



Automated Map Production in QGIS: Microplanning Atlas Case Study

Contents

Introduction.....	2
Part 1 - Setting up Atlas configurations	3
Part 2 - Dynamic page titles	6
Part 3 - Preview your atlas maps.....	9
Part 4 - Controlling the map extent using Atlas.....	10
Part 5 - Dynamic scale bar	10
Part 6 - Dynamic symbology to highlight map extent	11
Part 7 - Additional dynamic map elements.....	14
Challenge 1: Create an appropriate overview map.....	17
Challenge 2: Create an advanced overview map and feature-based extent indicator.....	18
Challenge 3: Create rule-based labels to display health facility names.....	19
Part 8 - Publishing your Map Atlas.....	20



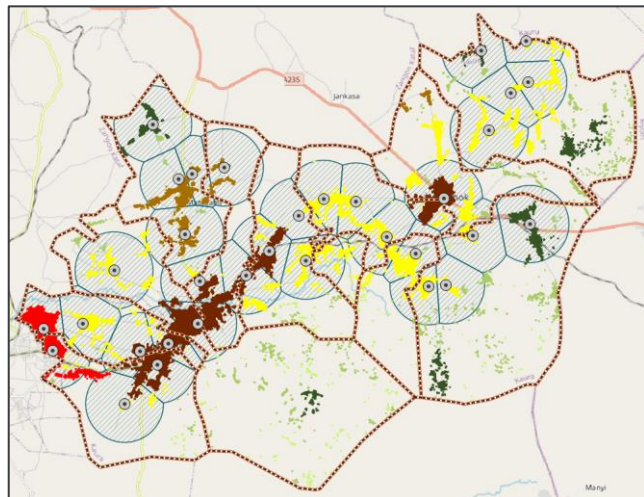
Automated Map Production in QGIS: Microplanning Atlas Case Study

Introduction

This exercise builds on the course *Supporting Microplanning with GIS and Spatial Data* in which you used GIS and spatial data analyses of communities to improve coverage of, and access to health facilities. In case study 2, the objective was to evaluate settlements that fall within a 5km buffered distance of health facilities in two local government areas (LGAs) in Nigeria. The examination of this data for Takai and Sumalia LGAs, in Kaduna State, focused on interpreting settlement patterns and highlighting population clusters that currently lack coverage. Such clusters become priority locations for the placement of new health facilities, or temporary health posts, to enhance service accessibility.

Scenario:

Another LGA in Kaduna State is planning a field survey to assess the location of existing health facilities. You are the supervisor for the health facility location microplan for Kaura and are overseeing the field survey team. The LGA is divided into 10 wards. You currently have an *LGA Map* covering the extent of the 10 wards which clearly displays their boundaries (see map on right). You have been asked to show each ward in more detail by generating an additional 10 maps, each map covering the extent of one ward.



Creating 10 separate maps is a timely process. Using the Atlas function in QGIS will speed up this process and allow you to create separate maps for each ward using one template.

By the end of this session, you will be able to:

- Configure your Map Atlas
- Map title iteration
- Preview your atlas
- Automate map features
- Use advanced features – dynamic masking and map elements



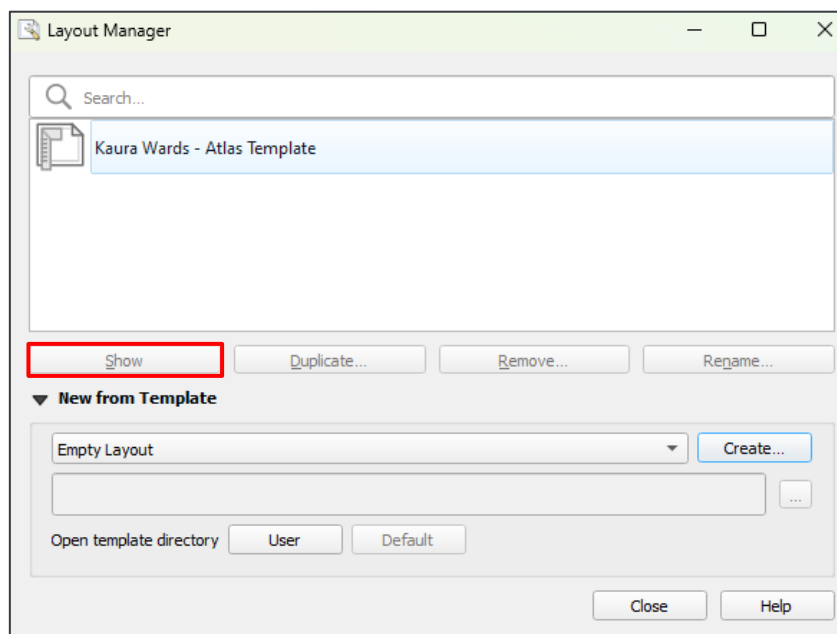
Part 1 - Setting up Atlas configurations

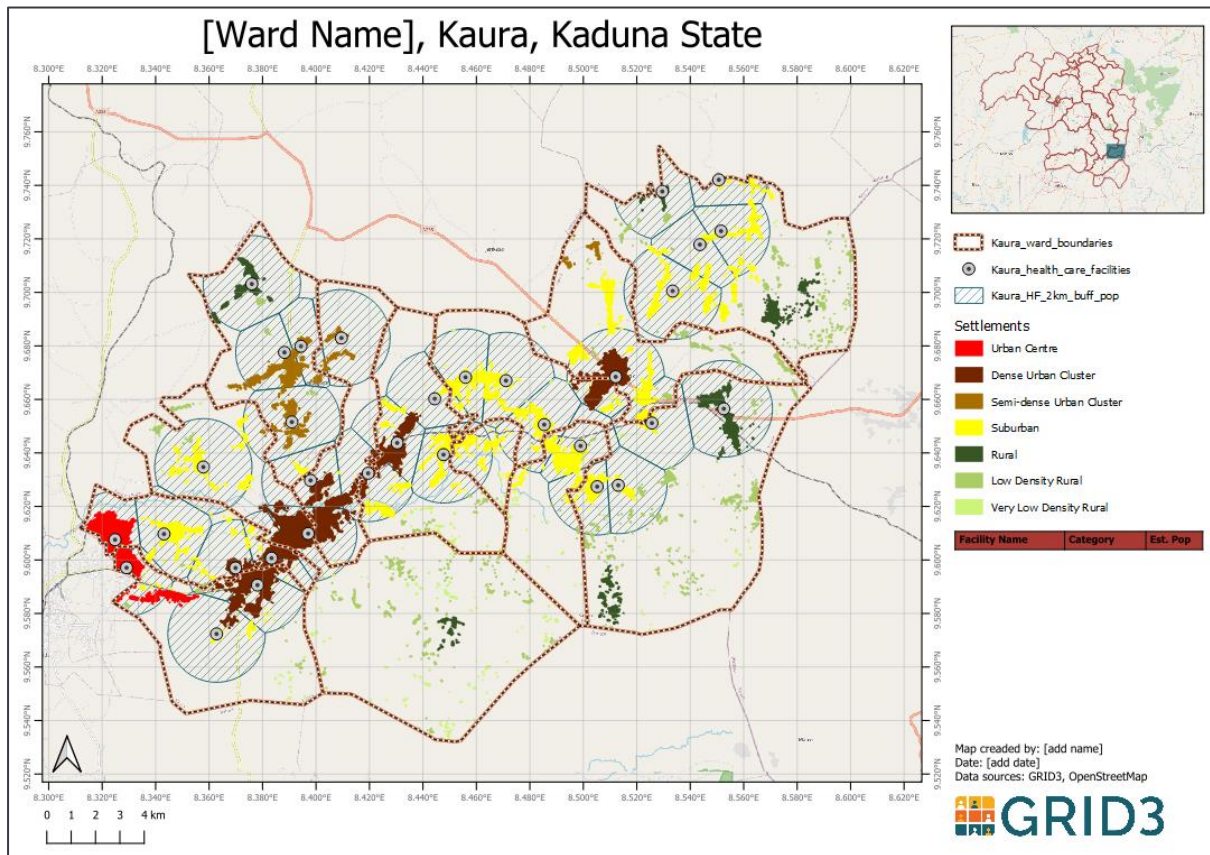
1. Open a QGIS project file

- Open the following project: `C:\G3_GIS_Training\Automated_Map_Production_Exercise.qgz`

Displayed in your QGIS window you'll see the Kaura ward boundaries, health facility point locations, settlement extent polygons and health facility 2km buffers.

- From the **Project** drop-down menu, select **Layout Manager**,
- Select **Kaura Wards - Atlas Template** and click **Show**





This is the LGA map you have been provided. It shows the Kaura ward boundaries, health care facilities, the 2km buffer zone around each health facility and settlement clusters classified into 7 different settlement types. From this, you need to create the 10 individual ward microplanning maps. You can automate this process by using the QGIS map atlas.

This map template has been created using processes covered in the GRID3 QGIS Foundations programme (Basic Map Production course). To create a similar template, use the 'toolbox' toolbar in the *Print Layout* window.

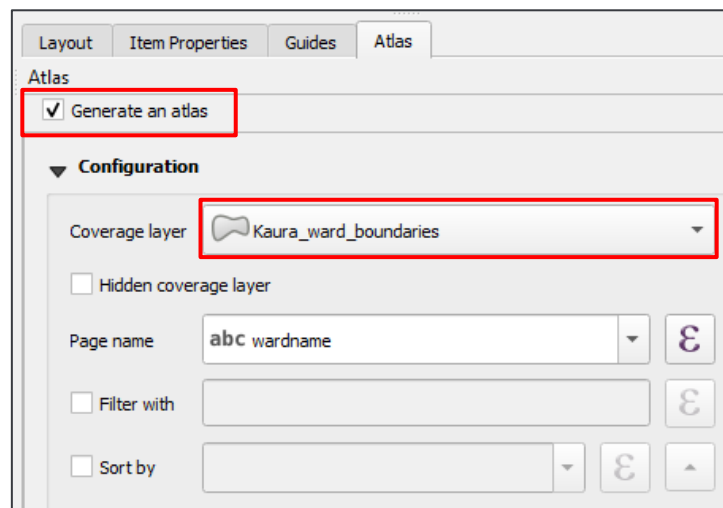


Please note: Attribute tables showing population counts or other information can be added to the map layout directly, using the highlighted button above.



2. Setting up a Map Atlas

- Locate your **Atlas** tab in the right-hand panel of the layout window (see screenshot below – if it is not visible, enable it from **View > Panels > Atlas**)
- Tick the box next to **Generate an atlas**
- Within the Configuration section we need to set the **Coverage layer**. This is the layer in the map that will drive the Atlas extents. In this exercise this will be the health areas polygon layer. From the **Coverage layer** drop down menu select *Kaura_ward_boundaries*



There are further options within the Configuration section to drive the atlas including a hidden layer – to hide the features in the **Coverage layer** so they don't appear in your map but still determine the extents – and also to filter the coverage layer – for example, if you have a settlements coverage layer you might wish to create maps just for towns and not villages; an expression could be written to query out just the towns. Another important setting is the page name – this determines how the Atlas tool identifies each individual map – we will use the name provided for each health area.

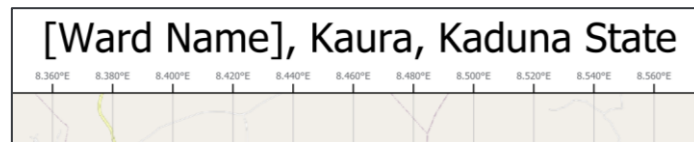
- For **Page name** select *wardname*



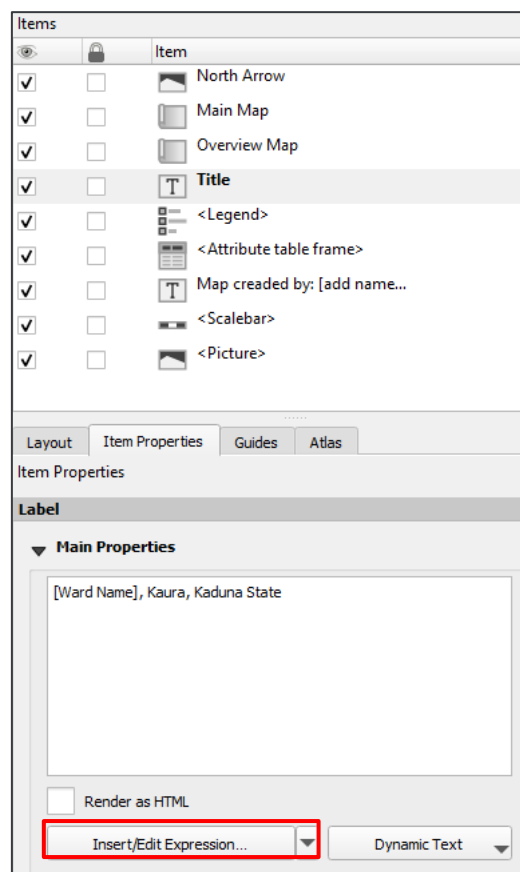
Part 2 – Dynamic page titles

Now turn to the map title shown below. The text *[Ward Name]* is a placeholder. We need each map in the atlas to display the name of the relevant ward, and we don't want to manually edit this for each map!

To achieve this automation, we need to write an expression.



1. Build an expression to automate map titles
 - In the **Items** list (i.e. the layers or components of your map) click on *Title*, as shown below
 - Within the **Item Properties** tab click on the **Insert or Edit an Expression** button as shown below:

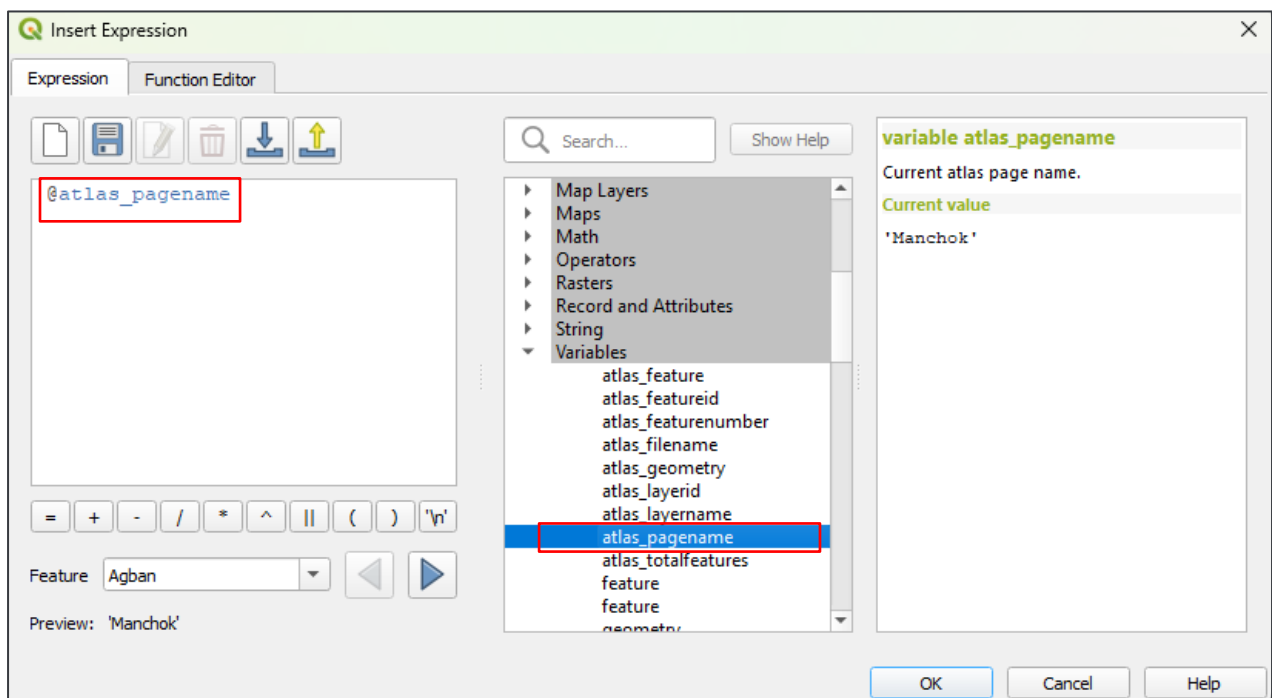


- The **Insert Expression** dialog box should pop up
- Within the functions section, in the central section of the window, expand the **Variables** section.



As stated by the **Help** section, “This group contains dynamic variables which can be inserted into your expressions”. Each of the functions in this list prefixed with ‘atlas_’ are written specifically to hook into the Atlas features and enable a wide range of map automations.

atlas_pagename is a variable that returns the current atlas page name , you will use this now.

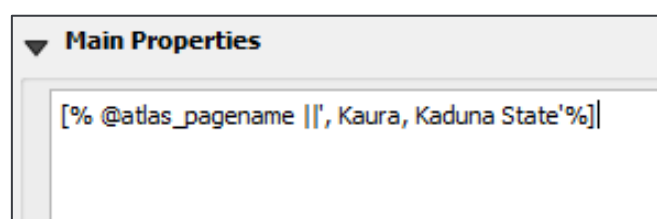


- Double-click **atlas_pagename**; it will appear as `@atlas_pagename` in the expression window
- Continue to type the following expression into the expression box to create the full expression shown below

`||', Kaura, Kaduna State'`

`@atlas_pagename ||', Kaura, Kaduna State'`

- Click **OK**
- The expression should now appear in the box underneath the Main Properties window, with additional syntax added automatically ‘[% %]’. The original Map Title text is also in the box. Delete this so the box appears as shown below:





The output maps will now be titled with the relevant health area name. Next, we will preview the Atlas.



User tip: it is well worth studying the help documentation for each of the atlas functions. Getting familiar with these and mastering their use and application unlocks a very powerful side of QGIS, not only in map production and cartography but also a wide range of data processing, querying and geoprocessing tasks. Click here to see the *Variables* section of the online QGIS User Guide*: [14.3.25. Variables](https://docs.qgis.org/3.22/en/docs/user_manual/)

Variable Name	Description
@atlas_feature	Returns the current Atlas feature that is iterated. Can be used with the 'attribute' function to return attribute values from the current atlas feature.
@atlas_featureid	Returns the feature id of the current row while using atlas. Enables the use of features of the atlas in rules e.g. to coordinate with other layers in the map.
@atlas_featurenumber	Returns the current feature number that is iterated over in the Atlas.
@atlas_filename	Returns the filename that will be used for the current atlas page if exported
@atlas_geometry	Returns the geometry of the current Atlas feature that is iterated. Enables the use of Atlas geometries in rules e.g. only show geometries of other layers when their geometry intersects the iterated Atlas feature.
@atlas_layerid	Returns the current atlas coverage layer ID
@atlas_layername	Returns the Atlas coverage layer
@atlas_pagename	Returns the current Atlas page name that is iterated over in the Atlas.
@atlas_totalfeatures	Returns the total number of features within the coverage layer.

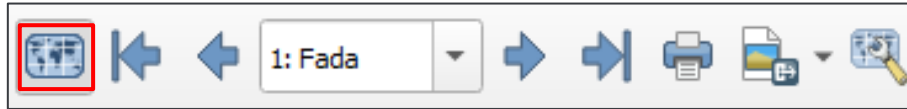
* QGIS User Manual home page: https://docs.qgis.org/3.22/en/docs/user_manual/



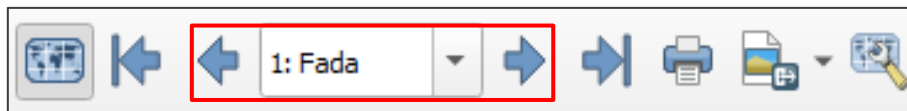
Part 3 – Preview your atlas maps

1. The Atlas is ready for preview.

- Click onto the Preview Atlas button along the top tool button bar as shown below:



- The first page of the Atlas should appear showing the title of the ward being displayed in the main map window
- Using the arrows on the Atlas toolbar click forward to preview the next page of the Atlas as shown below:



You may notice that the health area name changes, but the map window extent does not change to zoom into the next area. We will address that issue now.



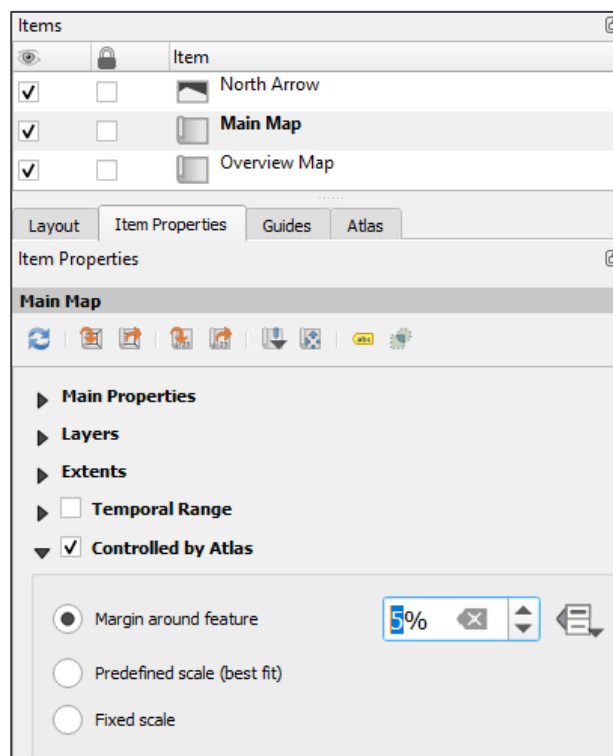
Part 4 – Controlling the map extent using Atlas

1. Configuring the Atlas to adjust the extent of each map to its ward boundary

- In the **Item** list click **Main Map** to activate its **Item Properties**

Remember: if the layer is locked you will need to untick it!

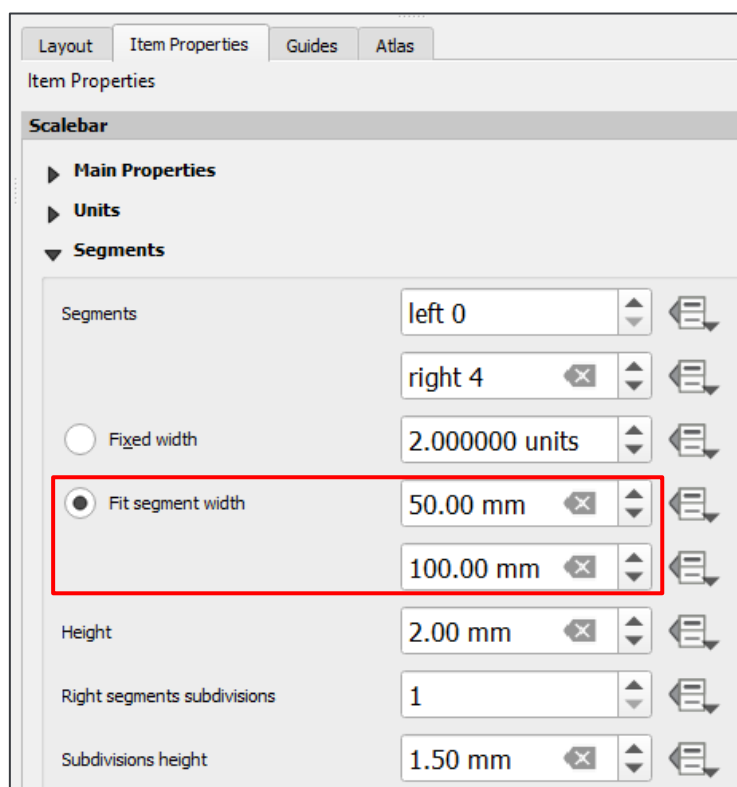
- Scroll down to **Controlled by Atlas** and tick the checkbox; this will allow the extent of the map to be driven by the Coverage layer
- If you are still in preview mode, then the extent of the map should change immediately to zoom in on the current health area – test it now
- You can play around with the margin around feature % to display the wards aesthetically



Part 5 – Dynamic scale bar

You may notice that the size of the scale bar does not adjust to each new extent; we will configure this now.

1. Ensuring that the scale bar adjusts to varying scales of each map
 - In the **Item** list click **<Scalebar>** to activate its **Item Properties**.
 - In the **Item Properties** for the scale bar, scroll down to **Segments**. By default, the scale bar is set to **Fixed Width**.
 - Change to **Fit segment width**

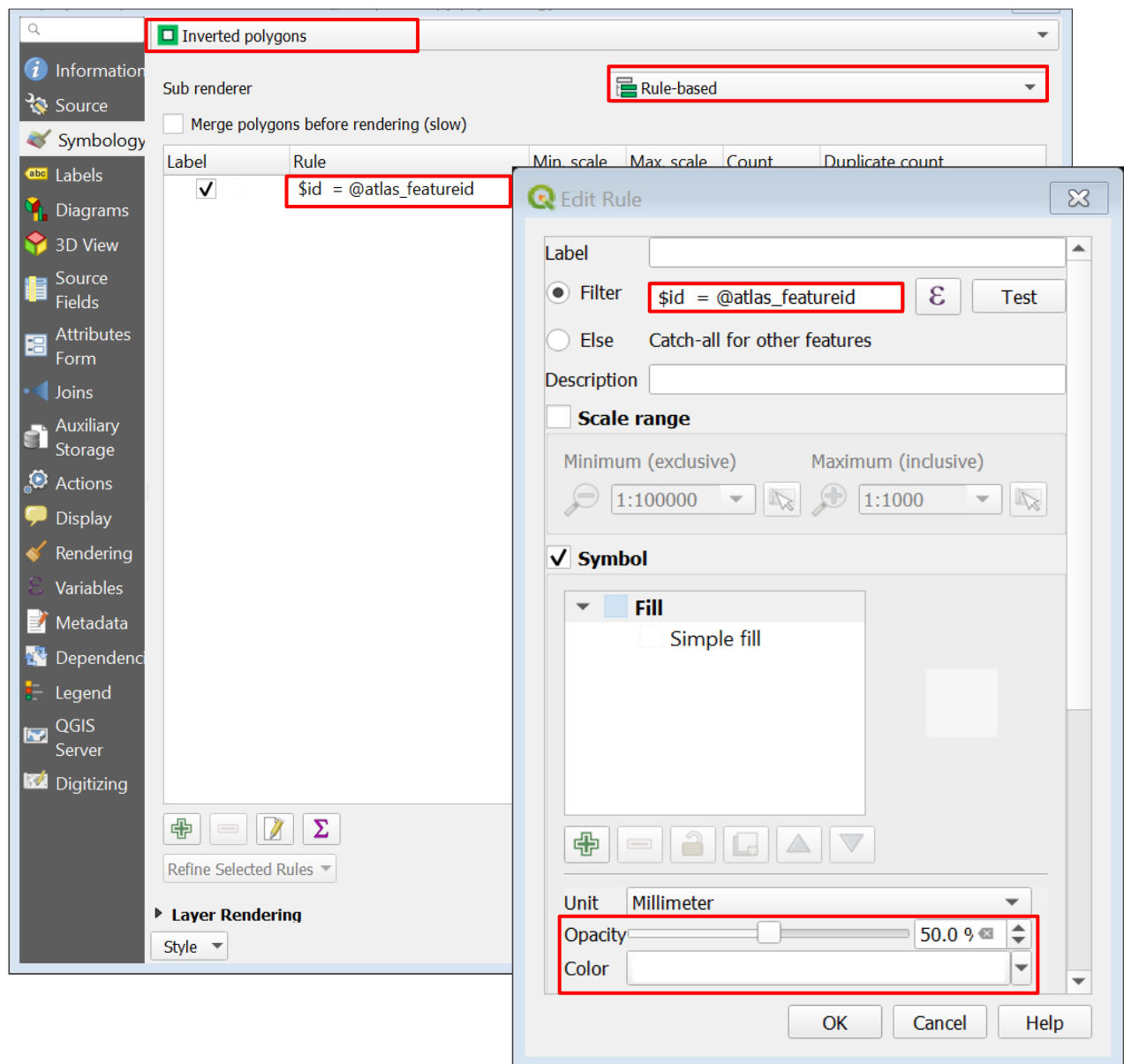


- Set **minimum** and **maximum** to 50mm and 100mm respectively
- Test your Atlas to ensure that the scale bar adjusts correctly for each map and stays within the bounds on the page

Part 6 – Dynamic symbology to highlight map extent

There are additional techniques which will make your maps much clearer and potentially reduce spatial errors from survey teams in the field. At present the boundary of one ward to another is defined by the thick orange, black and white boundary lines. To make the target ward for each map clearer, we will create a semi-transparent white mask to cover the surrounding areas.

1. Adapt the symbology of the *Kaura_ward_boundaries* layer
 - Return to the **main project window**
 - First, duplicate the layer *Kaura_ward_boundaries* (right-click the layer and select **Duplicate Layer**)
 - Right click on the duplicate layer and rename it to *Kaura_ward_mask*
 - Tick on the new layer to make it visible
 - Now open up layer **Properties** for the new duplicate mask layer
 - First, disable **Labels**, then select **Symbology** tab

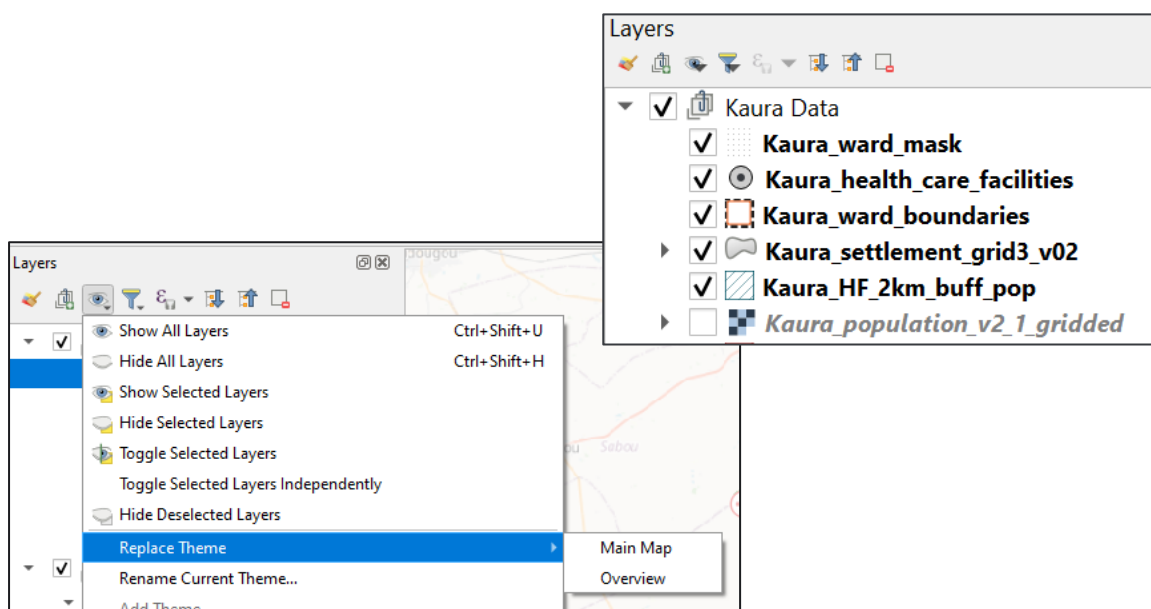


- Select Inverted Polygons renderer
- For Sub-renderer select Rule-based
- In the table that appears, double-click where it says (no filter)
- In the **Edit Rule** window that appears, use your knowledge of building expressions to set the following: `$id = @atlas_featureid` (Hint: In the **Expression String Builder** window: Look in **Record and Attributes & Variables**)
- Click **OK** to confirm the expression
- Back in the **Edit Rule** window, scroll down, and set up a style of your choice for the mask effect; here we have used a white fill with 50% opacity. You may want to add a fill effect (such as dots or hashed lines) to your mask layer
- When you are happy with the mask style click **OK**, then click **OK** again on the **Layer Properties** window

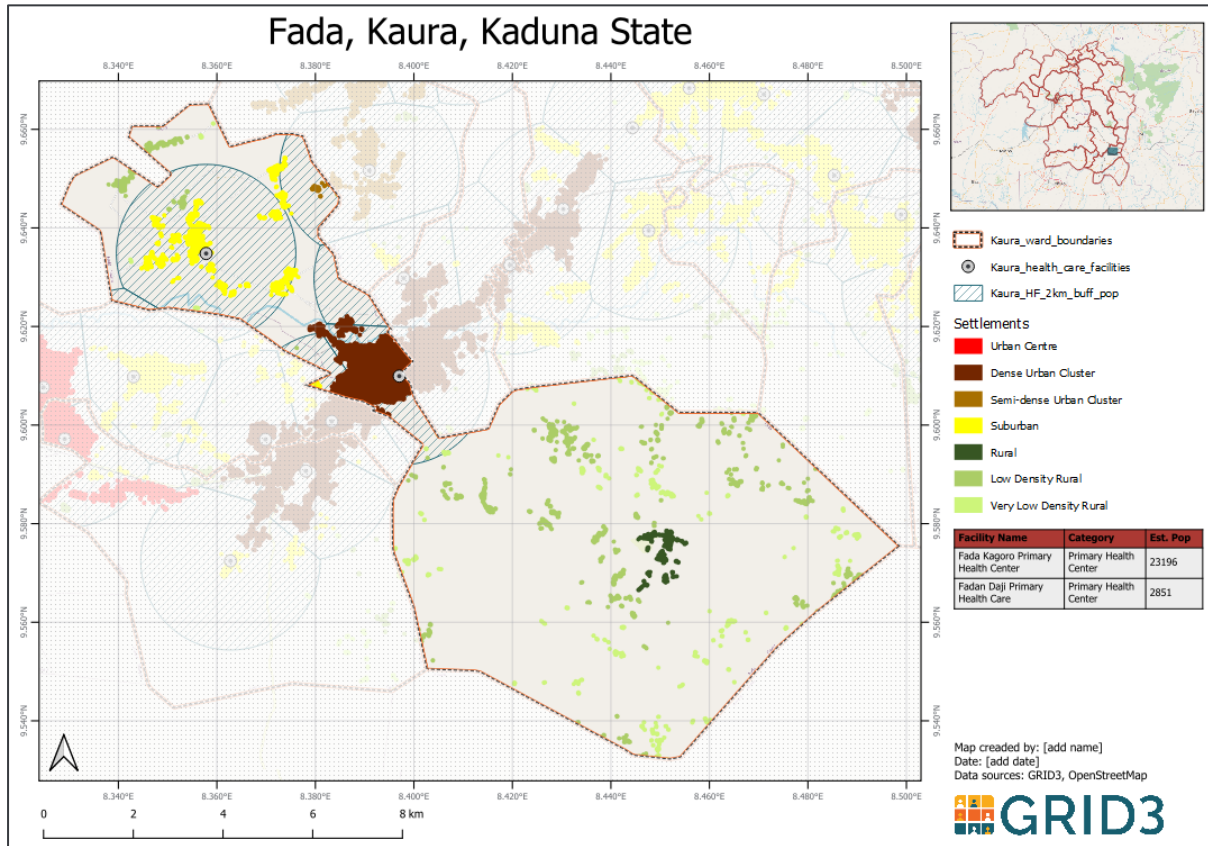


Here we have established a semi-transparent fill which will be applied to all regions outside of the current Atlas feature, i.e. in programming-speak, a style rule that is applied to the 'inverse' of features that meet criteria $!id = @atlas_featureid$. Whilst you might see the new effect in the main project window, if you return to your map layout and preview it, you will notice that nothing has changed. This is because the map layout uses the pre-set **Theme** called *Main Map* to decide which layers to display, so you will now need to update this theme.

- In the main project window, in the **Layers** panel, click **Manage Map Themes**, as indicated
- Then **Replace Theme > Main Map** (you will be prompted to confirm, click **Yes**;
- **Main Map** theme should display the layers indicated below (right)
- Move your map layers around in the layers panel to place the new mask layer on top of your original health areas layer and your health facilities layer



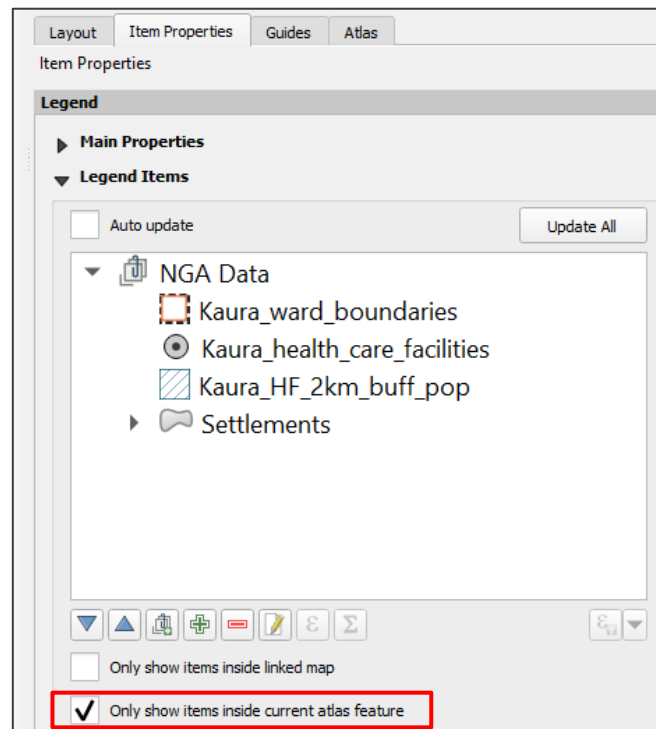
- Finally, return to your map layout and test (preview) your Atlas to ensure the effect is as you anticipate. You may have to refresh the map theme connection in the item properties of your main map (under layers).



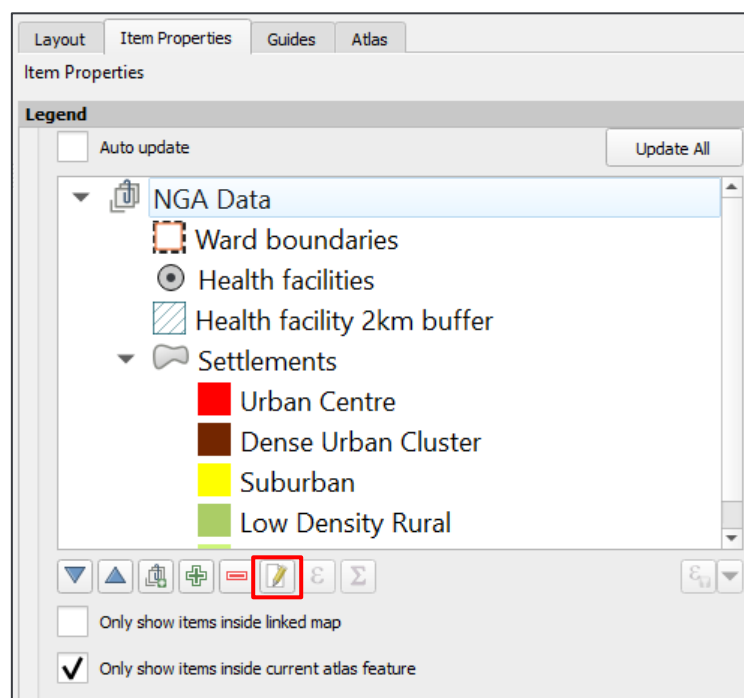
Part 7 – Additional dynamic map elements

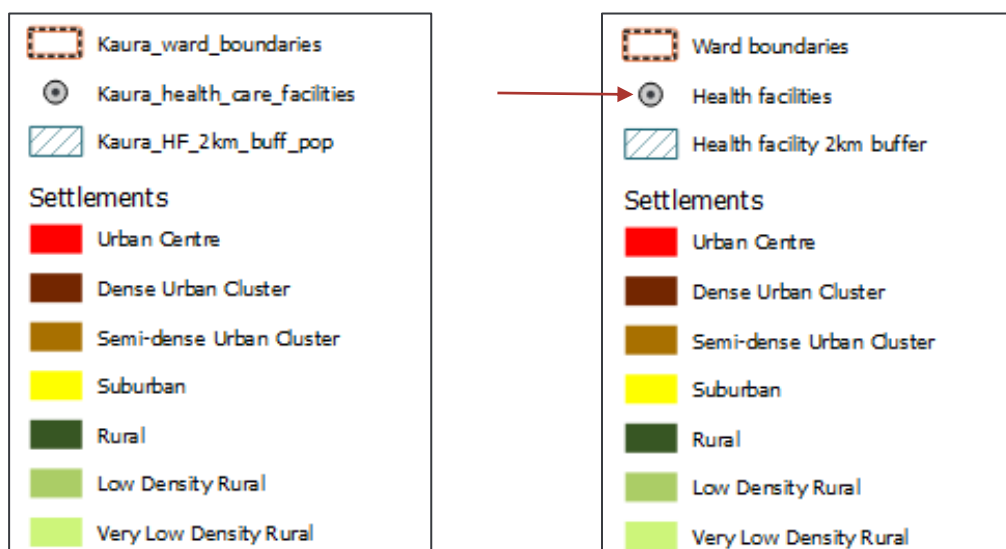
The extent of your main map window, map page title and scalebar are already being driven automatically by the atlas. We can also add this dynamic factor to other map elements, such as the legend and estimated population counts attribute table.

1. Create a dynamic legend
 - Navigate to the legend item properties
 - In the Legend Items drop-down, tick the *Only show items inside current atlas feature* box
 - If you are still in preview mode, you should see the legend adapt to only show the settlement classes present in the target ward (e.g Fada has no Urban Centre (red) settlements, so Urban Centre is not shown on the legend)



- While you are amending your legend properties, it is good practice to change the legend item names to remove messy characters and long file names

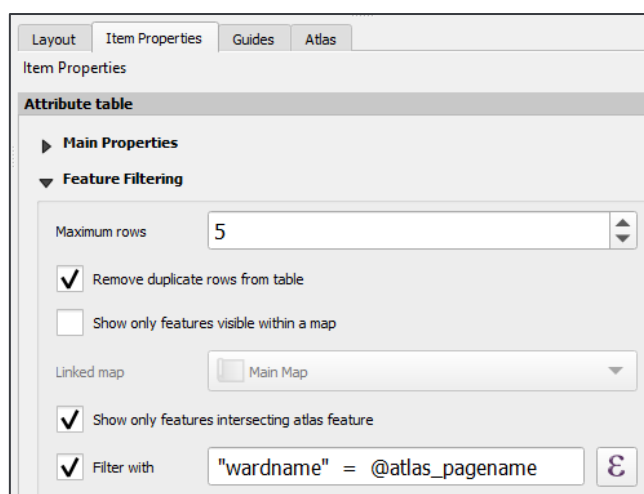




2. Create dynamic attribute tables

The table shown on the atlas template displays the estimated population for each 2km health facility buffer area. It is linked to the attributes table of the *Kaura_HF_2km_buff_pop* layer. These estimates were calculated from the *Kaura_population_v2_1_gridded* layer using Zonal Statistics. The table has been modified to only display 3 of the total 23 fields available to display from the buffer layer.

- Navigate to the item properties for *<Attribute table frame>*
- Tick *Show only features visible within a map*
- To ensure only the records for health facilities within the target ward are displayed, enter the filter expression shown below. If we do not filter, records for partial buffers that intersect the target ward, but belong to health facilities within other wards will be displayed on the table



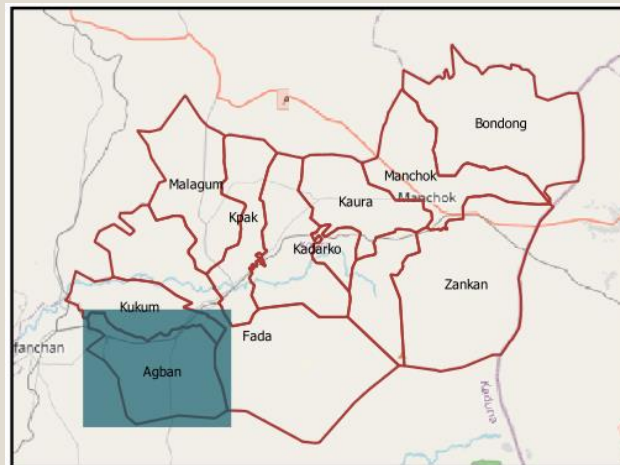


- You can investigate different ways to display this table data using the *Attributes* button under main properties and the *Appearance* drop-down

Challenge 1: Create an appropriate overview map

You will notice that the overview map still presents a large area of Nigeria. It still reflects the requirements of the 'LGA map' showing all LGA's within Kaduna state. This is not particularly useful for the ward maps. The maps you are now producing require an overview map that indicates the position of each ward within Kaura - see the example below.

HINT: You will need to change the zoom of the overview map and alter the overview map theme as we did before with the main map theme.







Challenge 2: Create an advanced overview map and feature-based extent indicator

Below is an alternative overview map and extent indicator. It employs an Atlas-driven symbology technique. To replicate this effect, you will need to return to the main project window and change the symbology of the ward boundaries layer as shown below:

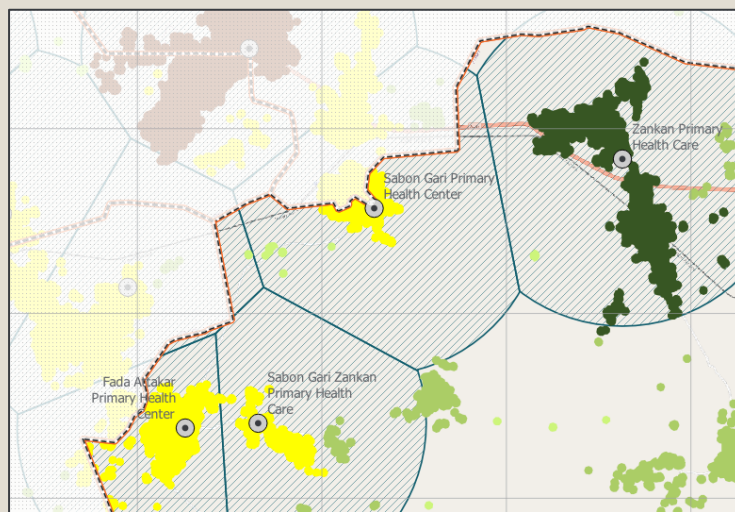


Rule-based	
Label	Rule
<input checked="" type="checkbox"/> 	atlas feature Sid = @atlas_featureid
<input checked="" type="checkbox"/> 	everything... ELSE



Challenge 3: Create rule-based labels to display health facility names

Displaying health facility name labels is a good way to help surveyors in the field who are interpreting the microplan maps. However, if we switch labels on for all health facilities, the maps can look quite messy. By using rule-based labelling, we can display the labels for the health facilities in the target ward, and not those under the mask.

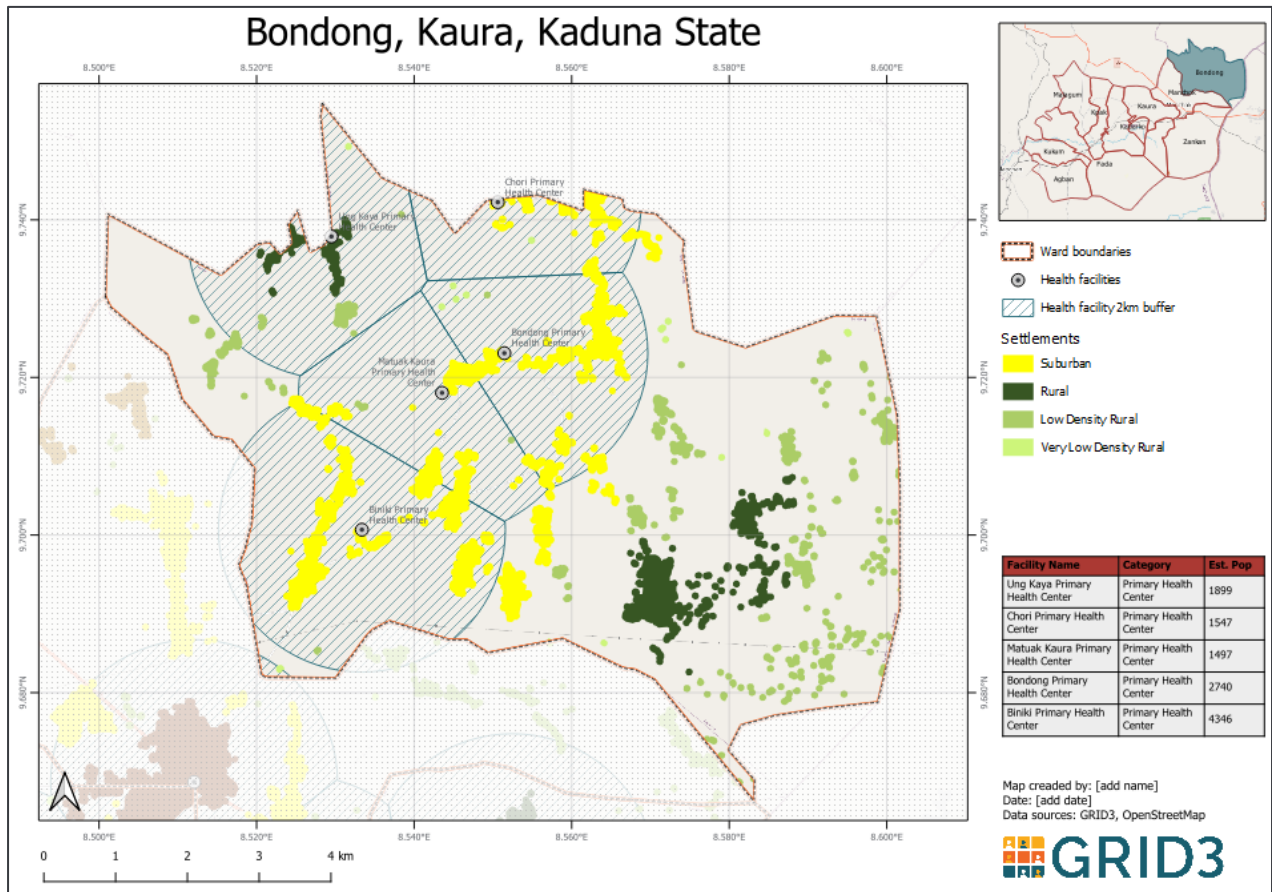


Rule-based Labeling	
Label	Rule
<input checked="" type="checkbox"/> Aa	"wardname" = @atlas_pagename



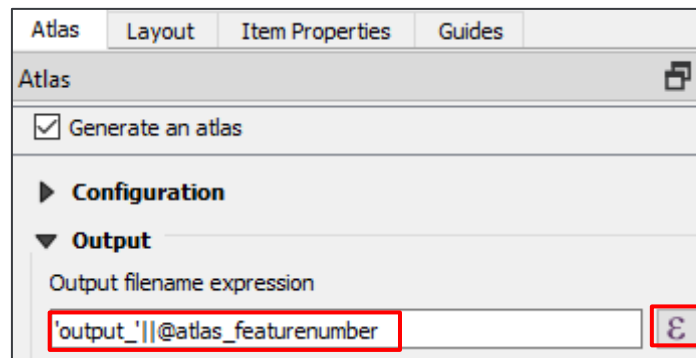
Part 8 – Publishing your Map Atlas

Your final maps should look similar to the example below:




You are now ready to export your atlas maps to your chosen format. QGIS allows you to export to many image formats, but we will generate individual PDFs for distribution.

Refer back to the settings in the **Atlas** tab, in the **Print Layout Window**. There is a section at the bottom called **Output**. The **Output** section allows you to define individual filenames for each of your output maps. As default this is 'output_' || @atlas_featurenumber. If this remained unchanged and we exported all our ward maps, the output names would be: *output_1.pdf*, *output_2.pdf*, etc... These names are not very useful as they do not describe the maps. A better output name would be *Agban_Ward_Map.pdf*, *Kpak_Ward_Map.pdf*, etc... To achieve this, we need to edit the expression.



1. Building an expression to auto-generate filenames for output maps

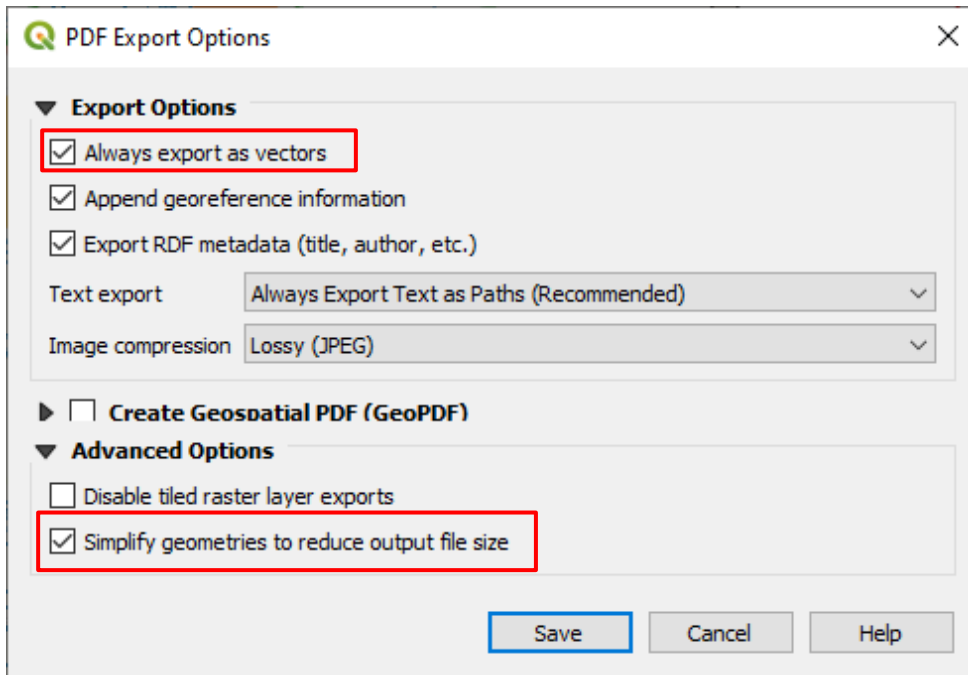
- Click the **Expression based filename** button  highlighted above (if this is unavailable/greyed out, untick **Single file export when possible**)
- Using the method outlined earlier in Part 2 to adapt the expression to the following:
@atlas_pagename || '_Ward_Map'

The components of this expression are defined below:

@atlas_pagename	Page name of the current atlas iteration
	- 'concatenate' – this enables a text string to be joined with a variable
'_Ward_Map'	a text string which will appear in every filename (use single quotes)

2. Export your Atlas maps to files

- In the main Print Layout window, click on **Atlas** drop-down menu
- Select **Export Atlas as PDF...**
- In the file browser that appears select folder: **C:\G3_GIS_Training\Exercises\Data\Output_data**
- In the PDF Export Options window select the settings as displayed below



- Click **Save**
- Once exported (you will be notified when the Task is complete), navigate in **Windows Explorer** to the *Output_data* folder, then open and inspect your maps!

