must be a consistent format within the same table to be correctly read by the GIS software.

Most common geographical coordinates formats and equivalences are provided in the Table 4:

*Table 4. Conversion of degrees, minutes and seconds into decimal degrees that can be read as a location by GIS software.*

	Degrees, minutes, seconds	Degrees, decimal minutes	Decimal degrees	Decimal degrees signed
Longitude	79° 58’ 56” W	79° 58.933’ W	79.982° W	- 79.982
Latitude	40° 26’ 46” N	40° 26.767’ N	40.446° N	40.446



The format in which we receive geographical coordinates can vary based on the device used to obtain this information (e.g., GPS, maps, mobile phones, etc.), so it is common to receive location information from multiple sources. and it is important to verify that the geographical coordinates.

Consequently, in the table the city's coordinates were acquired from Wikipedia, and it was provided in "Degrees, minutes and seconds" format. Since this coordinate format includes characters for degrees, minutes, seconds, and a letter cardinal direction (N, S, E, W), this column is a text type column. To import this table into a GIS software and interpret the coordinates, it is preferable to transform this information into numeric values instead of text (Table 5).

*Table 5. Table with longitude and latitude converted into decimal degrees (numerical field).*

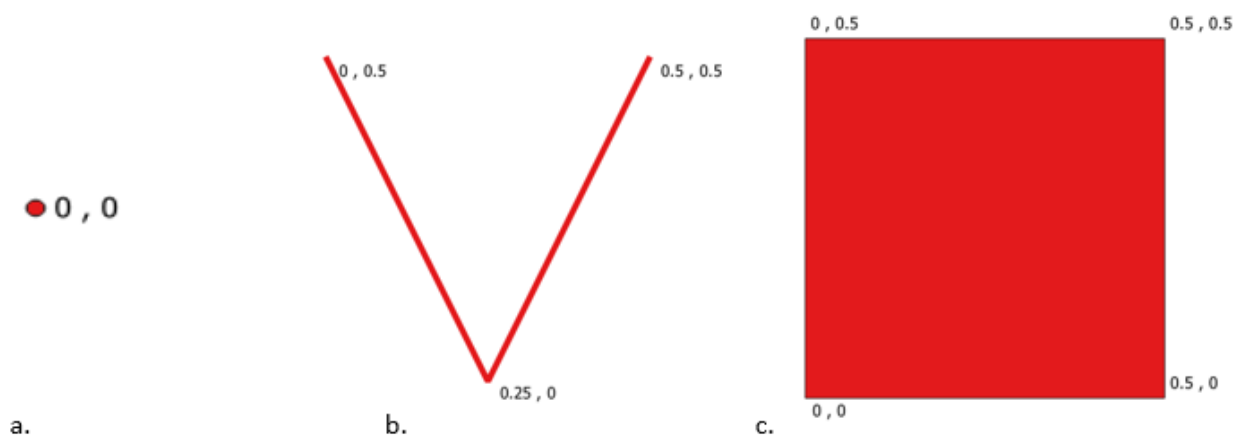
city	population	jobs	-longitude	latitude	longitude_decimal degrees	latitude_decimal degrees
London	8,962,000	6,148,000	0°7'39"W	51°30'26"N	0.1275	51.50722
Buenos Aires	2,891,082	1,254,321	58°22'54"W	34°36'12"S	-58.38167	-34.60333
Beijing	21,542,000	12,218,643	116°23'51"E	39°54'24"N	115.3975	39.90667
Sydney	5,312,163	2,272,722	151°12'34"E	33°51'54"S	151.2094	-33.865

## Geometry information as text:

Well-known text (WKT) is a way of representing vector geometry as text. There are three elements of WKT that can describe an objects geometry: **the type of geometry** (point, line, or polygon), **the coordinates of the vertices**, and **the spatial reference**. For example.

- ('POINT (0 0)')
- ('LINESTRING (0 0.5, 0.25 0, 0.5 0.5)')
- ('POLYGON ((0 0, 0 0.5, 0.5 0.5, 0.5 0, 0 0))',4326)

The above examples correspond to the following geometries:



*Figure 9. Representation of WKT as vector geometries*



The WKT format allows for vector geometry to be stored in a database, facilitating visualisation and analysis of geometries from within a geodatabase. Spatial databases store geometry data in a binary (numerical) format, that can be translated to and from WKT so it can be read and understood.

The coordinate system that describes the location of the vertices is a cartesian grid, plotting points on a flat, planar surface. The x and y coordinate refer to the Euclidean distance from a known point along the x and y axes.

The spatial reference determines where these points are positioned on the earth, as opposed to the flat surface of the cartesian grid. This is important when the curvature of the earth will affect calculations such as measuring distance, angle, or area. In well-known text, the spatial reference is represented by the EPSG number, in our example the EPSG is 4326, which represents the WGS84 (World Geodetic System 1984) coordinate reference system (CRS). The EPSG is a dataset that contains a collection of definitions and information on coordinate reference systems. These may be global, national, regional or local, but each CRS has its own unique EPSG.

## Storing vector data:

The geometry and attribute information of geographic vector features are most commonly stored as shapefiles. Shapefiles can contain multiple vectors of the same type, each with their own geometry and attribute data. When a shapefile is saved, supporting data is automatically created and stored in additional files alongside the shapefile. For each shapefile there are three mandatory files, each of these files must be present when stored and shared. These three files have unique extensions.

- .shp – The feature geometry, this is the file that is loaded into and edited within a GIS.
- .shx – An index file, used to search forwards and backwards.
- .dbf – A database file used to store attribute data and object IDs.
- .prj—The file that stores the coordinate system information.
- .cpg—An optional file that can be used to specify the codepage for identifying the character set to be used.
- .xml—Metadata for ArcGIS—stores information about the shapefile.

There are some other common formats for storing GIS data:

- geoJSON - An open standard format for representing geographical features and their associated attributes.
- CSV – A file where attributes and geometry data are stored as text, separated by commas.
- File Geodatabase – A collection of files capable of storing, querying, and managing spatial and non-spatial data. These are specific for ESRI products and can be created and used in ArcGIS.
- Geopackage – An open standard format for representing geographical features and their associated attributes capable of multiple layers in a single file.



# Raster Spatial Data Model

A raster dataset is an array of rectangular cells with each cell containing a value that represents data. This matrix of cells (or pixels) is organised into rows and columns.

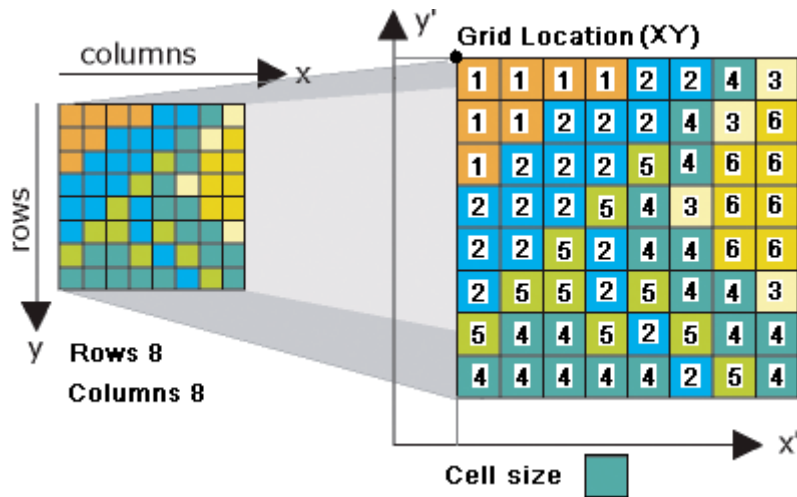


Figure 10. Thematic raster with associated values

The data in a cell is known as a Digital Number (DN) or pixel value and can represent a wide variety of geographical data. Raster data can represent discrete thematic data categorized into a classification such as substrate type (a) and can also represent continuous data such as temperature, depth, or spectral data (b).

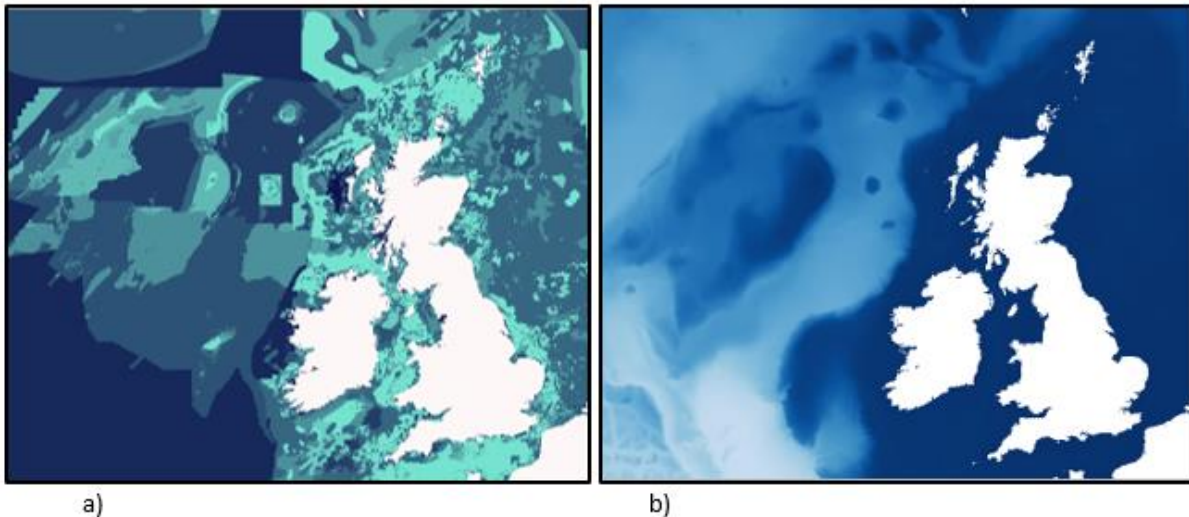


Figure 11. a) discrete substrate data and b) continuous bathymetry.

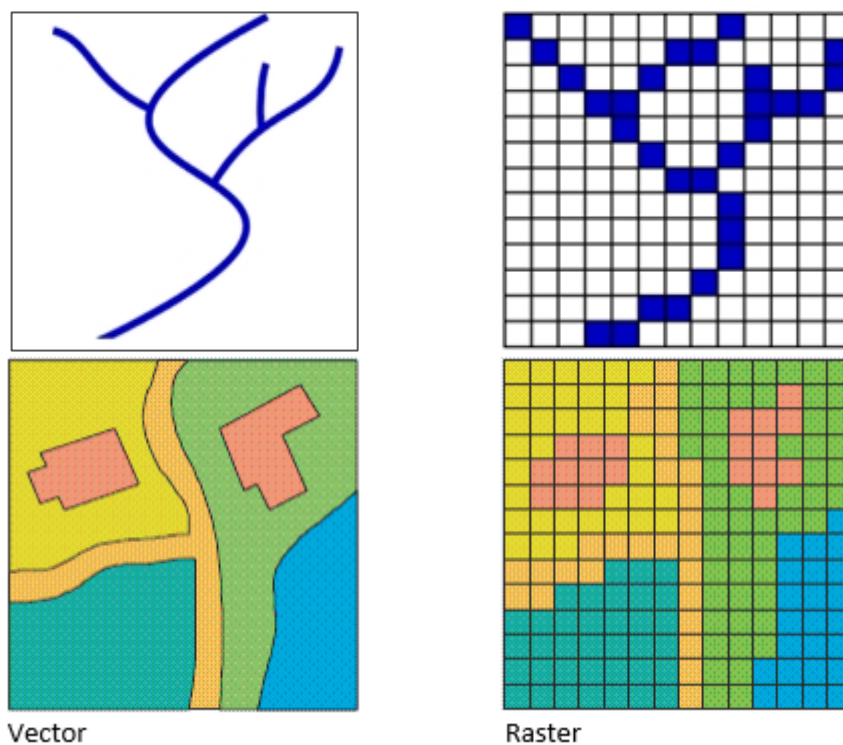
Thematic rasters with pixel values that represent a class or category will also contain a raster attribute table. When generated, a raster attribute table will contain data on the value of each pixel and the number of cells with that value within the raster. Additional data can be added to a raster attribute table, such as a class name, code, or description.

## Storing Raster Data

There are many raster file formats that differ in type of compression, attribute table compatibility and supported data types. Common examples of raster file formats include.

- .tif
- .img
- .netcdf
- .jpeg

## Raster vs Vector Data



*Figure 12. Objects represented in the vector and Raster format.*

The above images represent the same real-life objects presented as vector and raster datasets. The accuracy of raster data is determined by the spatial resolution. The spatial resolution of a raster is the size each cell represents on the ground. The quality of a vector is determined by the accuracy of the digitization process that created the polygons or the accuracy of the coordinates used to create the vertices.

# Simple geometry operations

Having geometry information stored in a table allows us to perform geometric and geoprocessing operations to further analyse and interpret our data. Below are some examples of simple geometry operations that are commonly used in GIS.

**Intersection** – returns the portion of two or more geometries that are shared or overlap. All features that overlap will be part of the output feature, and their attributes will be preserved.

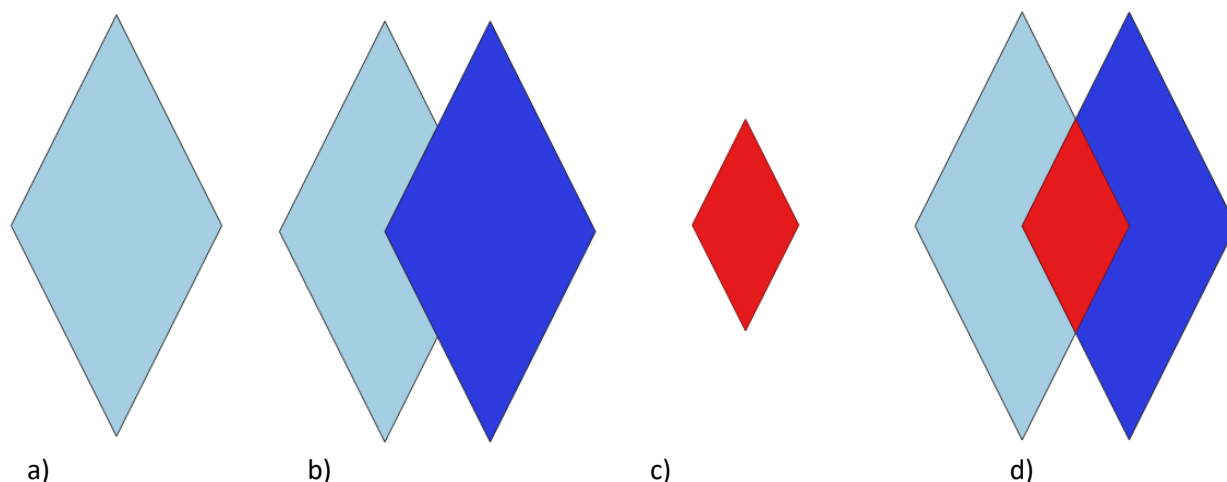


Figure 13. Example of QGIS intersection operation

We start with a single polygon (a) and add a second polygon with a portion that overlaps our first (b). Performing an intersection creates a third polygon (c) which is the portion of polygons a and b which overlap (d).

**Union** – joins geometries and returns a new geometry that is the combination of the inputs.

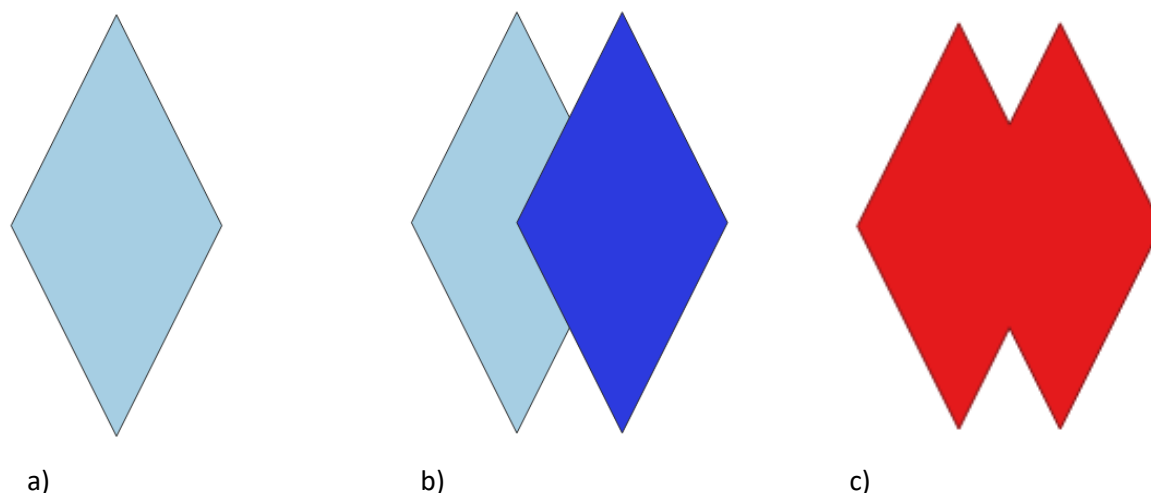


Figure 14. Example of QGIS Union operation



**Buffer** – returns a new geometry where the vectors are a specified distance from our existing geometry's vectors. It creates a buffer 'around' our existing geometries. The result of a buffer is a polygon, despite the input geometry.

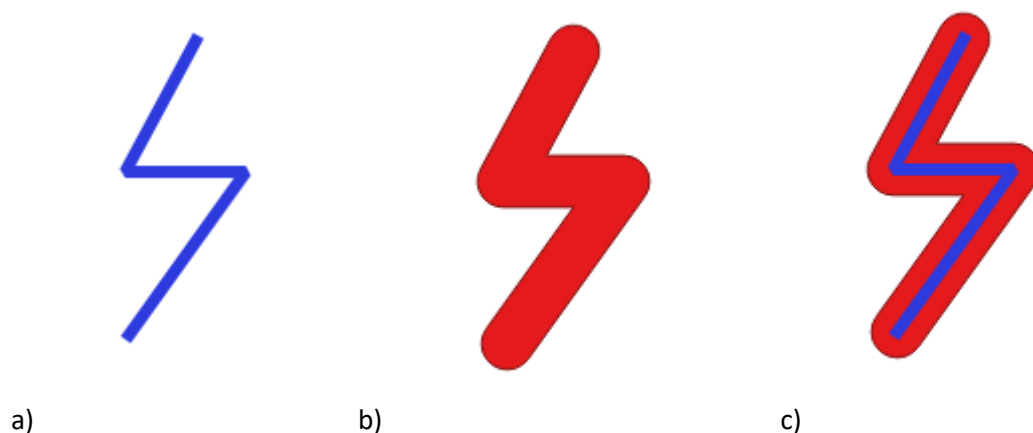


Figure 15. Example of QGIS buffer operation

Our output (b) is a polygon that is the same shape as our input (a) but with the vectors a specified distance away from those in the original. Figure c) shows our input on top of our output.

## Introduction to GIS using QGIS

### Installing QGIS

#### For Cefas staff:

The most recent version of QGIS can be downloaded from the software centre.

#### For all other users:

QGIS can be downloaded from <https://www.qgis.org/en/site/forusers/download.html> and selecting the most recent, or long-term release for your operating system.



If you don't know whether you need the 32- or 64- bits version of QGIS, you can check on your computer by going to **Start > Settings > System > About** and checking the System type. You will want to download the version that matches your system's operating system.

## 1. Setting up a Project

### 1.1. Create a folder in which to save your data and QGIS project.

It may be useful within that folder to have separate folders for different data types. This will make finding your data easier in the future and will help maintain a consistent folder structure.

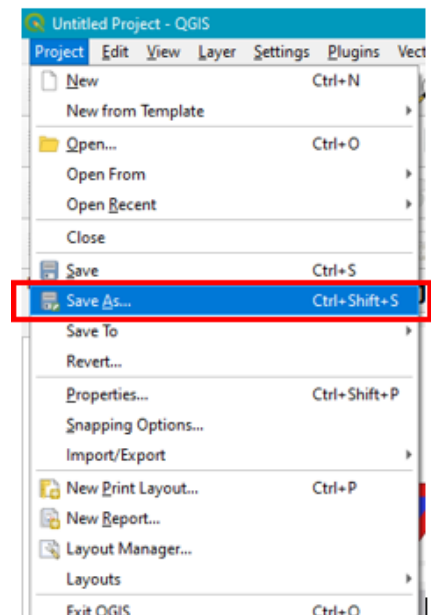
Name	Date modified	Type
01 CSV	06/02/2020 10:50	File folder
02 Vector	06/02/2020 10:50	File folder
03 Raster	06/02/2020 10:50	File folder
04 Export	06/02/2020 10:50	File folder

### 1.2. Open QGIS

Open the QGIS desktop application by searching or opening in the start menu of your computer.

### 1.3. Create a new project and save

When you open QGIS, you will be starting a new project. To save this project navigate to **Project > Save As**. Save this project in your newly created folder as a **.qgz** file, when you have data in your project you can save the project to save the layers with their current settings. **Use informative project names so you can identify the project for them name.**

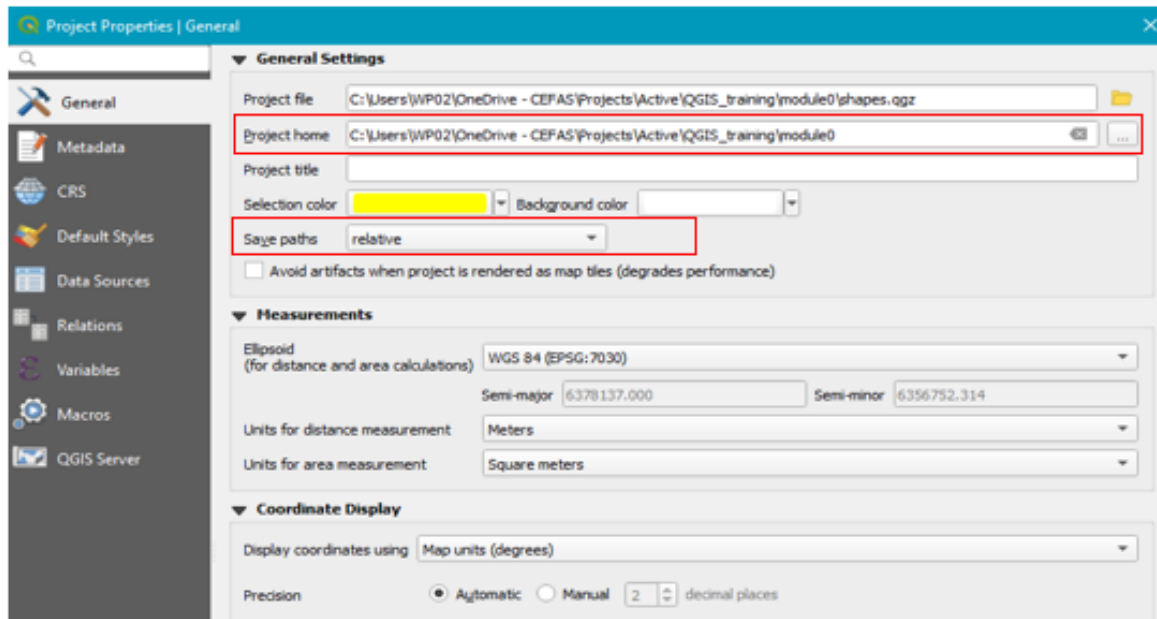


### 1.4. Set Project properties

- Go to **Project > Properties > General**.
- Ensure your Project Home is set to your newly created folder.
- Set 'Save Paths' to relative.
- Click 'OK' at the bottom of the page to save these settings.

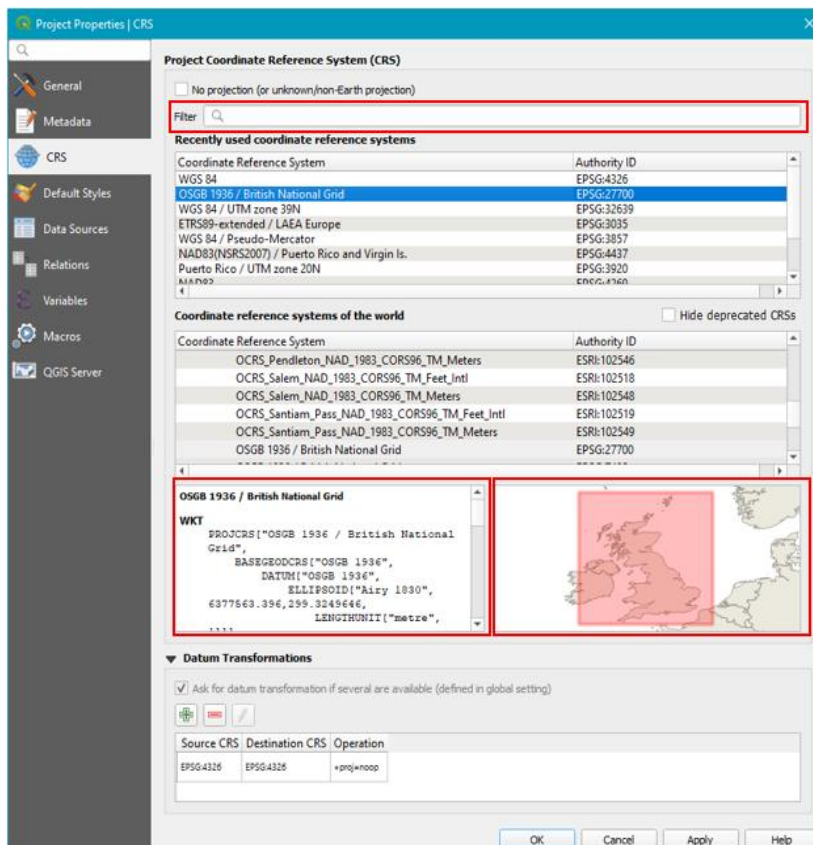


Here you can also play around with other settings, including the how the coordinates are displayed in the project, and the highlight colour for selected items.



### 1.5. Set Coordinate Reference System

For any project you will need to set a Coordinate Reference system (CRS), this will determine the projection or coordinates used for layers in your project.




- Go to **Project > Properties > CRS**

- Search for your desired CRS in the filter bar. Search by using the name of your CRS or the EPSG number if known e.g., OSGB 1936 / British National Grid – EPSG:27700

- The map in the bottom right of the page will show you what area the CRS is designed to be used with. The window to the left of the map will give more information on the selected CRS. Click 'OK' to save the CR settings.

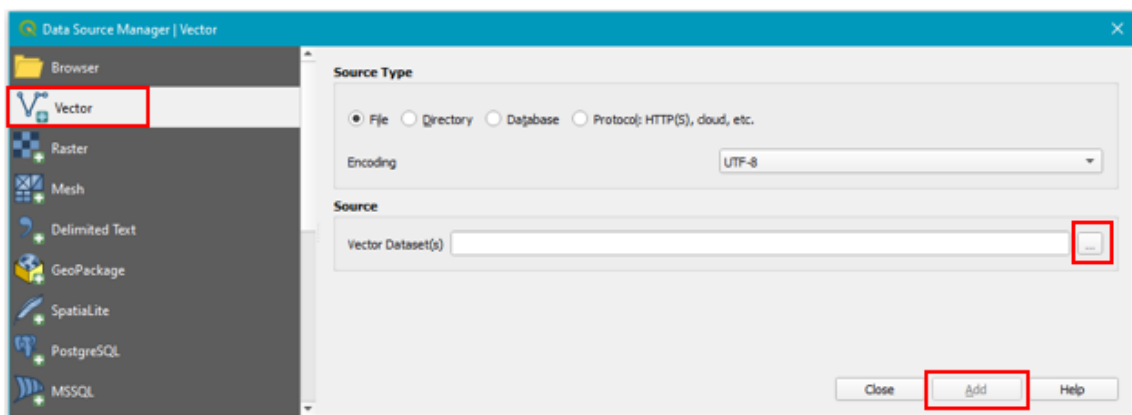
## 1.6. Save project

Click **Project > Save** to save this project with these settings, click the  button, or simply use **Ctrl + S** on your keyboard.

## 2. Loading data into QGIS

### 2.1. Load Vector Data

Click the 'Open Data Source Manager' button or press 'Ctrl + L' to open the window and load data.



To load vector data into your QGIS project, navigate to the 'Vector' panel on the left of the window. Use the button to search your files for vector data to add. Once data has been selected, click 'Add' to load the data into your project. Vector data can also be 'dragged and drop' directly into your QGIS window from the folder where your data is stored. Try this with the 'World\_boundaries.shp' file in the data folder.

### 2.2. Load Raster Data

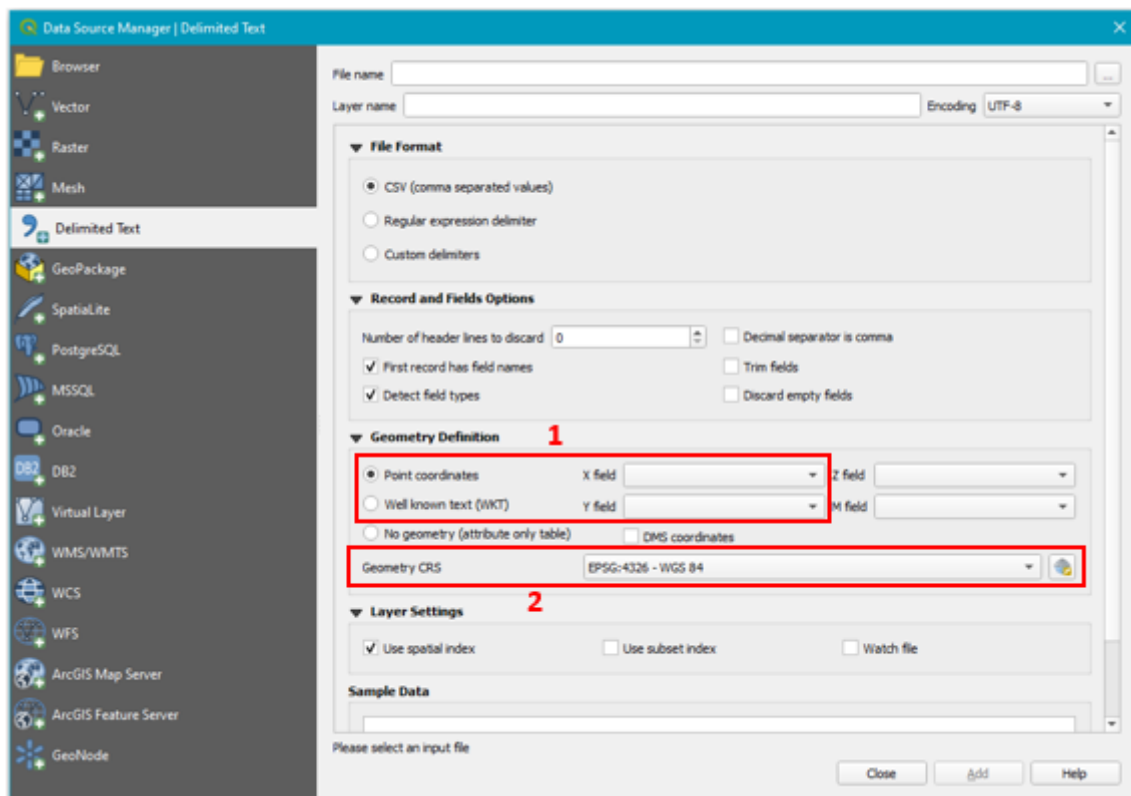
To load raster data, follow the instructions for loading vector data, but select the 'Raster' panel on the left of the window. Raster data can also be 'dragged and drop' directly into your QGIS window from the folder where your data is stored.

### 2.3. Load CSV Data

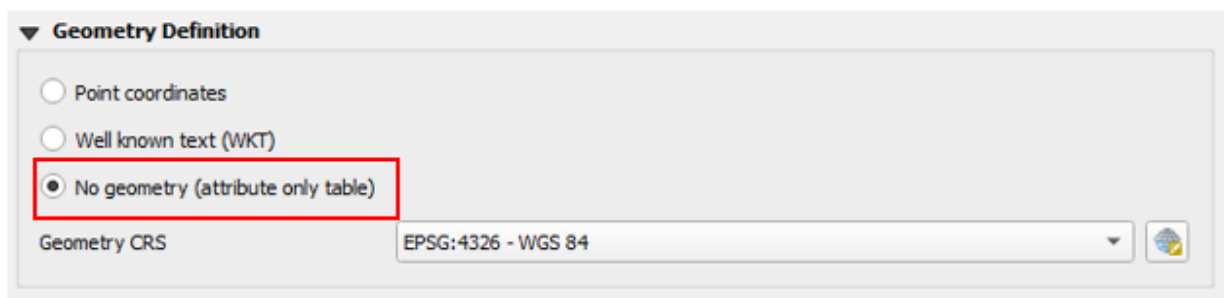
To load CSV data, follow the instructions for loading vector data, but select the 'Delimited Text' panel on the left of the window. CSV data that contains latitude and longitude columns or other coordinate values can be loaded into QGIS as a point vector dataset. Alternatively, CSV data can be loaded into QGIS as an attribute table, so will have no geometry.

- To load CSV data with geometry, you must identify the latitude and longitude columns within the dataset (1). To populate the 'X field', select the longitude column within your dataset and use the latitude column to populate the 'Y field'.
- We can also select the Coordinate Reference System that we want to display our newly created points in (2).





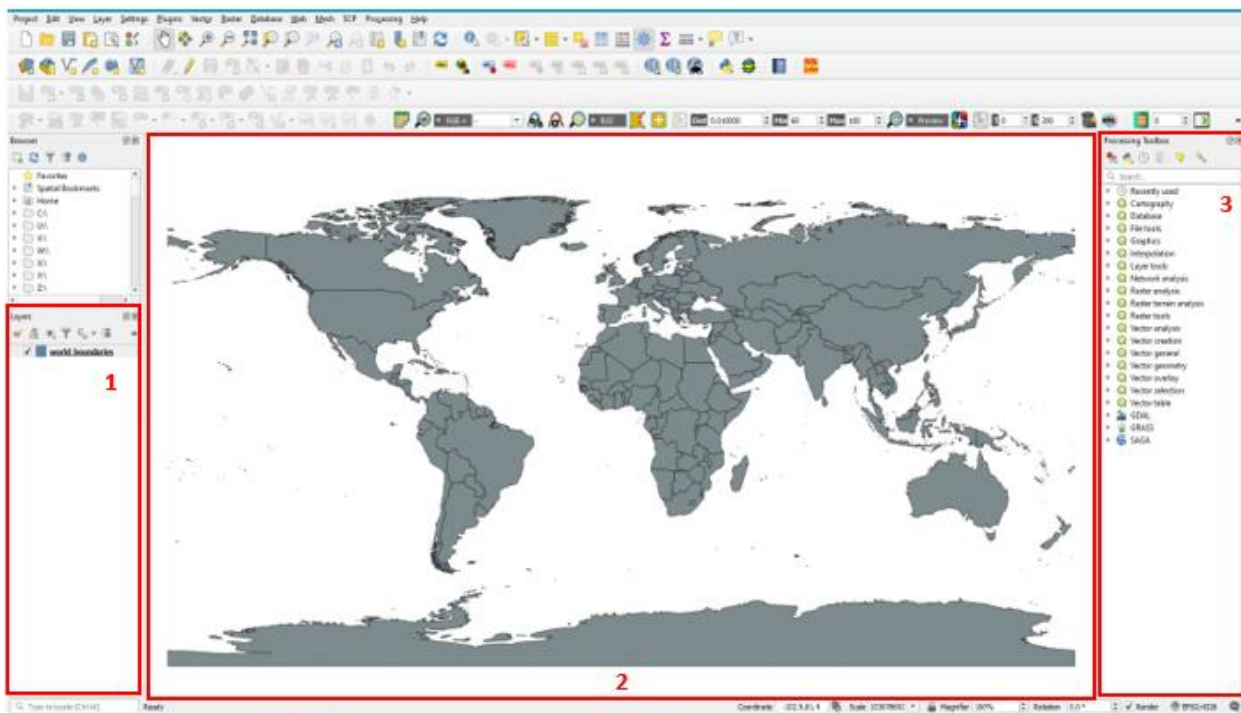
- To add the new geometry data to your project, click 'Add'.
- To load CSV data with no geometry as an attribute table, check the button 'No geometry'.
- Click 'Add' to load the table into your QGIS project.



## 2.4. Loaded Data

Loaded data will appear as a new Layer in your QGIS Project. Layers are added to the 'Layer' panel (1) and can be visualized in the main panel (2) of QGIS. Available toolboxes can be selected in the 'Processing Toolbox' (3).

Additional toolboxes can be added to the QGIS window in **View > Panels** and checking desired panels.



### 3. Visualizing Data

Interacting with your data

Within QGIS we can interact with our spatial data in many ways, the simplest of these are panning, zooming, and selecting features.



Panning



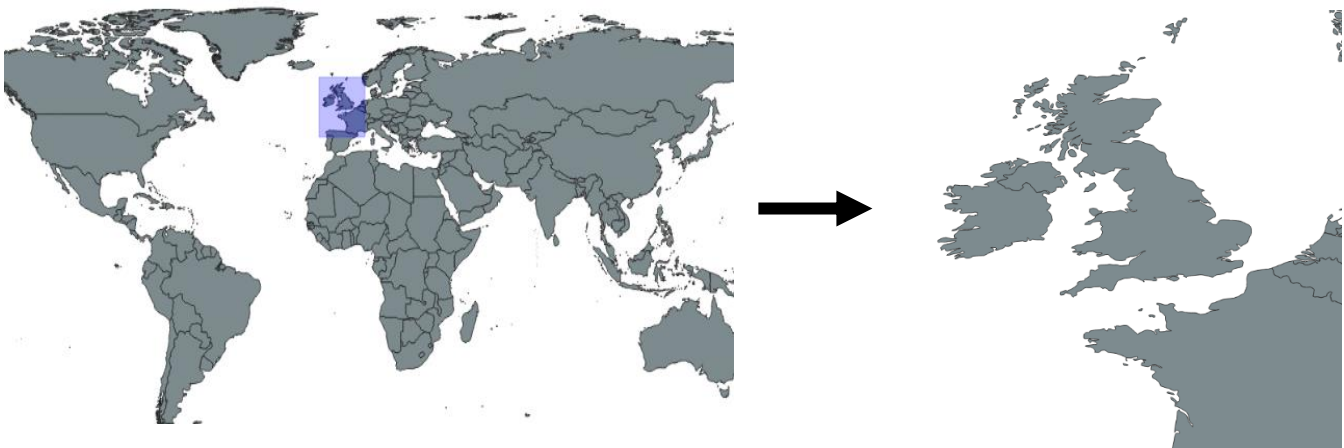
Panning allows the user to move around the map and data by clicking and dragging on the map image. By doing this we can explore our data and move to different areas of our maps.

Zooming

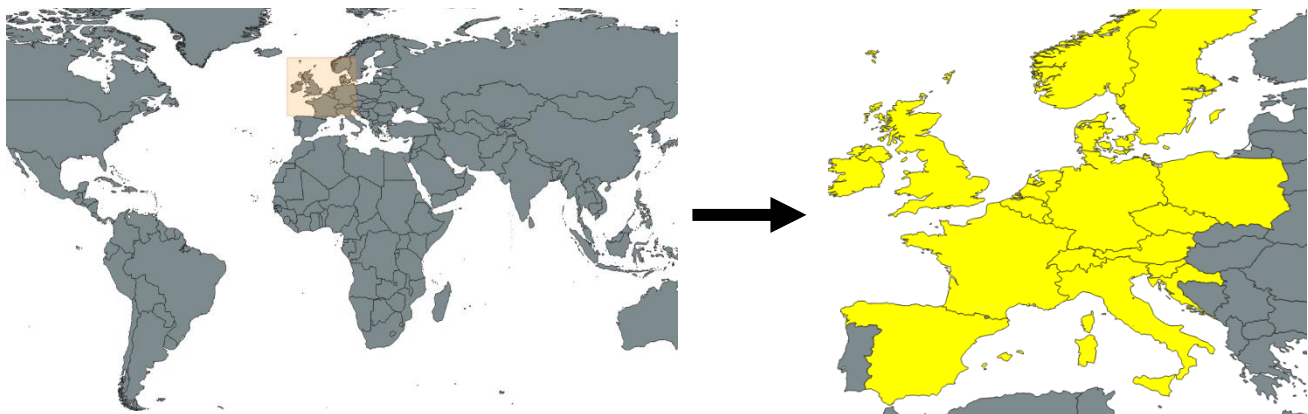


We can zoom in or out of our images using the zoom buttons. If we select a zoom button, we can draw a polygon around our area of interest and zoom in or out of that area. Alternatively, you can zoom using the scroll function of the mouse.





Selecting features in QGIS can provide information on their associated attributes or can allow us to perform analysis or processes on selected features only.



This dropdown option allows us to select features by clicking on them. Alternatively, we can use the mouse to draw a rectangle or polygon around an area of interest, this will select all features that intersect our selection area.



This dropdown options allows us to select all features in our layer. We can also select features from by value or expression, where all features that match the desired values will be selected.

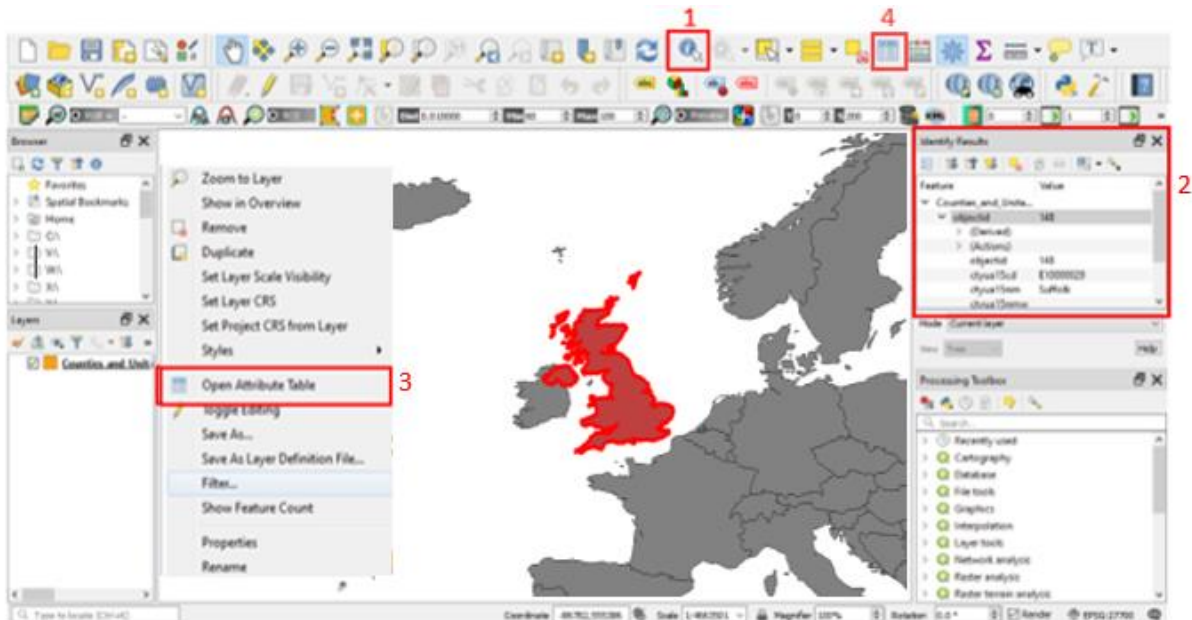


This button will deselect all selected features.



## 4. Using Attribute Tables

The attributes of an individual feature can be visualized by clicking on the feature using the **Identify Features cursor** (1) and viewing attributes in the **Identify Results panel** (2). Alternatively, all features can be viewed in the attribute table by right-clicking on the feature layer in the Layers panel and selecting **Open Attribute Table** (3), clicking the **open attribute table tab** (4) or pressing F6.



World\_boundaires :: Features Total: 241, Filtered: 241, Selected: 0

	SOVEREIGNT	SOV_A3	ADM0_DIF	LEVEL	TYPE	ADMIN	ADM0_A3	GEOU_DIF	GEOUNIT
13	Australia	AU1	1	2	Dependency	Heard Island an...	HMD	0	Heard Island an...
14	Australia	AU1	1	2	Dependency	Indian Ocean T...	IOA	0	Indian Ocean T...
15	Austria	AUT	0	2	Sovereign coun...	Austria	AUT	0	Austria
16	Azerbaijan	AZE	0	2	Sovereign coun...	Azerbaijan	AZE	0	Azerbaijan
17	Bahrain	BHR	0	2	Sovereign coun...	Bahrain	BHR	0	Bahrain
18	Bangladesh	BGD	0	2	Sovereign coun...	Bangladesh	BGD	0	Bangladesh
19	Barbados	BRB	0	2	Sovereign coun...	Barbados	BRB	0	Barbados
20	Belarus	BLR	0	2	Sovereign coun...	Belarus	BLR	0	Belarus
21	Belgium	BEL	0	2	Sovereign coun...	Belgium	BEL	0	Belgium

Show All Features


Within the attribute table, editing can be done by clicking on the pencil in the top left corner (1). This allows the user to delete features and edit attribute data. Data can be selected using queries within the attribute table pane (2). Columns can be added and removed (3), and fields can be generated manually or based on equations or geometry (4).





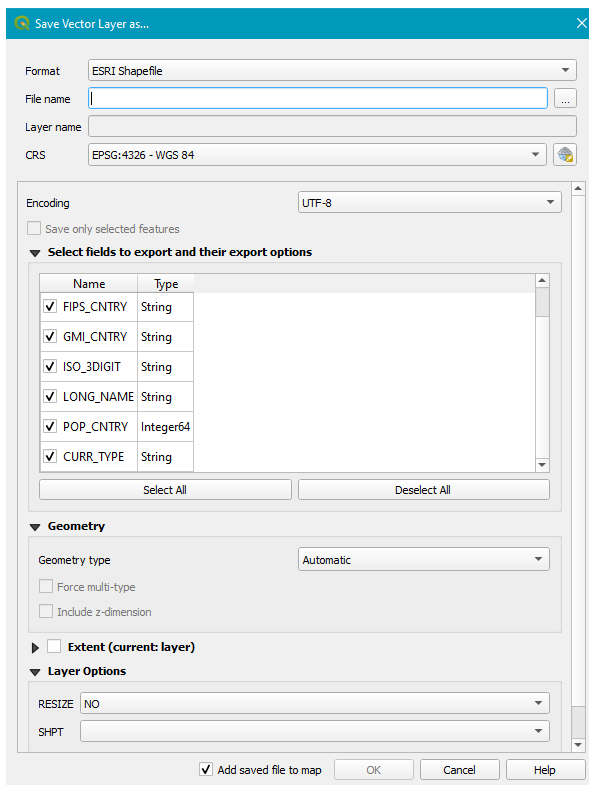
## 5. Saving and Exporting Data

### 5.1. Saving a Project

Saving your project will save all the layers in your QGIS project in their current states, as well as all the settings, such as CRS and any preferences. To save your project you can click **Project > Save**, click the  button, or simply use **Ctrl + S** on your keyboard.

### 5.2. Saving your data

To export a vector or raster layer; right click on your layer and select **Export > Save Feature As**. Here you can select the format, name, location, extent, and CRS of the layer being exported.



**Save Vector Layer as...**

Format: **ESRI Shapefile**

File name:

Layer name:

CRS: **EPSG:4326 - WGS 84**

Encoding: **UTF-8**

☐ Save only selected features

**Select fields to export and their export options**

Name	Type
<input checked="" type="checkbox"/> FIPS_CNTRY	String
<input checked="" type="checkbox"/> GML_CNTRY	String
<input checked="" type="checkbox"/> ISO_3DIGIT	String
<input checked="" type="checkbox"/> LONG_NAME	String
<input checked="" type="checkbox"/> POP_CNTRY	Integer64
<input checked="" type="checkbox"/> CURR_TYPE	String

**Geometry**

Geometry type: **Automatic**

☐ Force multi-type

☐ Include z-dimension

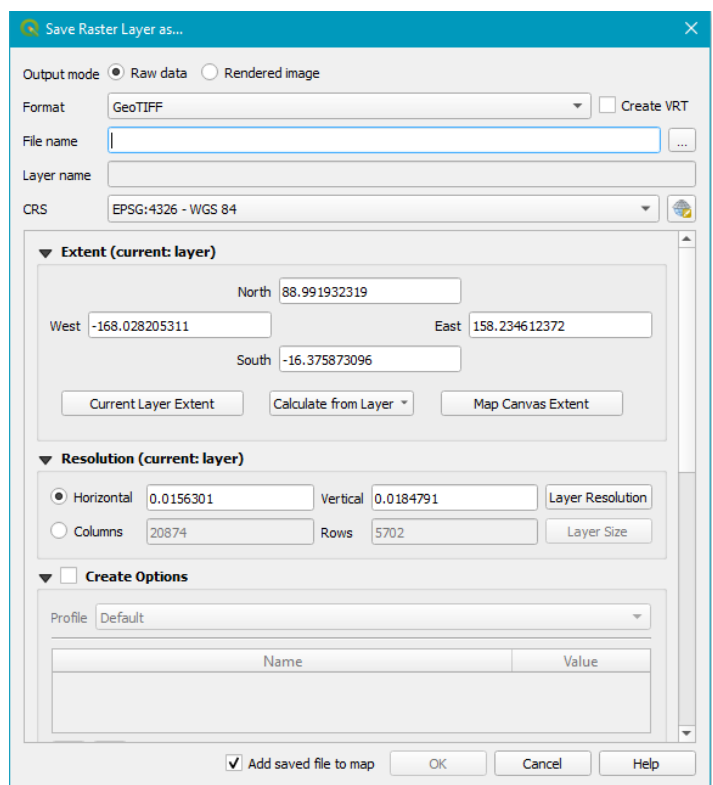
☐ **Extent (current: layer)**

**Layer Options**

RESIZE: **NO**

SHPT: **POINT**

☒ Add saved file to map



**Save Raster Layer as...**

Output mode: ☒ Raw data ☐ Rendered image

Format: **GeoTIFF** ☐ Create VRT

File name:

Layer name:

CRS: **EPSG:4326 - WGS 84**

**Extent (current: layer)**

North:

West:  East:

South:

**Resolution (current: layer)**

☒ Horizontal

☐ Columns

☐ **Create Options**

Profile: **Default**

Name	Value

☒ Add saved file to map

## Links

### Figures in text:

Figure 1: [https://saylordotorg.github.io/text\\_essentials-of-geographic-information-systems/s05-01-spatial-thinking.html](https://saylordotorg.github.io/text_essentials-of-geographic-information-systems/s05-01-spatial-thinking.html)

Figure 4: [https://en.wikipedia.org/wiki/Cartesian\\_coordinate\\_system](https://en.wikipedia.org/wiki/Cartesian_coordinate_system)

Figure 5: [https://help.perforce.com/visualization/jviews/8.10/jviews-maps810/doc/html/en-US/Content/Visualization/Documentation/JViews/JViews\\_Maps/\\_pubskel/ps\\_usrintmaps81.html](https://help.perforce.com/visualization/jviews/8.10/jviews-maps810/doc/html/en-US/Content/Visualization/Documentation/JViews/JViews_Maps/_pubskel/ps_usrintmaps81.html)

Figure 7:

<https://slideplayer.com/slide/17655716/105/images/19/GIS+Theory+%E2%80%93+Spatial+Data+Types.jpg>

Figure 8: [https://saylordotorg.github.io/text\\_essentials-of-geographic-information-systems/s08-02-vector-data-models.html](https://saylordotorg.github.io/text_essentials-of-geographic-information-systems/s08-02-vector-data-models.html)

Figure 10: <https://desktop.arcgis.com/en/arcmap/10.3/manage-data/geodatabases/raster-basics.htm>

Figure 12: <https://desktop.arcgis.com/en/arcmap/10.3/manage-data/geodatabases/raster-basics.htm>,  
<https://mgimond.github.io/Spatial/chp02-0.html>

### Material for cover page:

1. <https://images.app.goo.gl/oqQ2ies3WvbBNZDA9>
2. <https://moderndiplomacy.eu/2019/02/26/seize-the-opportunities-of-digital-technology-to-improve-well-being-but-also-address-the-risks/>

