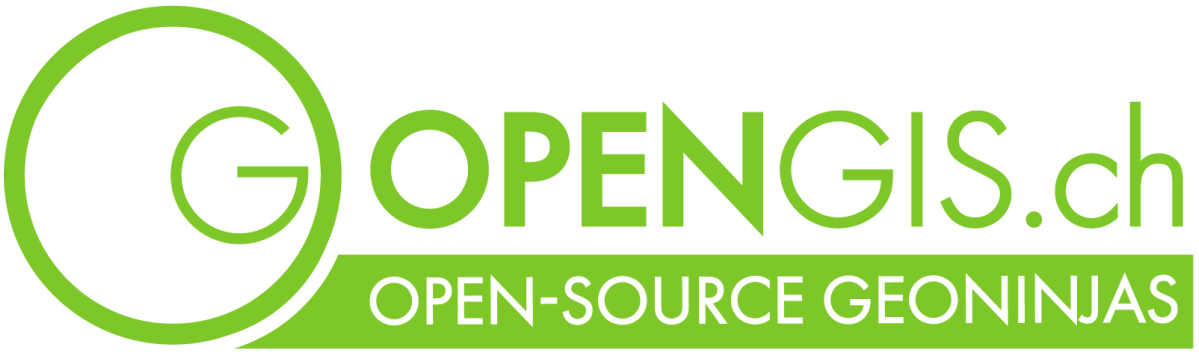

QGIS Basic Course



Course Goal

At the end of this course we know the main functionalities of QGIS Desktop. We are able to import data from various data sources and to symbolize and label them. We can capture new features and edit their attributes and geometries. We can create a map with a professional print layout.

Required Files

Data:

- Beehive symbol: beehive.svg
- (Data on bee species: bee_species.csv)
- Data on bee diseases: bee_disease.csv
- Matteo's file: data_01_botanical.gpkg

Introducing Maya Mielina

To show you the QGIS functions, we will tell you a story. The story of Maya and her bees. The characters and events are entirely fictional. To make the story more interesting, we may have gotten carried away here and there. We hope you will forgive us any inaccuracies and liberties taken, especially if there are any beekeepers among the participants.

This is Maya Mielina.

During her professional career, Maya worked as a GIS specialist in Zurich. Now she is retired and moves back to her home village of Lavertezzo. As a child she already helped her grandmother with beekeeping, and now, as a hobby beekeeper, she is finally making her long-held dream come true. For planning and documenting her beekeeping, she would like to use her favourite software, QGIS. With QGIS she also wants to create a map showing her beehive locations, which she can show to her friends or even print and hang on the wall.

Maya has freshly downloaded QGIS on her new laptop.

QGIS Project File

Full of enthusiasm, Maya starts her project
We open QGIS to create a project for Maya's beekeeping.

> The free and open-source software QGIS allows you to visualize, analyze, edit, or transform geodata. The QGIS project (a file with the format `.qgs` or `.qgz`) is at the centre of this. Note: no geodata is stored in the project itself; the data is only linked to the project.

> However, what *is* stored in the project file is, for example, the cartographic representation, the data **sources** (as a link), map layouts, and the formatting of input forms.

User Interface

Maya has freshly downloaded QGIS on her new laptop. She adapts the environment to her needs.

There are different components in the user interface. We look at the most important ones.

Menu

(List is not exhaustive)

- Project
  - Open / Save etc. the file
  - Project properties
  - Print layouts
- Edit
  - Functions for editing features and their geometries
    (as long as no layer is loaded, most functions are deactivated)
- View
  - Control the map view
  - Control the layer view
  - Show and hide panels and toolbars
- Layer
  - Data source manager
  - Create and edit layers
- Settings
  - Profile management
  - Style and interface management
  - Options
    (properties of QGIS, across projects, such as language)
- Plugins
  - Manage plugins
- Python Console
  (Console to control QGIS with PyQGIS commands)
- Vector
  Advanced vector-specific functions
- Raster
  Advanced raster-specific functions
- Database
  Database manager *(to access databases and manage data, like a small internal PGAdmin)*
- Web
  Metasearch to browse the metadata of web services

In general: in the Vector, Raster, Database, and Web sections, installed plugins can also be invoked.

- Mesh
  Mesh calculator / Raster calculator to export converted raster layers
- Processing
  - Show the toolbox
  - Graphical Modeller *(to graphically model process sequences.)*
- Help
  - Links to online documentation
  - Information about the application

Toolbars

- Project
  Open, save, etc. the project as well as print layouts
- Navigation
  To navigate and zoom on the map
- Spatial bookmarks
- Attributes
  Selection of existing features, link to the attribute table, and tools such as field calculator, measurement, statistical summary, and info boxes
- Data source manager
  Loading data from various data sources
- Digitizing
  Creating, editing, and deleting geometries
- Labeling
  Functions for the manual placement of labels

Panels

- Browser
  for managing, editing, and loading data sources
- Layers
  List of the layers in the project

Map View

- the main window, the map view

Status Bar with Interactions

- Search
  can also be opened with `Ctrl+K`. This allows quick access to functions. The Swiss Locator extension will also be integrated into it.
- Empty area for status
  progress of processes, etc.
- Coordinates or extent
  specific coordinates can also be entered here
- Option for mouse pointer
  whether to display the coordinate of the mouse pointer
- Magnification
  zoom level
- Lock scales
- Rotation
  rotation of the map
- Render
  update the map on zoom or movement
- Projection
  the main projection of the project
- Error messages display
  speech bubble with 3 dots, error message only when there are any

> **Tooltips** QGIS is full of hover help. Let's look at a few.

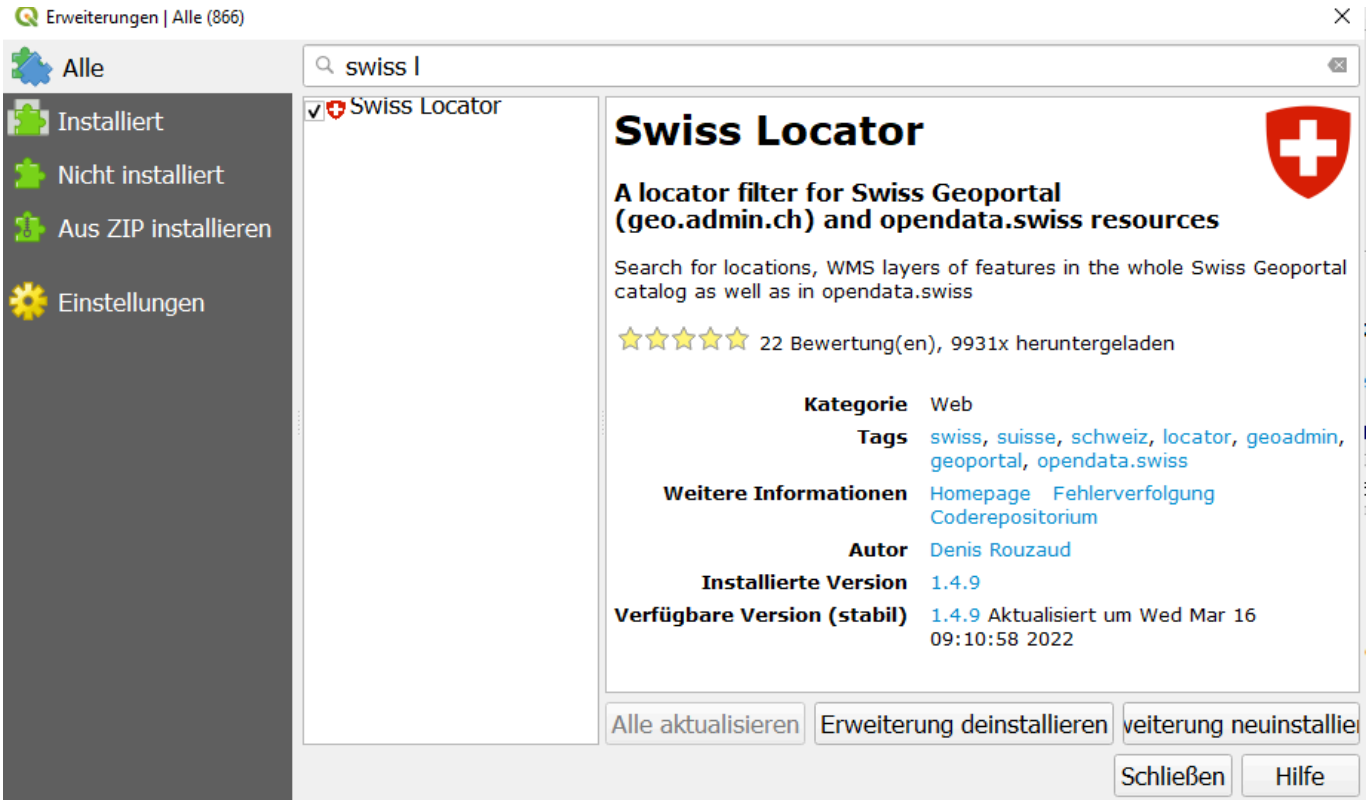
Extensions (Plugins)

> Plugins allow you, in addition to the tools and algorithms available by default in QGIS, to integrate many more functionalities in a very quick and easy way.

Using the plugin manager (**Plugins > Manage and Install Plugins**) we can find, download, and install all plugins.

Further information about the plugins: https://plugins.qgis.org/plugins

We load plugins for QGIS:



Swiss Locator
Search for locations, WMS and WMTS layers, or features in the entire Swiss Geoportal catalogue

possibly: **QuickMapServices**
Collection of easy-to-integrate background rasters

possibly: **SwissGeoDownloader**
Collection of easy-to-integrate background rasters

Coordinate Reference System (CRS)

In the **Project Properties** (alternatively bottom right) we can define the projection (CRS). For the course exercises we choose the official Swiss coordinate system: CH1903+ / LV95 (EPSG:2056).

Loading Background Data

"Maya needs maps to get started with the project."

We drag OpenStreetMap into the empty map view (**Browser panel → XYZ Tiles > OpenStreetMap** click and drag into the map view).

"Maya loves the Swiss maps; she also wants to load them in her project."

Swisstopo provides countless map products as web services. They can be loaded into the project, among other things, with the **Swiss Locator** plugin. We type `Karte swissTLM (farbig)` into the **search field in the status bar**. We set the projection of the project to `EPSG:2056` at the bottom right. We add more maps or orthophotos as desired.

> **Show and hide panels and GUI**
> `Ctrl-Tab`: show/hide panels and docks
> `Ctrl-Shift-Tab`: show/hide the entire GUI
> `F11 + V`: full-screen mode

> **Navigation, zooming in the map**
> Shift-Drag: draw a rectangle to zoom
> Move the map: "Space bar" + left mouse button

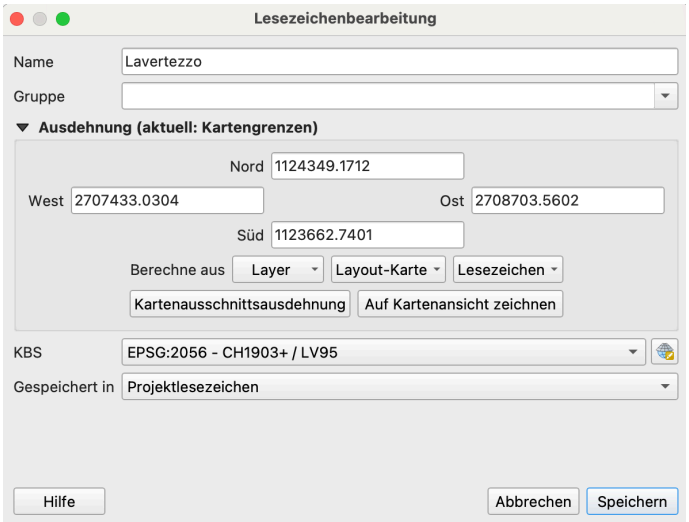
Localization

In the status bar we have the search. We search for the `town of Lavertezzo` and find it in the Verzasca Valley.

And there is Maya's grandparents' house.

Spatial Bookmarks

(Optional) We add a spatial bookmark to save the location. **New spatial bookmark** 



Exploring the Environmental Conditions

Before she starts producing honey, Maya checks whether the conditions are right. What about traffic volumes? Are there neophytes that could interfere with her plans?

"For instance, is there the garden lupine (Lupinus polyphyllus)? The bees love it. But because it is an invasive species, its spread would be accelerated by the increased pollination by bees." Maya checks the conditions with WMS/WMTS layers from the Swiss Geoportal.

We load this map data via Swiss Locator in the search bar with the prefix `chw` for the WMS layers of the Swiss Geoportal: `chw lupine`

We go to the layer properties and make a few settings (**Right-click on the layer in the Layers panel > Layer Properties > Symbology**). In the Layer Rendering section we choose *Blending mode Multiply* (button **Apply** or **OK**).

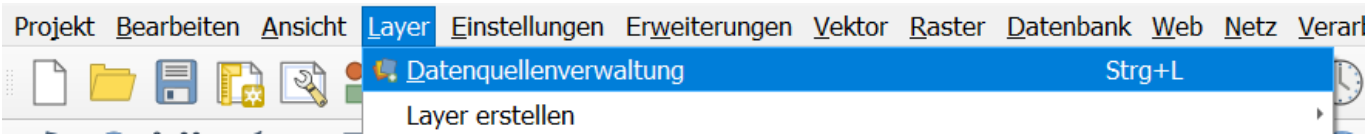
We check freight traffic `chw gueterverkehr`.

The conditions turn out to be good; Maya is now sure she wants to start beekeeping.

Optional: Load interesting WMS/WMTS layers (via Data Source Manager)

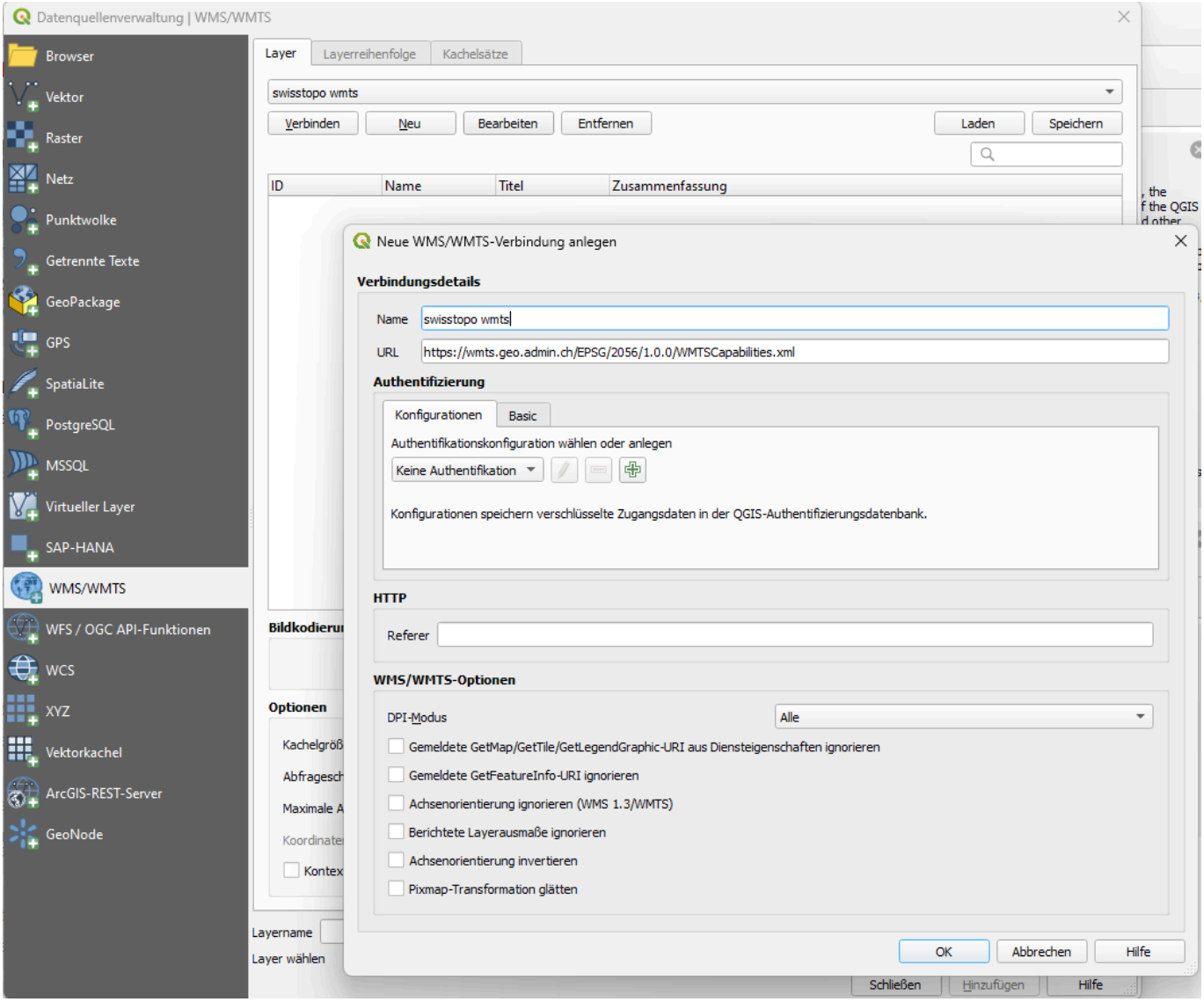
> With the **Data Source Manager** we can connect a wide variety of sources to our QGIS project and load data from them into the project.

Let's load WMS and WMTS layers from Swisstopo:



URL WMS (German): https://wms.geo.admin.ch/?VERSION=1.3.0&lang=de
e.g. `ch.are.landschaftstypen`

URL WMTS: https://wmts.geo.admin.ch/EPSG/2056/1.0.0/WMTSCapabilities.xml
e.g. Relief or Swissimage orthophoto



> Search for Swiss data, e.g.:
>
> - geodienste.ch (cadastral data and other standard works, among others)
> - opendata.swiss (freely accessible data from Swiss authorities)
> - geocat.ch (geodata catalogue)

Loading data from an external PostGIS database

Maya remembers that she and a few friends had entered addresses into an external PostGIS database during a PostGIS training course. There she wants to look up the address of Matteo, a former fellow student. She also loads this data into her project.

Layer > Data Source Manager > PostgreSQL > New

Name: (freely chosen, e.g.) `demo db`
Service:
Host `demopg.opengis.ch`
Port: `21699`
Database: `coursedemo`
SSL mode `require`

> Test connection

Authentication, e.g. **Basic**
Username: `course_participant`
Password: `qgis!`

> Connect

We choose the table `public.mayas_friends` and add it to the project.

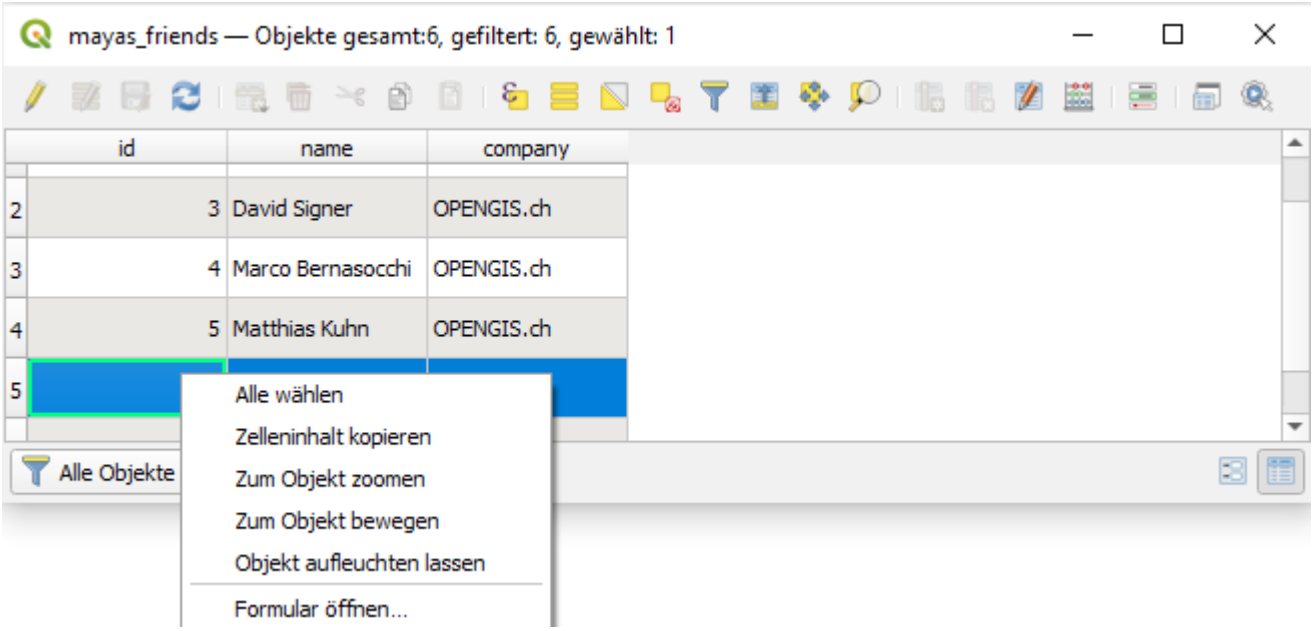
Thanks to this layer we now find out where Matteo lives and also which house Maya lives in.

Locating entries from the attribute table

> The **attribute table** is another important component of QGIS. It allows you to display or edit all non-geometric data belonging to a layer. The display of the attribute table is linked to the map view. For example, selected features are visually highlighted (in yellow) in both places.

Right-click on **Layer > Open Attribute Table**

We search for Matteo's entry. We select Matteo and find his house on the map: **Right-click on row entry > Zoom to Feature**.



Maya pays Matteo a surprise visit in Bellinzona.

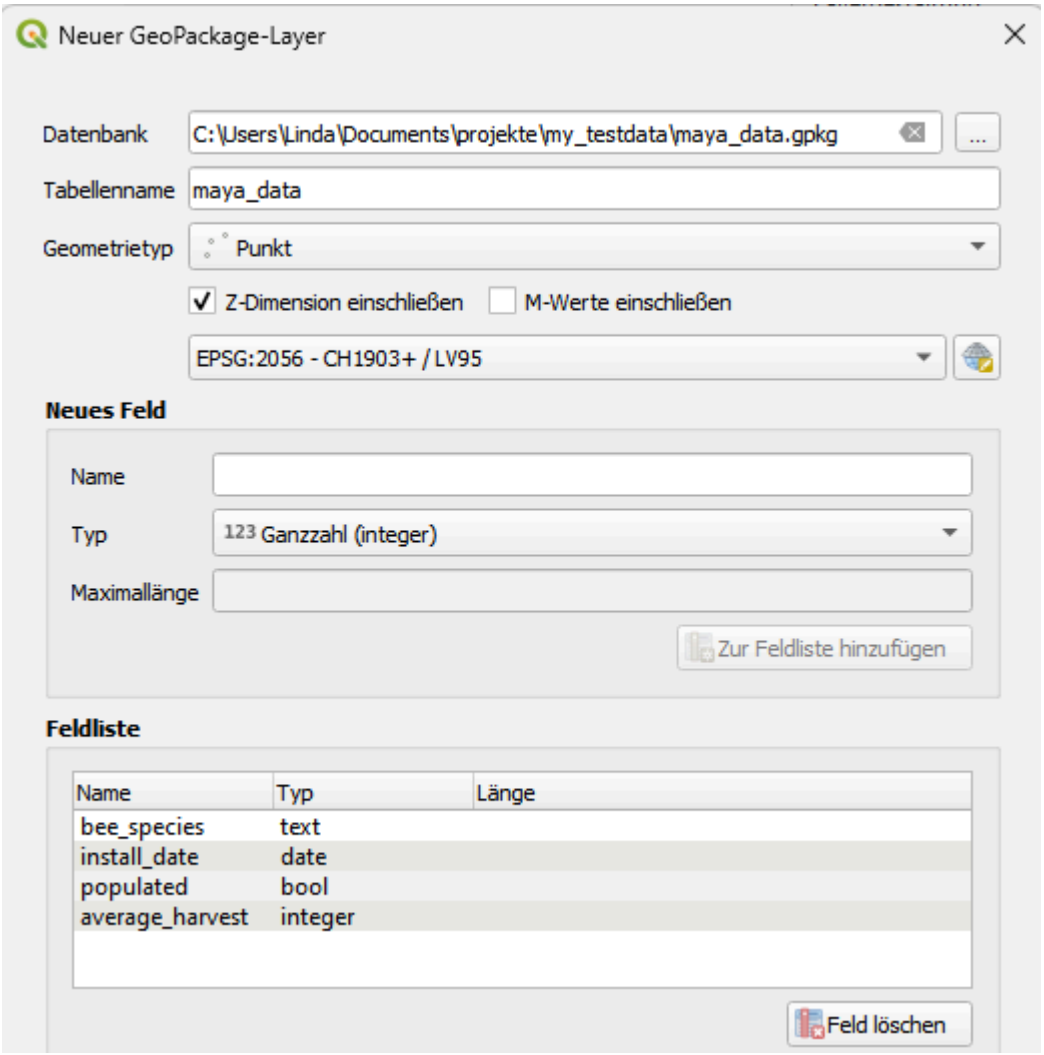
Create a new GeoPackage and layer for capturing the beehives

Maya has already finished repairing her grandmother's first beehives and painted them brightly, and she is excited to set up the boxes. Full of enthusiasm, she also prepares the layer for the beehives in QGIS. Since people from German-speaking Switzerland do not speak Italian and people from Ticino do not speak German, but Maya wants to present the project in both places, she chooses English for the technical parts.

We add a table/layer to map Maya's beehives. To do this we create a GeoPackage (`mayas_bees.gpkg`) and add a new table (`hive`).

Layer > Create Layer > New GeoPackage Layer

Geometry: `Point`
Projection: `EPSG:2056`



Now we add important attributes:

- `bee_species` (text)
- `install_date` (date)
- `populated` (boolean: true/false)
- `average_harvest` (int)

> - Texts in a database must not contain special characters (ö, ä, %, ..)!
> - avoid spaces (can be replaced with _ for example)
> - Use alias names if needed

The newly created layer appears in the Layers panel.

Organizing the project

Maya is a tidy person. She also keeps her QGIS project tidy.

We arrange the layers into groups and rename them in the Layers panel.

Capturing features (digitizing)

> It is important to note that changes to the data are not stored in the QGIS project itself, but directly in the data source linked to the project.

Maya installs her first 3 beehives right in front of her house, so she is ready when she receives her first bees.

Digitizing the first feature:

1. Select the layer to be edited `Beehives` in the Layers panel (it gets highlighted in blue)
2. Switch to editing mode (activate the pen button )
3. Activate the Add Point Feature button 
4. Place the point on the map (left mouse button)
5. Fill in attributes: installation date.
6. Confirm with "OK".

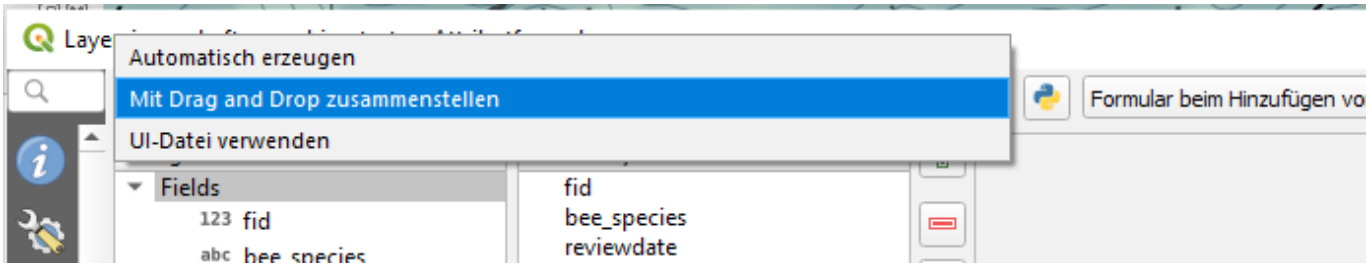
Configuring the input form (attribute form) for the beehives

Maya wants to prevent typos and other potential data errors, so she improves the input form.

> The attribute form displays the data of a single feature in a layer. In the layer's properties, the attribute form should be configured so that data entry and viewing can be structured. Among other things, constraints and value ranges can be defined, as well as drop-down lists, date fields, checkboxes, or sliders. The attribute form is linked to the map view. Selected features are visually highlighted (in yellow) in both places, among other things.

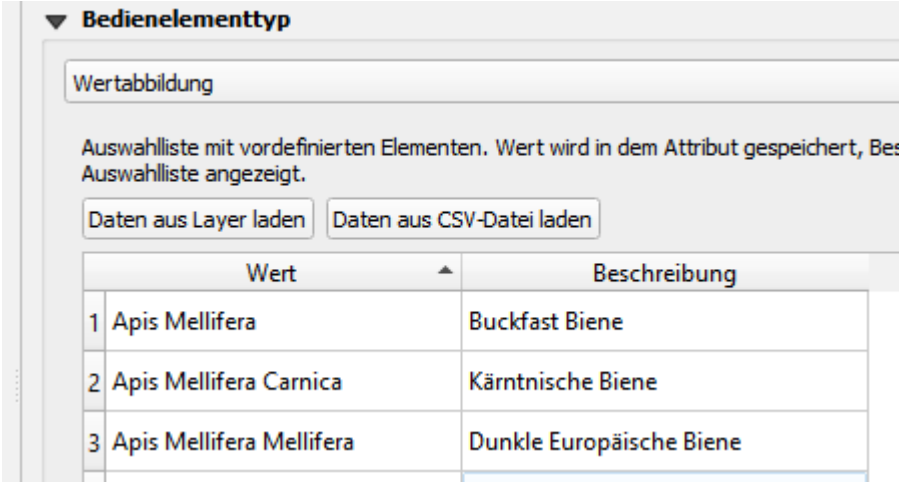
The attribute widget types are automatically detected from the data type. The form can also be designed according to your own needs.

Right-click on the layer > Properties > Attributes Form > Drag and drop designer



Configuration:

- Remove the `fid` field from the form
- Configure a **range** for the average honey harvest (0 - 100 kg)
- Configure a **value map** with the values:



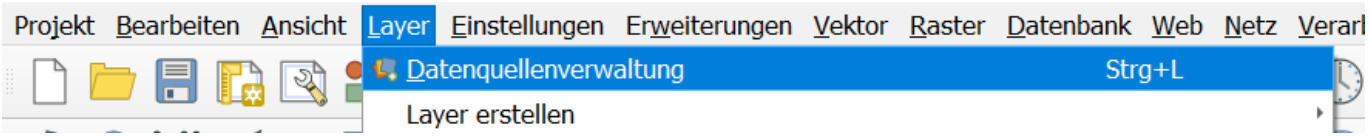
Editing existing features

Maya finally receives bee swarms, which are allowed to move into her already placed boxes.

We adapt the attributes. One way to subsequently modify already captured features is via the **Identify Features** tool. From the Identify Results panel the form can be opened to modify entries.

Loading data from Matteo's GeoPackage into the project

Matteo, whom Maya recently visited, works as a GIS specialist in Bellinzona. He wants to support Maya in her beekeeping and has just sent her a GeoPackage with botanical data for this purpose. With it, Maya can plan the placement of the beehives so that they feed on different blossoms and produce various types of honey.



Layer > Data Source Manager > GeoPackage > New > New connection to `data_01_botanical` > Connect > select `area` > Add.

Data management

Let's check where the data originates. Hold the mouse pointer over a layer in the Layers panel for a few seconds: a mouseover with the file path appears. Alternatively: right-click on **Layer > Properties > Information** The `hive` layer comes from the file `mayas_bees.gpkg`, and the `area` layer from the dataset with the botanical data.

It would be more elegant to have everything in a single GeoPackage. Maya wants to extend the data later, but at the same time keep Matteo's original data in its current state. That's why she wants to work with a copy of the

data and load this copy into her project.

From the Layers panel we remove the `area` layer again.

To manage the GeoPackages, the Browser panel is perfectly suited.

In the Browser the GeoPackages we have established a connection to in the Data Source Manager are listed. We import / copy the `area` table into the GeoPackage `mayas_bees.gpkg` by dragging it into the target GeoPackage.

Then we drag `area` from `mayas_bees.gpkg` into the project.

Cartographic representation of the plant data

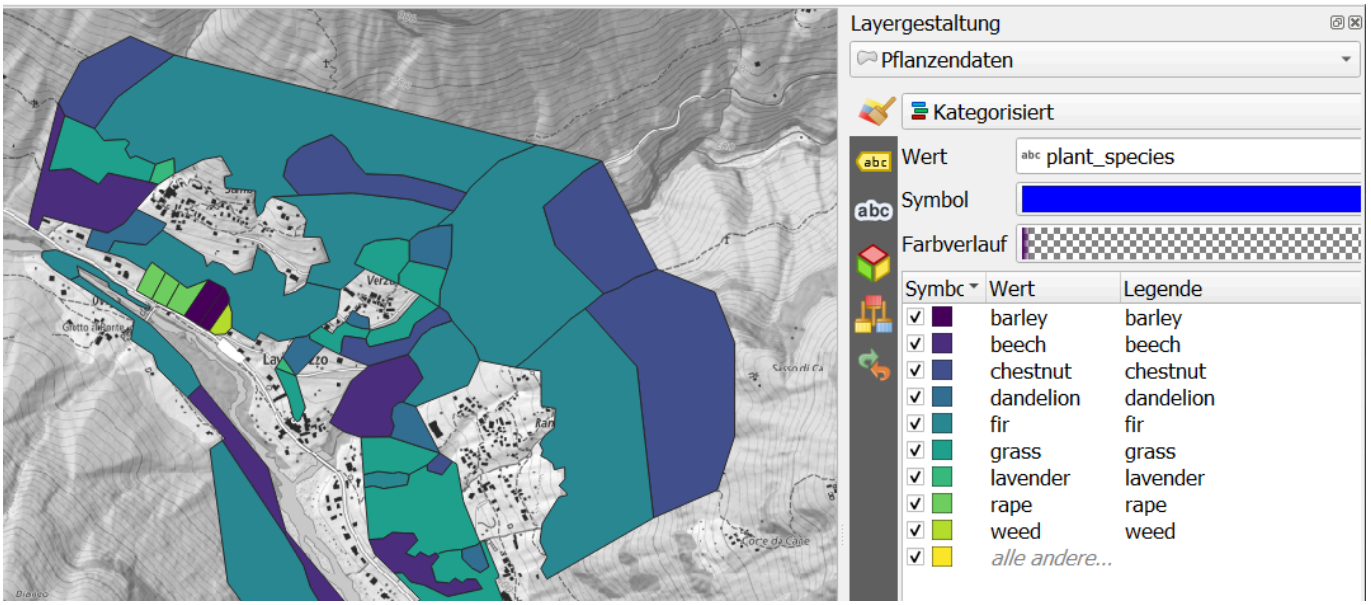
Maya wants to get a better overview of Matteo's plant data, first of all graphically. We adapt the symbology and the labeling.

> In QGIS, the cartographic representation, or "style", is defined individually for each layer, mainly via the **Symbology** and **Labels** properties.

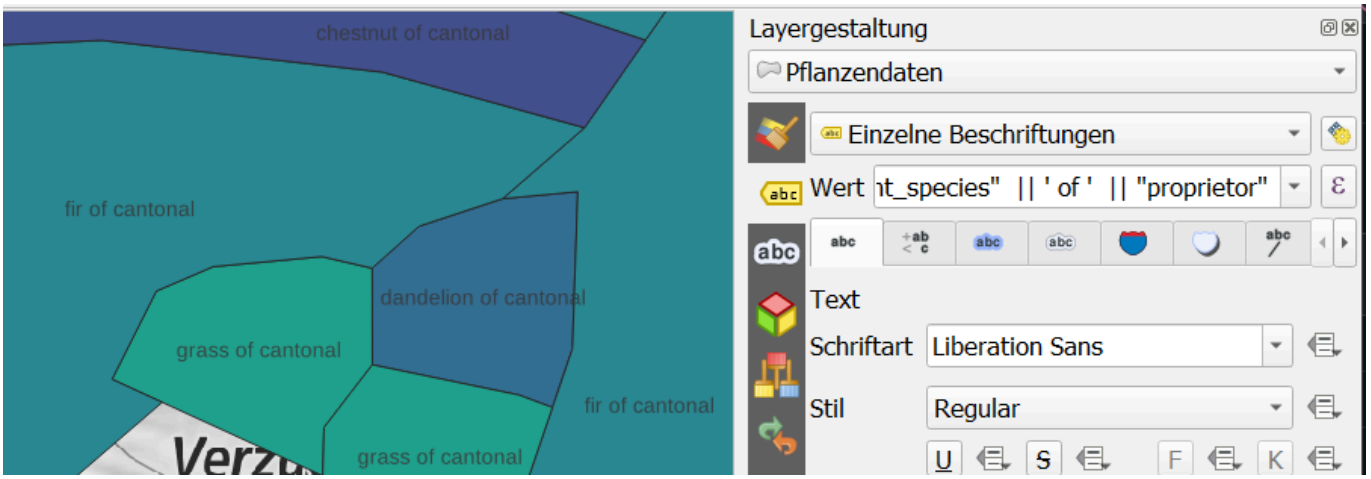
We work with the Layer Styling panel, since here (in contrast to the corresponding tab in the layer properties) changes take effect immediately. **View > Panels > Layer Styling**

There is the **Single Symbol** for the **Symbology** - here you can choose the fill.

We categorize by plant species **Categorized > Value:** `plant_species` **> Classify**

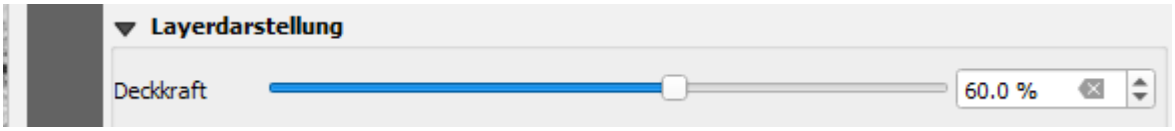


For the **Labels** we choose **Single Labels**. We can specify a field as a value or go a little deeper with an expression: `plant_species || ' (' || proprietor || ')'`.



Maya looks at the categorized areas on the map. She finds grass meadows (gras), dandelions (dandelion), firs (fir). And... chestnuts (chestnut)!

The polygons cover the background data; to remedy this we make the layer more transparent (it may be necessary to expand the small black triangle for this).



Repositioning beehives

Maya moves a beehive to a sunnier location. With the vertex tool  we pick up the beehive and place it at the desired new location (**Click on feature > move > place at the desired location with another click**).

Duplicating beehives

Maya is very motivated in her beekeeping. She receives more bee colonies and installs them in the area, like here at the chestnuts, because chestnut honey is her favourite honey. Where attributes are the same, Maya wants to copy the features and paste them at other places, so that she is as efficient as possible during digital capture.

With the Select tool  one or several of the beehive points are selected: (multiple: **click and drag a rectangle around the desired feature(s)**). The selected features now have a yellow symbol in the map view.

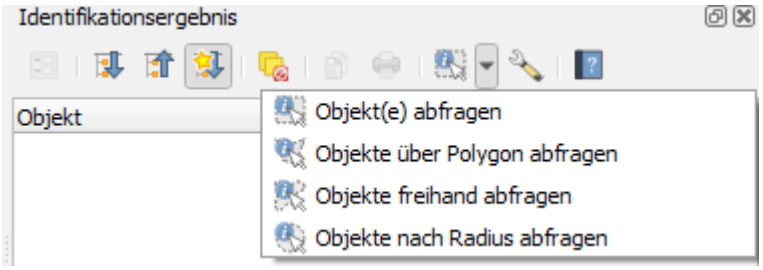
With **Deselect Features**  a selection can be undone.

With the shortcuts `Ctrl-C` and `Ctrl-V` the selected points are copied and, provided the same layer is still selected, pasted back into the same layer. (Alternative: via menu **Edit > Copy Features** and **Edit > Paste Features**).

The copied beehives now lie exactly on top of each other. With the vertex tool  we pick up the copied beehives and place them at the desired new location (**Click on feature > move > place at the desired**

location with another click). Now you should create a few beehives within / near the "plant data" polygons to generate some data.

Identifying features



With the Identify Features tool  we can identify one (**click on a feature**) or several features (**click > draw a rectangle around the features and release**). The Identify Results panel opens automatically (if not: it can be shown under **View > Panels > Identify Results**).

Maya tries out different options for the selection, e.g. different modes.

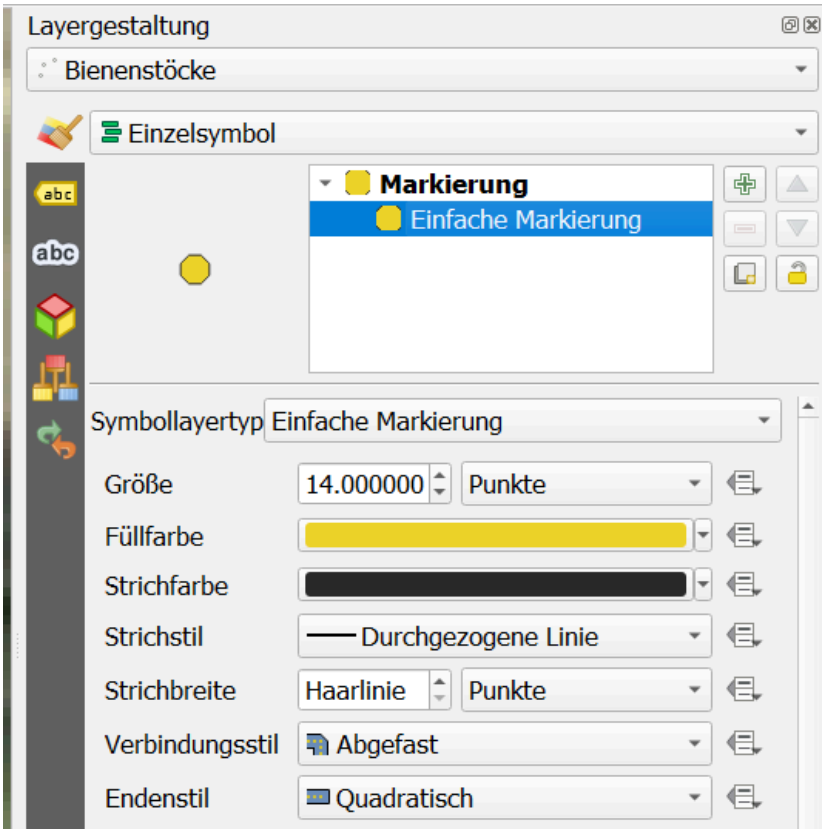
Cartographic representation of the beehives

> In QGIS there are countless configuration options for symbology and labeling. Different styles can be defined based on attribute values, properties of the geometry, a current selection, zoom levels, or any combination of these.

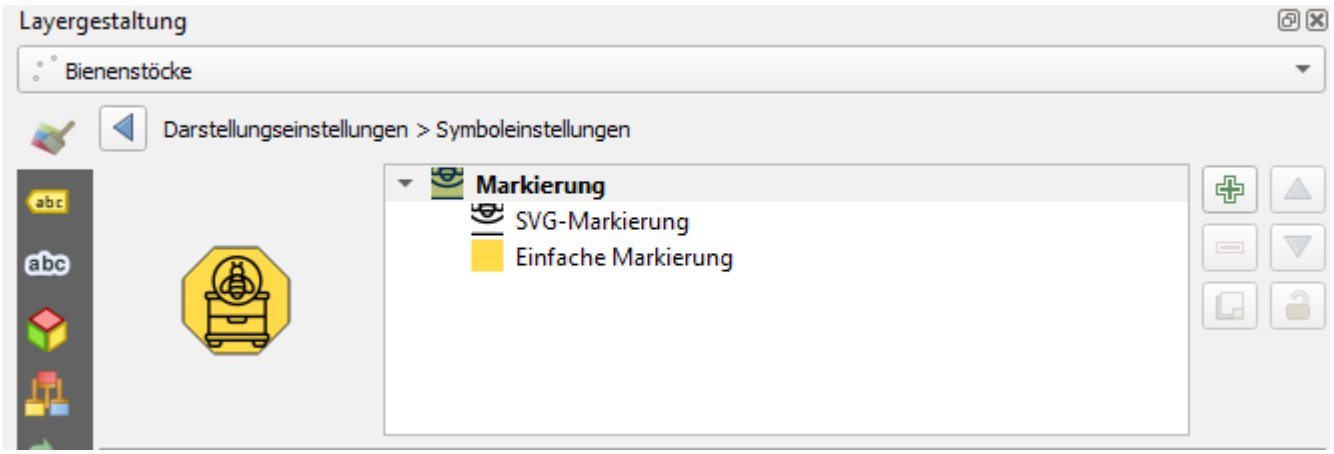
To get a better overview of her data, Maya creates suitable map symbols for the `Beehives` layer.

Panel > Layer Styling

We choose a **simple marker** and adapt the point symbol.

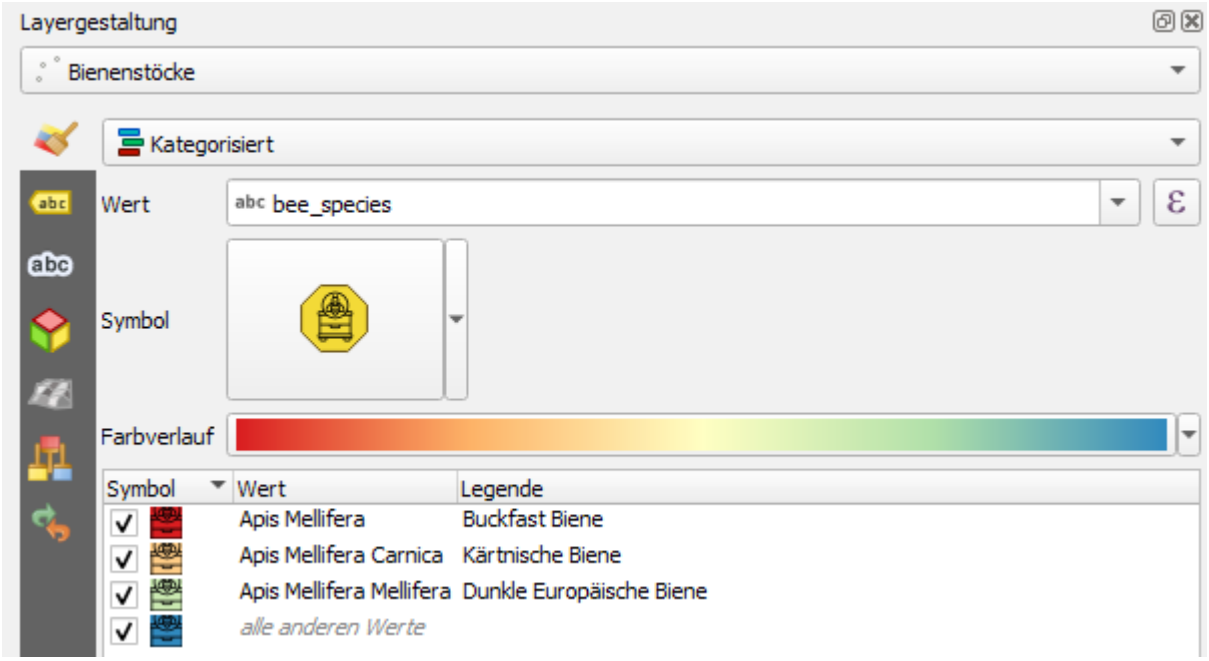


It is possible to compose a symbol from multiple layers. We add an **SVG marker** above the simple marker (green "+"). At the very bottom of the Properties window we load the file `beehive.svg`.

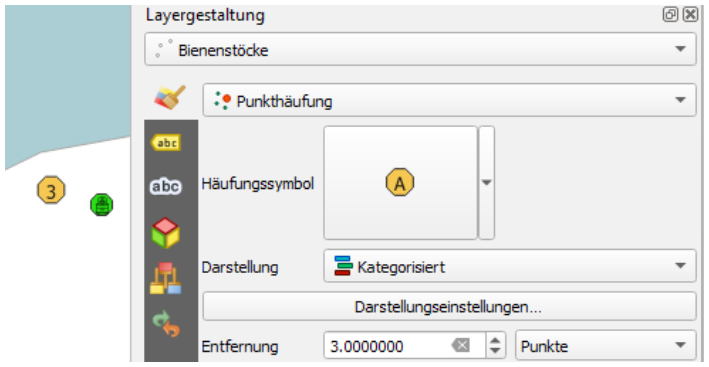
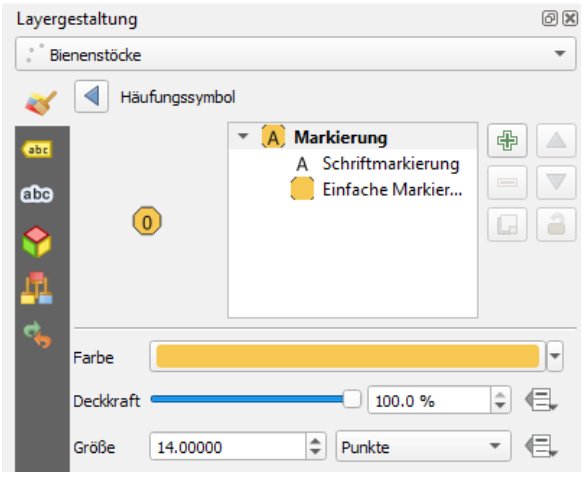


We enlarge the symbols and try out the offset. The settings can then be saved with **Click on marker > Save Symbol... > Name =** `Beehive`.

To get a better overview of the different bee species, Maya categorizes the values Renderer **Categorized > Value:** `bee_species`



Replace overlapping symbols with **Point Cluster Symbol**: When we zoom out a bit on the map, we see that the individual symbols overlap. To avoid this, **Point Cluster** can be used. We configure the Point Cluster symbol so that it matches the point symbol, including the information of how many points lie underneath.

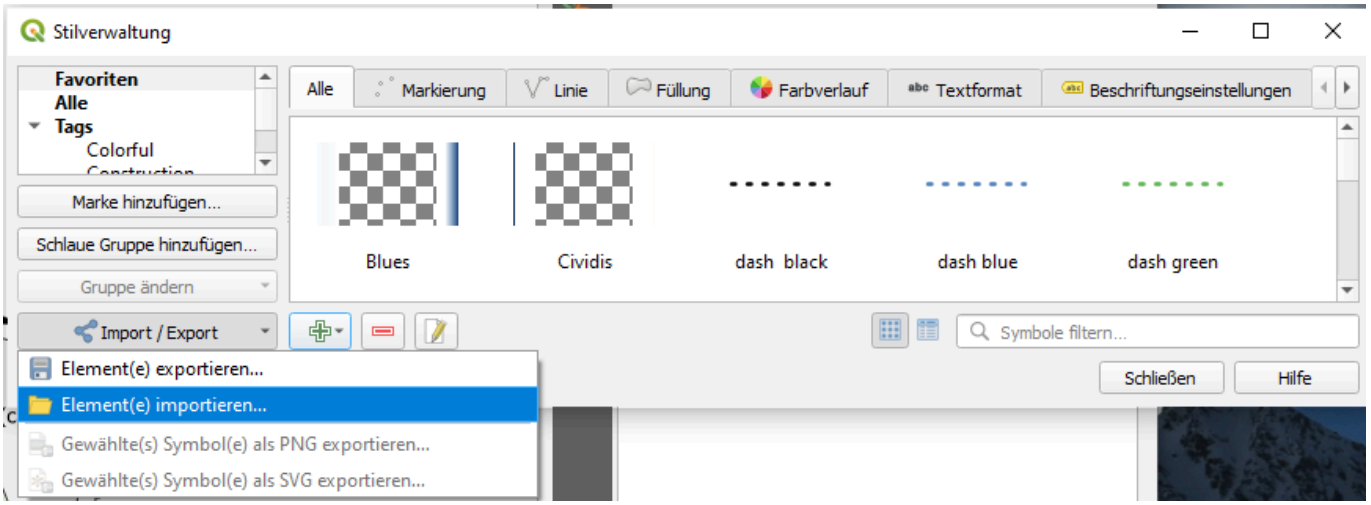
Point Cluster	Settings
	

Excursus: Practical use of QML styles - optional

Maya wants to show in the map view how large her overall area is. She does this by loading an existing style onto a temporary layer.

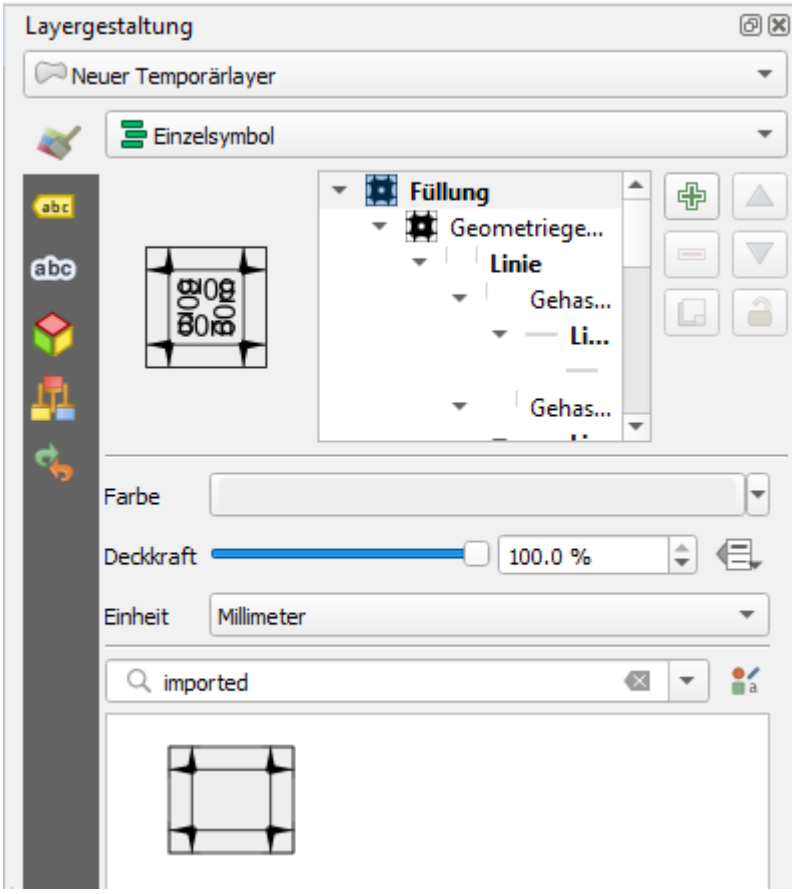
We create a new temporary layer **Layer > Create Layer > New Temporary Scratch Layer (Polygon)**. We roughly draw the area around the plant polygons and beehives.

We go to the **QGIS Style Hub** website (https://style-hub.github.io/) and copy the "Construction" style (**Click on Copy button**).



In the **Style Manager** we choose **Import Item(s)**.

Import from URL > enter URL and paste the previously copied style: https://style-hub.github.io/style-dir/construction.xml. In the Style Manager under **imported** the new style is now listed. We choose the newly loaded Construction style for the new polygon.



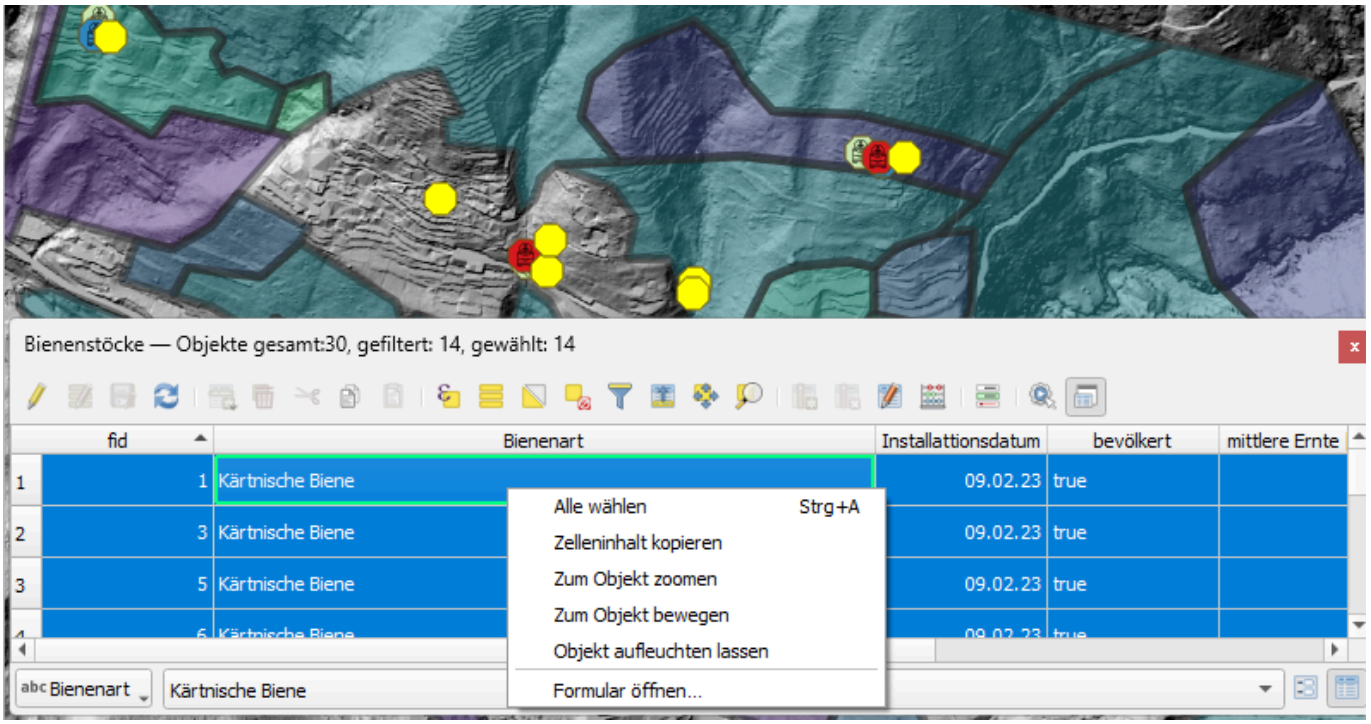
Bee data table

Selecting and filtering

Maya wants to see which beehives house the "Apis Mellifera Carnica" species. She opens the attribute table again for this. **Right-click on Layer > Open Attribute Table**. If desired, the attribute table can be docked: 

At the bottom left, the feature display can be filtered: dropdown at **Show all features > Field filter >** `Bee species: Carniolan bee`. If you select the filtered beehives, they are coloured blue in the attribute table.

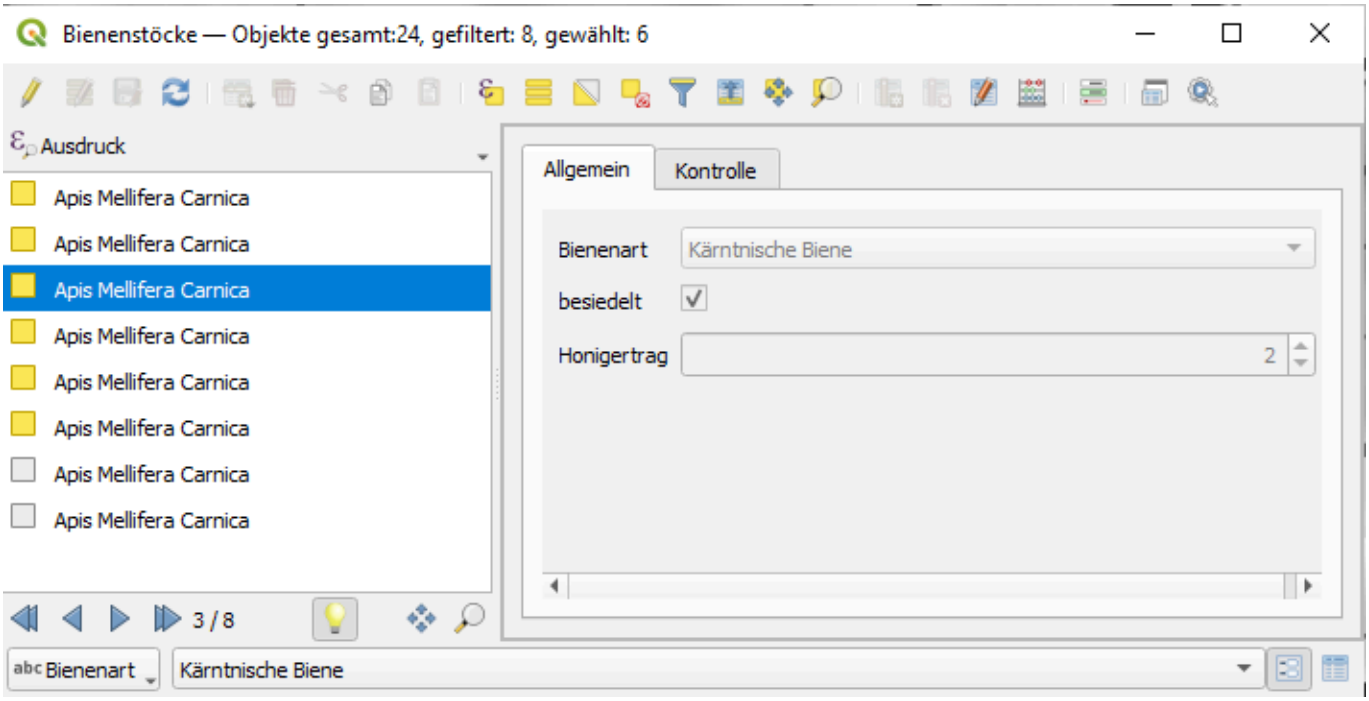
With the **Pan Map to Selection** button you can quickly visualize where the selected features are in the map view. 
In the map view, selected beehives are highlighted in yellow.



With a right-click in a row we can move to, zoom to, or flash a single feature on the map. **Right-click in a row > Pan to feature** or **Zoom to feature**.

Table vs. form view

With the icons at the bottom right of the attribute table we can comfortably switch between table view and form view. In contrast to the table view, in the form view the selected features are marked with a yellow box. Blue is only the entry currently active in the form.



Changing visibility and order of columns (fields) in the project

We can make individual columns invisible (**Right-click on column name > Hide column**) or change the order of the fields. These settings do not change anything in the data itself; they are saved in the QGIS project.

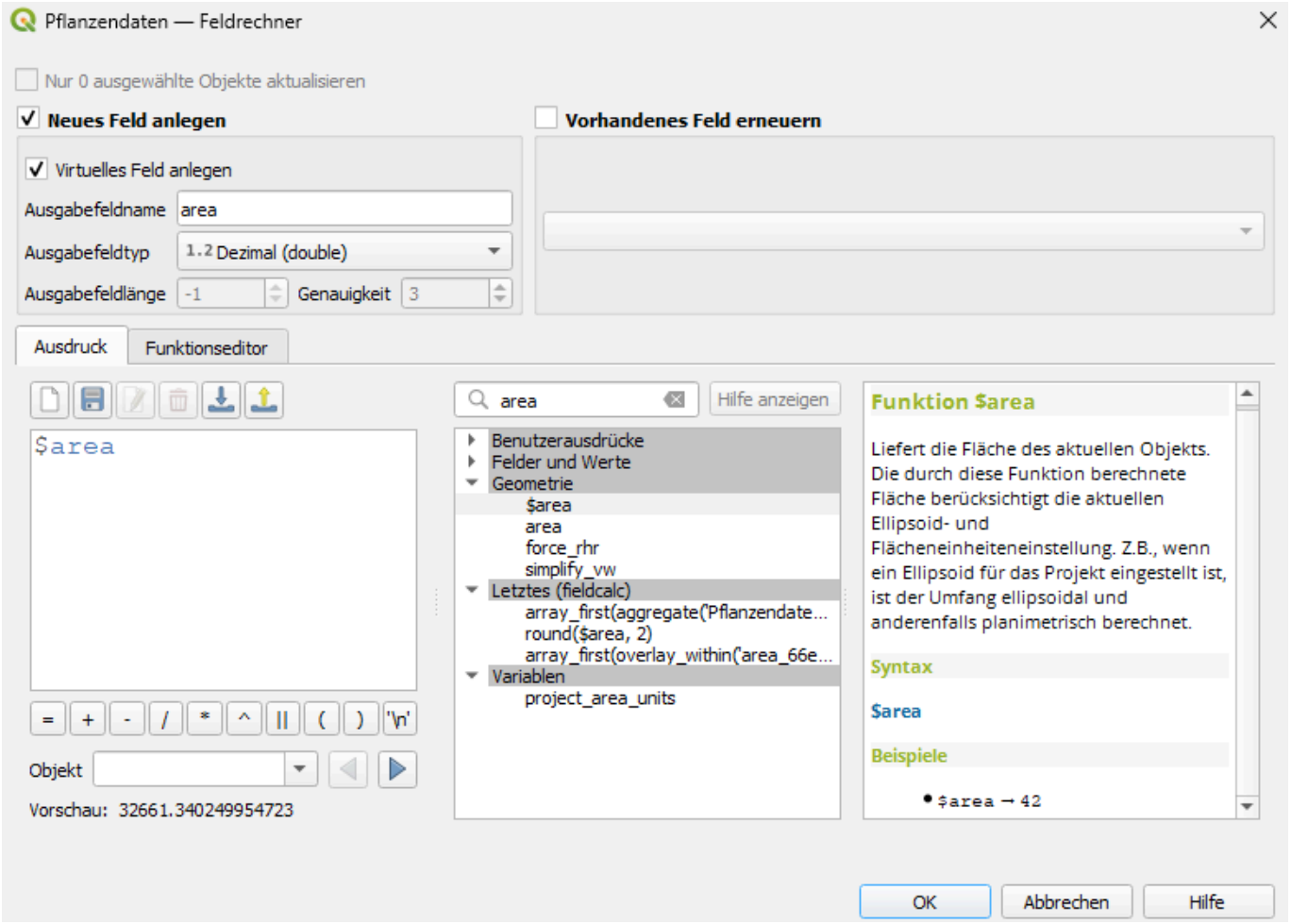
Creating a virtual (dynamic) field in the project

> Virtual fields are attributes that are stored **only in the QGIS project**, but not in the data itself. These fields are dynamic, i.e. they update automatically when a query or expression is defined behind them.

Right-click on the layer `Plant data` **> Open Attribute Table** We can also dock this table. This way we have the view of multiple table data simultaneously.

Maya wants to know how large the respective polygons are. The information should be in a dynamic field that adapts when the polygon is changed.

We open the field calculator and create a **virtual field** of type `decimal`.



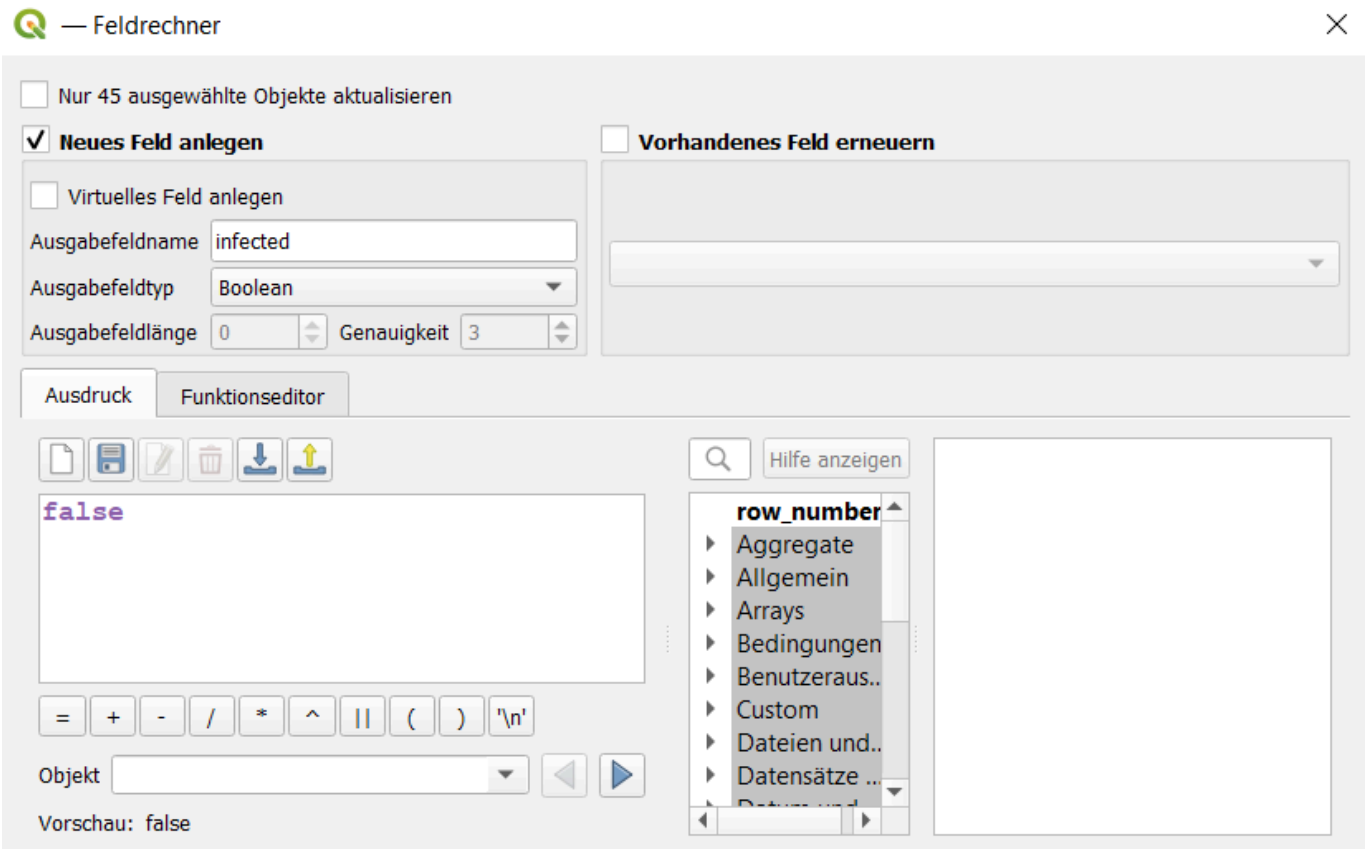
We use the expression `$area`.

In the expression editor we see information about the functions used. Let's look at the new field in the attribute table. It makes little sense to know the area to so many decimal places, so we extend the expression as follows: `round($area, 2)` To overwrite the `area` field, we simply call it `area` again. Otherwise we define a new name such as `area_round`.

Adding fields to the data source

> Non-virtual fields are stored **directly in the data source**. They are not dynamic. In the case of an area calculation, for example, it will not be updated if the geometry of the feature is changed. The value of the field corresponds to the value at the time of the calculation.

We select the `Beehives` layer and open the field calculator from the attribute window. We create a new field (non-virtual) with the name `infected`, the **output field type "boolean"** and the default value (as expression) `false`.



> The data structure of the source files can be changed directly in QGIS. It can be edited via the attribute table and form, or somewhat more clearly in the layer properties.

Open the properties of the `Beehives` layer with a right-click, go to Fields and edit the data source.

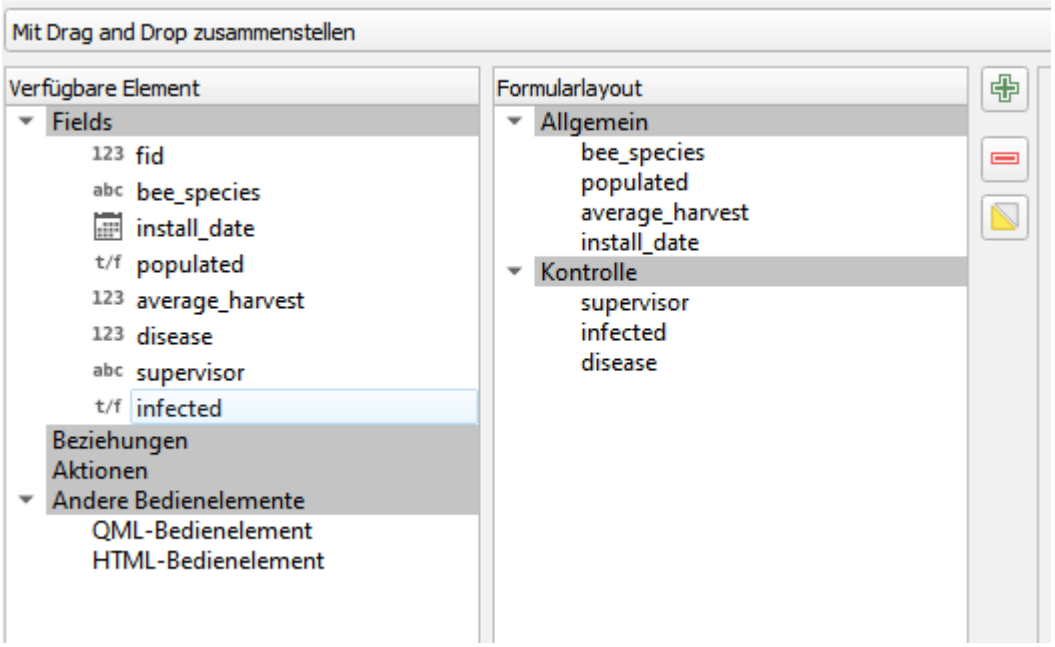
We add a field `disease` of type **text**, and a field `supervisor` of type **text**.

Improving the beehive input form

Maya now has a lot to do; luckily she gets help from her friends. So that they can collaborate on her project, structured forms and a simple form of input validation are needed. So Maya improves the input form for the beehives.

In the **Layer properties > Attributes Form**, tabs and groups can be added to structure the form.

- Tab `General` with the fields: `bee_species`, `populated`, `install_date`, and `average_harvest`.
- Tab `Inspection` with the fields: `supervisor`, `infected`, and `disease`.



If the tabs are switched to **Groups**, the appearance of the attribute form changes.

> A minimal data validation can be defined, for example, by constraints.

The bee species (`bee_species`) should not be allowed to be `NULL`: Checkbox **Not null** (alternatively: expression: `"bee_species" is not NULL`).

It should not be possible to save features as long as no bee species is specified. This constraint should be enforced: Select **Enforce not null constraint**.

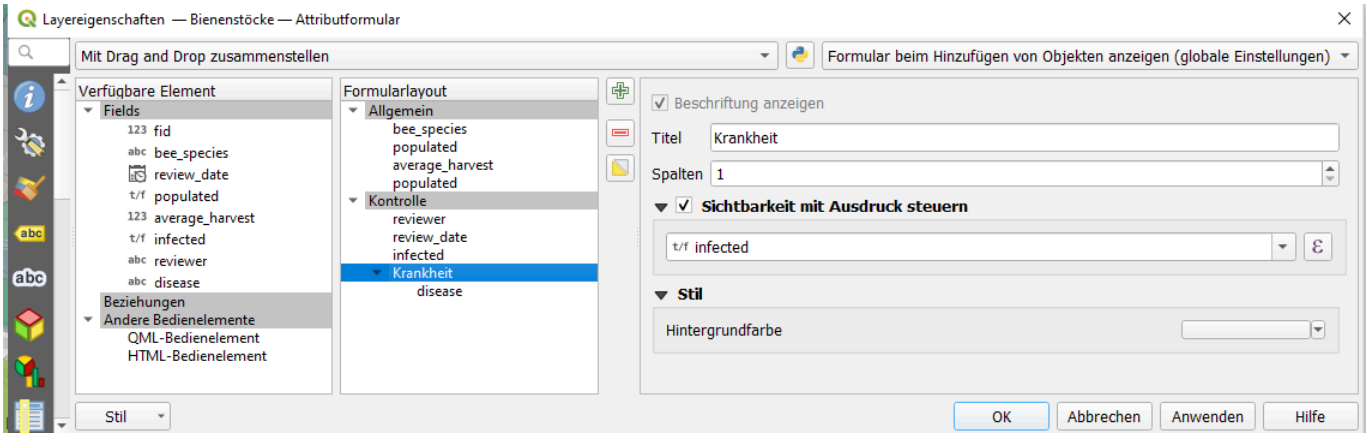
The value for the average harvest `average_harvest` should be greater than 1 kg: `"average_harvest" > 1`.

Since the harvest is still 0 when a new beehive is set up, no hard rule should be set, i.e. the constraint should not be enforced.

Controlling field visibility with an expression - optional

It makes no sense to assign a disease to a beehive that is not infected. So we make the visibility of the `disease` field depend on the value in the `infection` field.

We add a new group and control its visibility with an expression. We move the `disease` field into the new group **Disease**.



Loading the list of diseases as a value relation - optional

Maya has a list of bee diseases. She would like to import it into the project so that she can see it in the attribute form as a selection list.

In the Layers panel we add a new group "Tables". In it we import the file `bee_disease.csv`.

We go to the **Layer properties of** `Beehives` **> Attributes Form >** `disease`. There we choose the widget type **Value Relation** and choose as Key column: `code`
Value column: `description`

With this **widget type** we can also enable multiple selection.

Mobile QGIS - QField - optional

Maya wants to be able to take the project with her on her phone, since she does not bring a laptop when visiting her bees. But she wants to be able to record changes directly.

She downloads the **QField** app onto her phone (for Android on Play Store, iOS, Windows https://docs.qfield.org/get-started/).

Next, she loads the project and the data onto her phone to open them with QField and to test whether the adjustments made to the attribute form are practical for input on the phone.

For syncing between QField and QGIS, the QFieldSync plugin can be loaded. *This will be especially useful when Maya also shares the project with her friends, who sometimes check her beehives for her. Maya uploads the project from the QFieldSync plugin to QFieldCloud (qfield.cloud), where she has previously set up an account, and from there onto her phone.*

Editing polygon geometries

> Up to now we have mainly edited or added attributes (factual data). In editing mode the geometries of the selected layer can also be edited. Note: this again changes the data in the source file.

Matteo's data on the plant species is very helpful, but there are still many gaps. The bees don't care whether they collect their nectar from a captured area or not. Maya now knows the area like the back of her hand and wants to complete the plant data.

In the following, several editing tools are tried out.

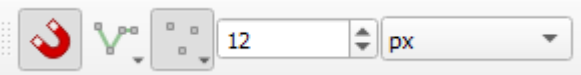
Adjusting boundaries with the vertex tool

Since dandelions around Maya's house have multiplied strongly, she enlarges the polygon with the vertex tool.

Snapping to existing features

(Snapping / Tracing)
Digitizing a new polygon seamlessly adjoining an existing one is relatively difficult without a helper tool. For this purpose we add the snapping toolbar:

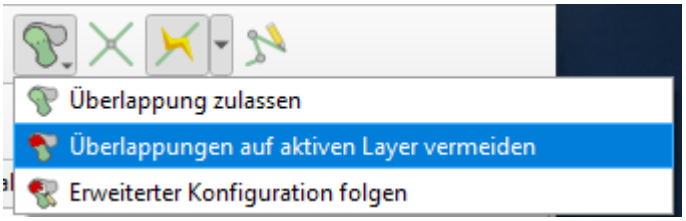
View > Toolbars > Snapping Toolbar



We activate **Snapping** and capture a polygon. We see that the vertices jump to other vertices.

Avoiding overlaps of polygons

When drawing a new polygon, the existing geometries are taken into account so that only the new part is drawn.



Tracing

By activating tracing, the lines of existing features are automatically traced without having to set every point. With this tool features can be digitized quickly.

Moving multiple vertices together

Another useful setting is **Topological Editing**. By activating this option, for newly digitized lines or polygons, vertices are also placed at all vertices of a boundary polygon, even on a straight line. This is the optimal prerequisite for subsequent editing with the **vertex tool**.

Advanced editing of geometries - optional

We add the **Advanced Digitizing** toolbar. **View > Toolbars > Advanced Digitizing Toolbar** Here we see the basic functions as icons. We can move, rotate, simplify, etc. geometries. The **Advanced Digitizing** tool is a powerful tool to capture complex geometries and to work with right angles, parallels, etc.

Data processing and analysis: number of beehives per plant area

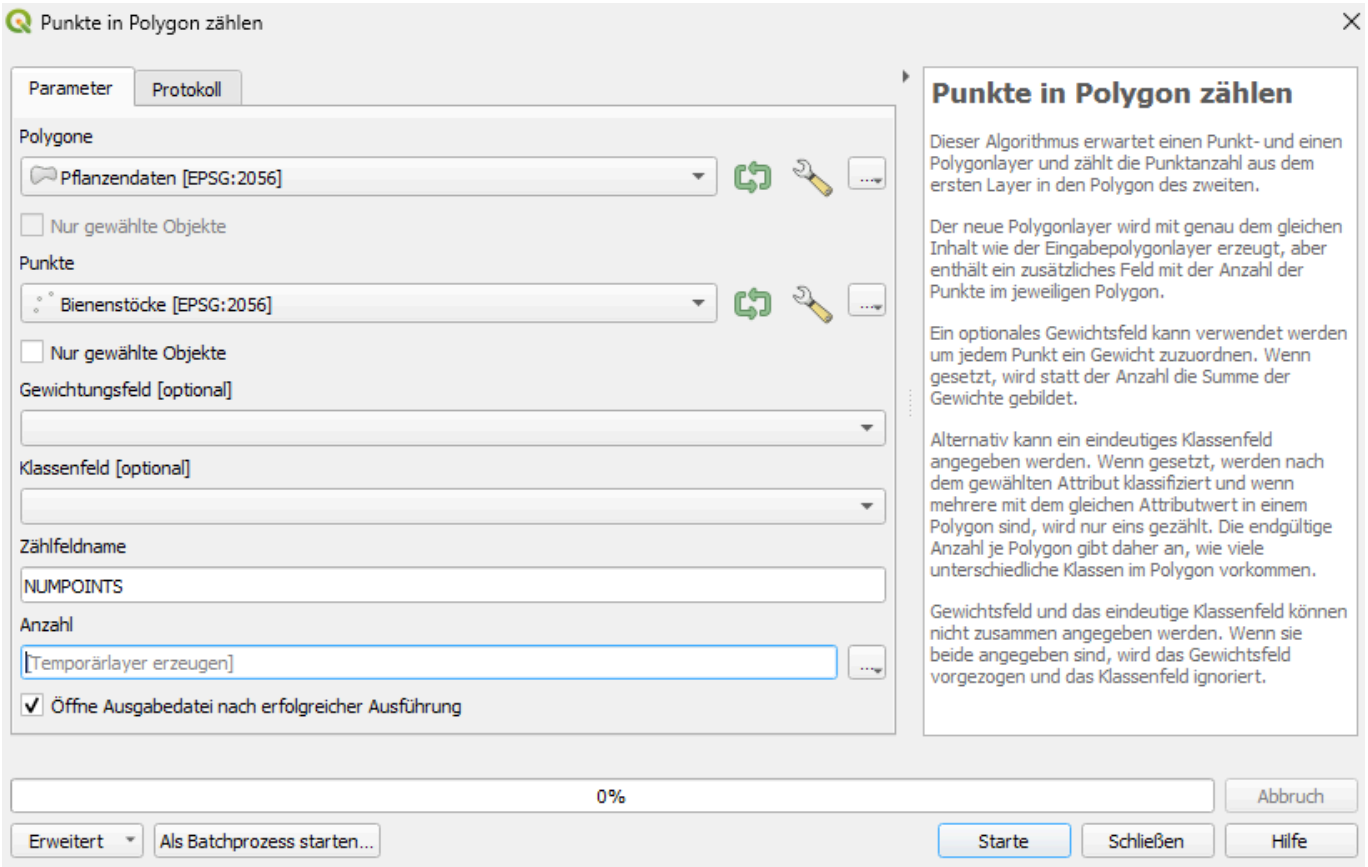
> The **Processing Toolbox** offers a large selection of processing tools and process algorithms for editing and analyzing geographic data.

To get an overview of her rapidly growing data, Maya uses the processing tools.

Processing > Toolbox

Maya needs an algorithm to count the beehive points within a particular plant species area.

We type "*count*" in the search bar at the top of the toolbox, and under Vector analysis the tool **Count points in polygon** is shown. (via **Vector > Analysis Tools** we can also get to the **Count points in polygon** tool)



We open the tool. We choose the `Plant data` layer under Polygons and `Beehives` under Points.

We do not specify an output folder and name and so generate a new temporary layer; a copy of the `Plant data` with an additional field for the number of points (`NUMPOINTS`).

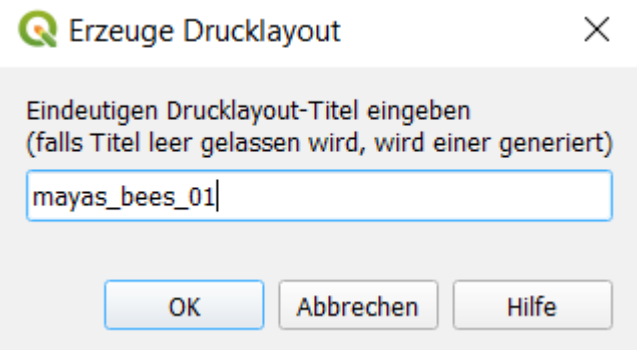
We can now display this new field as a label, making the number of points per `Plant data` polygon visible on the map.

Creating a print layout

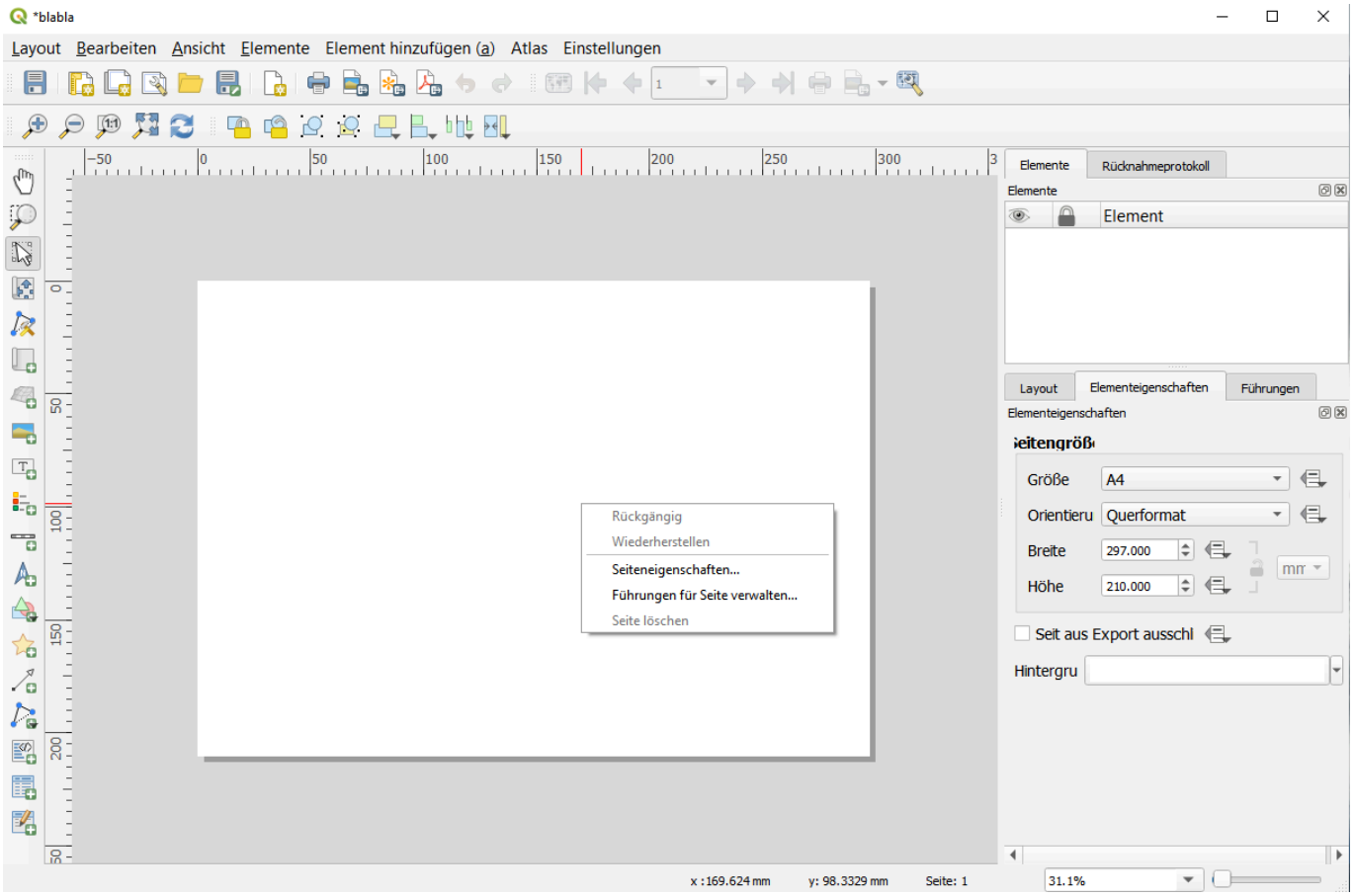
> With the **Print Layout** you can prepare map views for printing (or export as SVG or PDF file). Not only the map itself, but many additional elements such as information about the attributes, free text fields, photos, legends, and scale bars can be integrated into a print layout.

The municipality of Lavertezzo asks Maya whether she would like to set up an information sign on the road that informs tourists about her bees. Maya wants to create a nice layout for this.

We create a new print layout. **Project > New Print Layout**

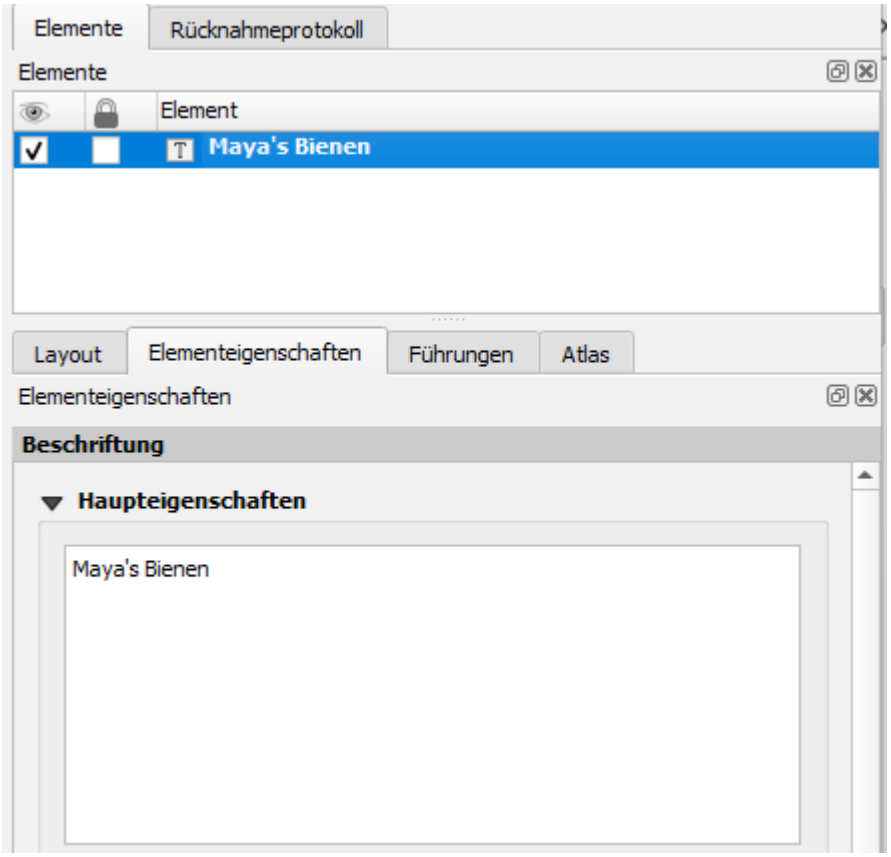


As preparation we choose the desired paper size and format. With a **right-click on the empty sheet** we get to the page properties.

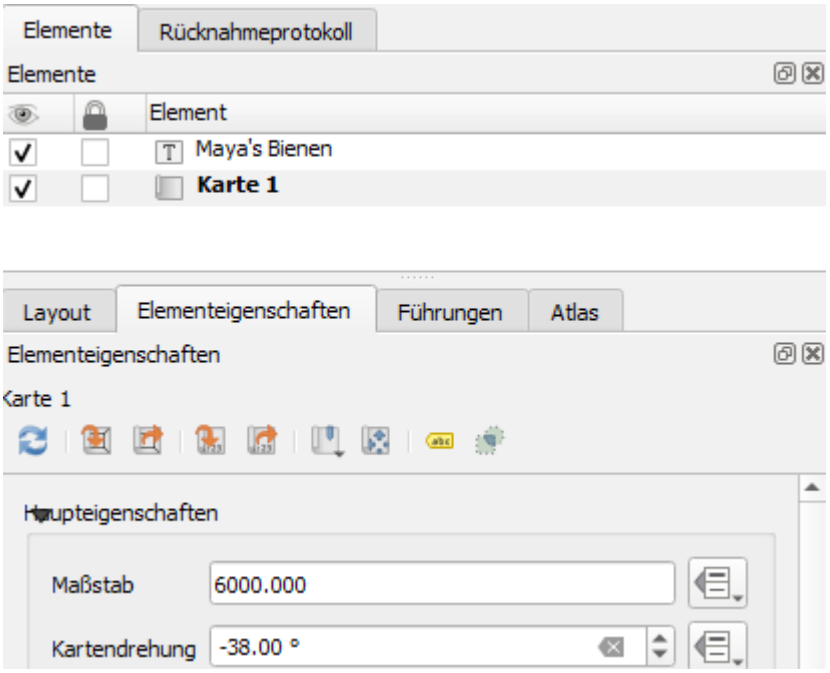


We choose A4 and landscape. As an aid for the graphical work, we activate **View > Smart Guides**.

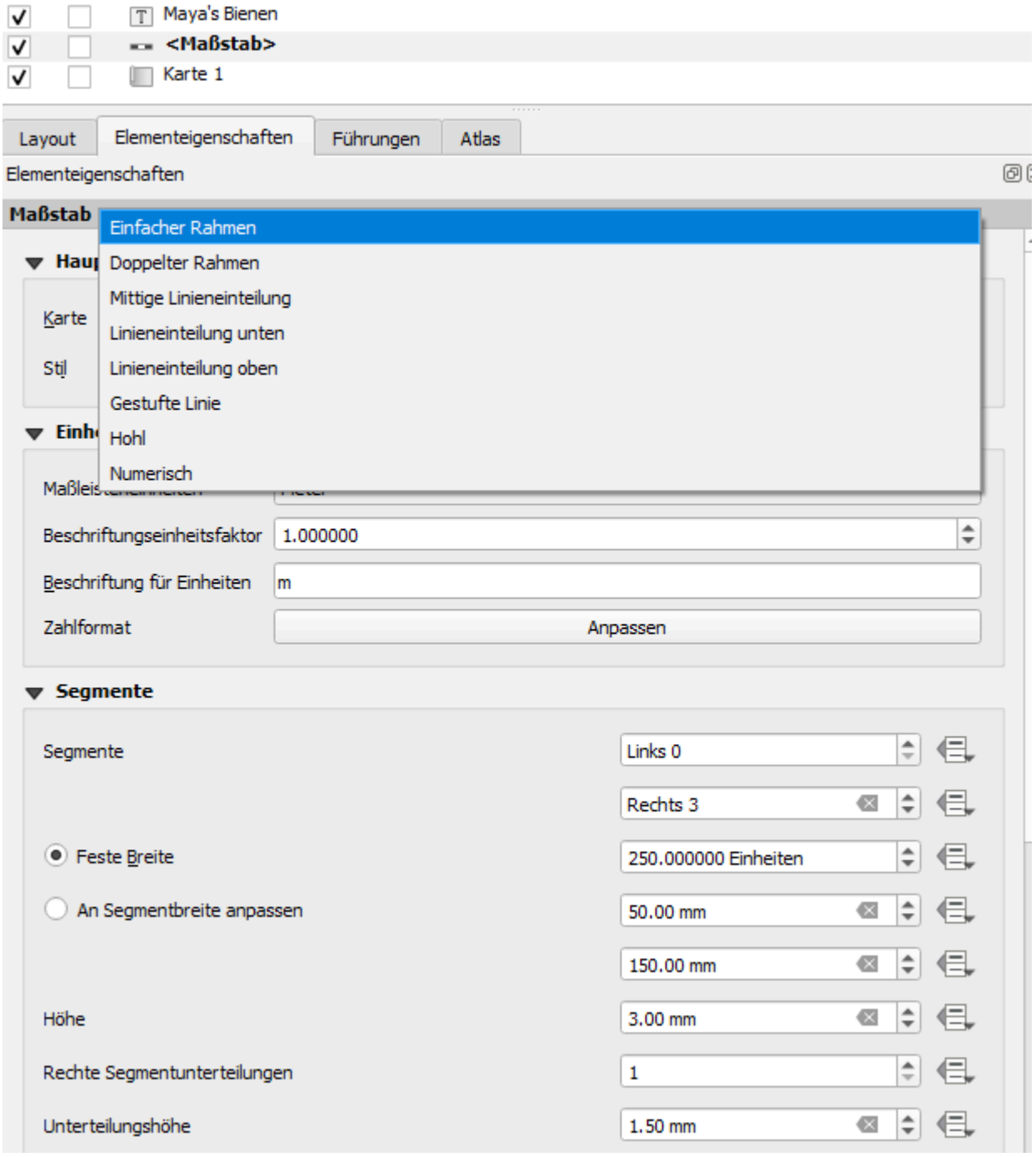
We add a label for the title: **Add Item > Add Label: click into the map > drag a rectangle with the mouse button held down > release**. In the item properties of the label we set the title: `Maya's Bees`.



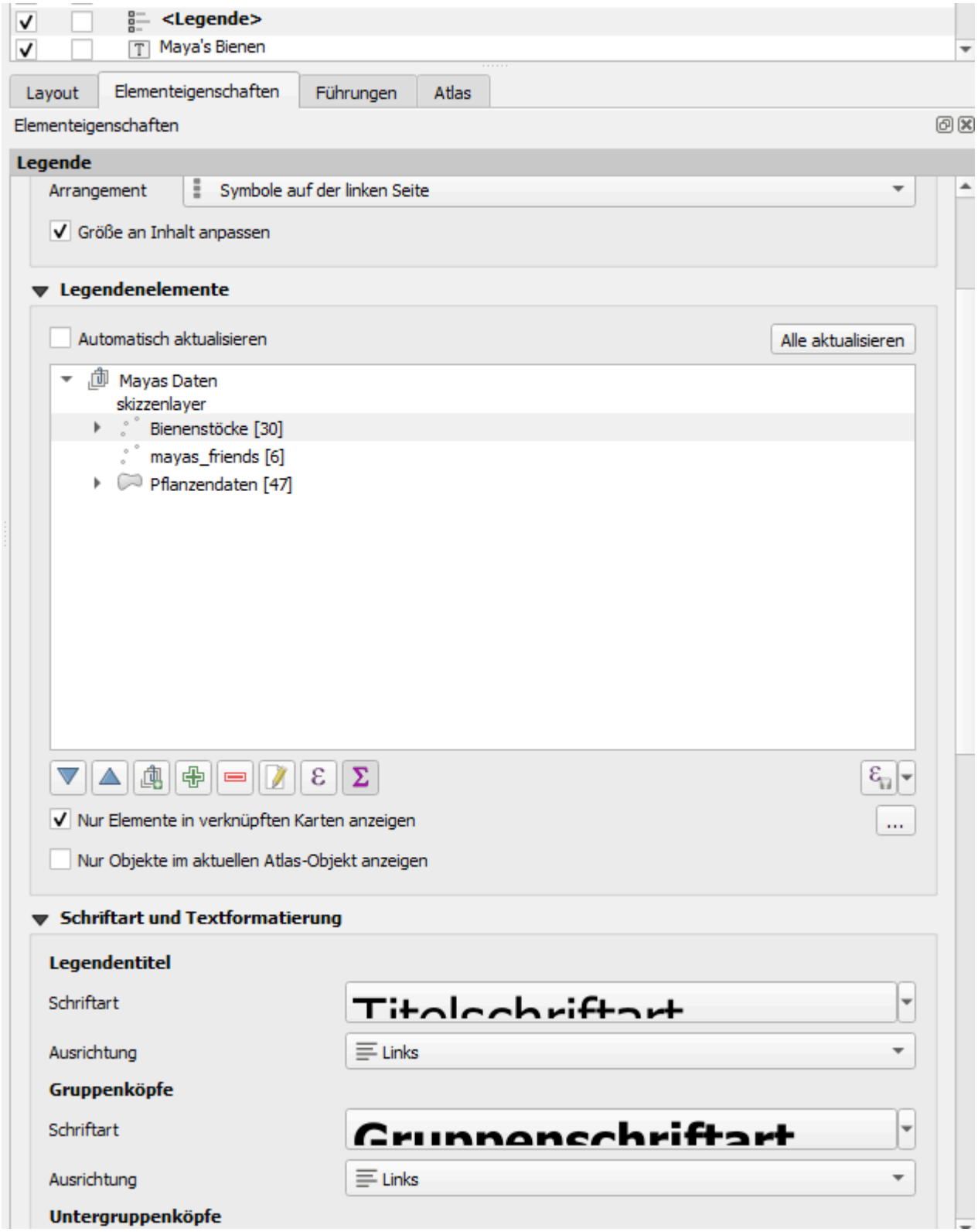
We adjust the font size and the style of the title: **Click on the "Font" button** (below Appearance). We add a map **Add Item > Add Map** (draw a rectangle) and choose the appropriate section with **> Move item content**. We set a suitable map scale.



The map can be rotated. So that it is always clear where North is, we add a **North arrow**. Then we add a **scale bar**. A scale matching the zoom level is automatically suggested. The properties such as style or number of segments can be adjusted in the properties of the scale element.

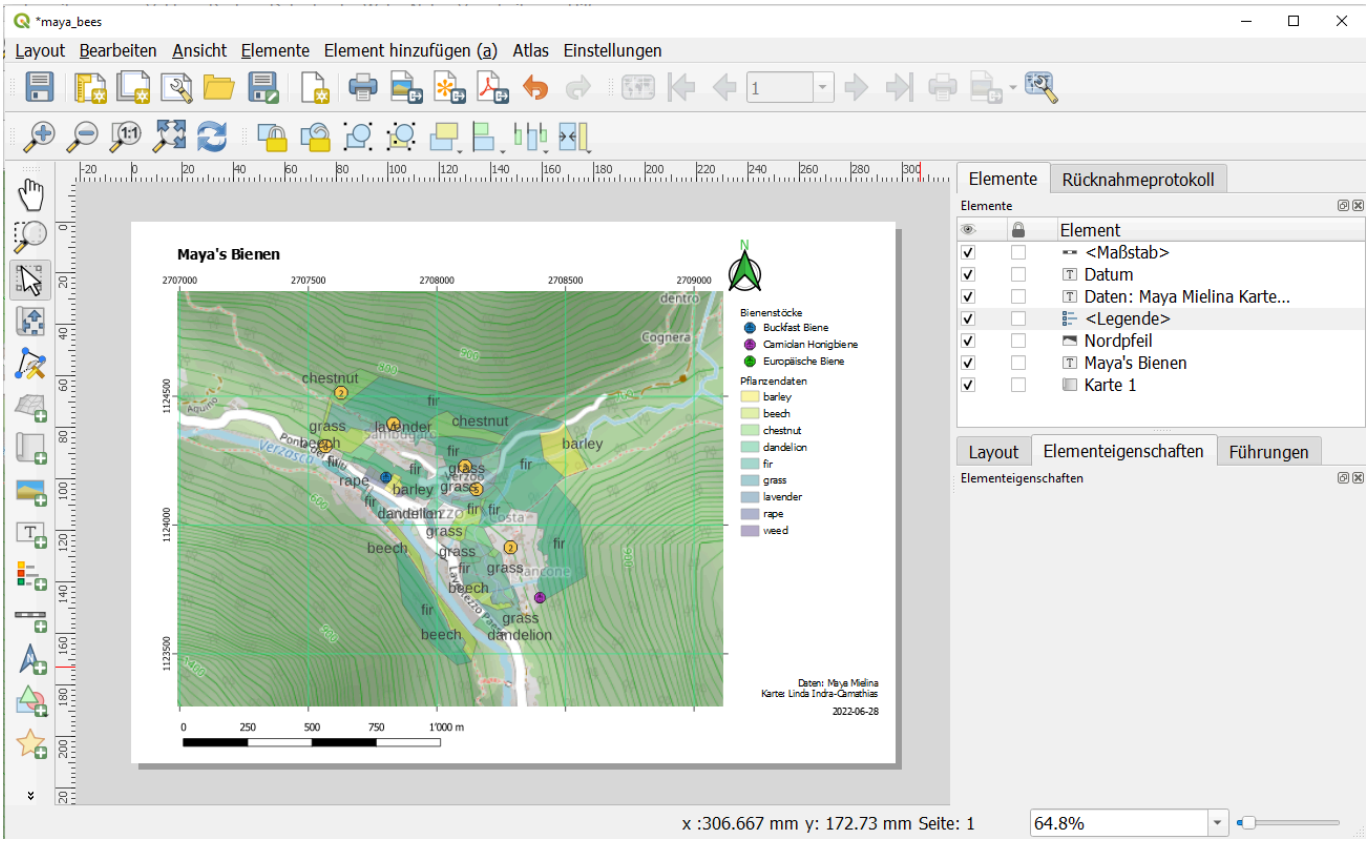


We add a **legend**. And adjust what should be displayed: deactivate **Auto-update**.

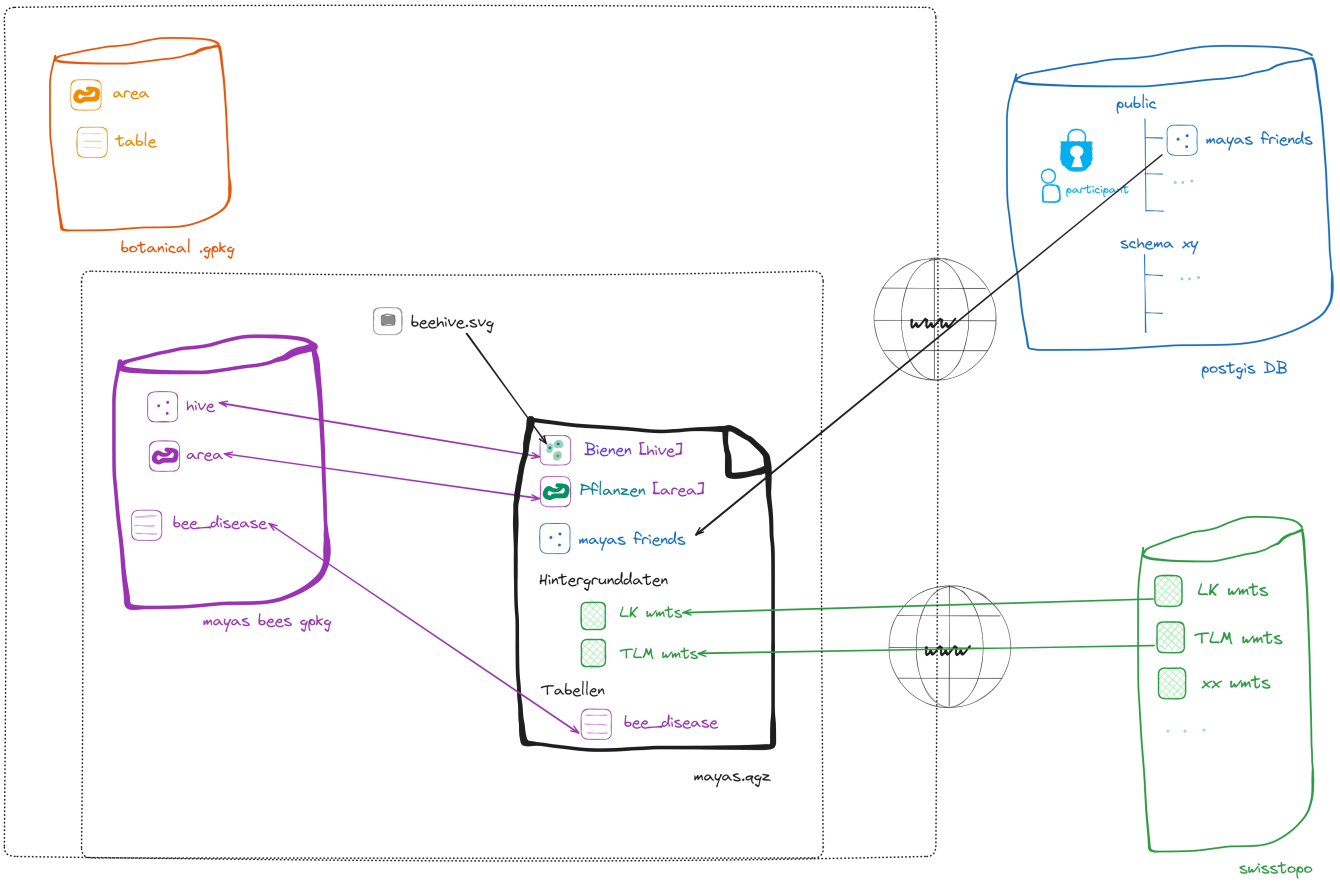


We edit the font under **Font and text formatting > Click** on the corresponding bar *Maya also wants her name on the map and to indicate the data source.*

We insert a label for this: **Add Item > Add Label**. In addition, the date of map creation should always be current automatically: **Add Item > Add Dynamic Text > Current date > choose desired format**.


*maya_bees
Layout Bearbeiten Ansicht Elemente Element hinzufügen (a) Atlas Einstellungen
Maya's Bienen
Bienenstöcke
Buckfast Biene
Carnician Honigbiene
Europäische Biene
Pflanzendaten
barley
beech
chestnut
dandelion
fir
grass
lavender
rape
weed
Daten: Maya Mielina
Karte: Linda Indra-Camathias
2022-06-28
0 250 500 750 1'000 m
Elemente
Rücknahmeprotokoll
Element
<Maßstab>
Datum
Daten: Maya Mielina Karte...
<Legende>
Nordpfeil
Maya's Bienen
Karte 1
Layout
Elementeigenschaften
Führungen
x :306.667 mm y: 172.73 mm Seite: 1
64.8%

Project structure



Documentation, further links

https://www.opengis.ch/ (commercial support)
https://qgis.org (QGIS main page)
https://docs.qgis.org/3.34/en/docs/user_manual/index.html (user manual)
https://issues-qos.org/projects/qgis/issues (report a bug)
http://plugins.qgis.org (plugins)
https://qfield.org (QField)
https://qfield.cloud/ (QFieldCloud)
https://qgis.ch (QGIS user group Switzerland)