Introduction to QGIS 3: Data Analysis (Chapter 4)

A worksheet

In the last chapters you've learned how geodata is captured and displayed in a GIS. Thus everything is prepared for the "speciality" of GIS, namely the derivation of new information from existing spatial data. This derivation of new information is called **analysis**.

Learning objectives

This chapter has the following learning objectives

- State the difference between queries and manipulations in your own words

- List which query types there are and explain them with the help of one example each

- Formulate a correct SQL query for a thematic query

Theory

❗ *Exercise 1*
As an introduction to the data analysis chapter, solve the following tasks:

1. Search 1: Go to the homepage http://tel.local.ch/de/ and enter the name and, if known, the place of residence of a person you know who might have an entry in the phone book (e.g. your parents or grandparents). Then click on the search button. What is the result of your search? In which form are the input and the output of the search, if you look at the search on the page tel.local.ch in general?

2. Search 2: Next, go to the homepage http://www.sbb.ch/ and look for the next connection from the station closest to your home to Chur. Look again at the input and output values. The results themselves are not so important as the form in which the results are output. Make some notes about it again.

3. Comparison: What do you notice if you now compare the type of results of search 1 and search 2? Write down your findings.

Keep your solutions in your mind. We will come back to them later in this chapter.

As already mentioned in the introduction, the derivation of new information from existing spatial data is one of the most valuable and most important functions of a GIS.

In general, we differentiate between two types of analysis: **queries** and **manipulations**. For the queries the data remains unchanged, manipulations either change the data or even generate new data.

Queries

There are three ways in which a query can be executed.

1. Queries based on factual data. Example: Select all lakes with a water depth greater than 100 meters. These queries are called **thematic queries or selections**.

2. Query relationships between different geometric objects. Example: Select all lakes that are completely within a canton. These queries are called **topological queries or selections**.

3. Queries about the geometry of objects. Example: Which main locations are at a distance of less than 20 km around Lake Lucerne? These queries are called **geometric queries or selections**.

❗ *Exercise 2*
Consider what kind of queries the introductory exercise was?

Database language SQL

To perform thematic queries, the database language **SQL (Structured Query Language)** is used. If you have already worked with databases, you will probably already know the language. Then the next section is Repetition. For everyone else it is a short introduction to the most important structures and expressions of a query with SQL.

A simplified *SQL query* is structured as follows:

```
SELECT select-expression
FROM table
WHERE where-expression
```

Overview of the variables that you should know for simple queries:

select-expression: Names of all attributes, which should be displayed as the result of the query. A * means, all attributes are displayed. For thematic queries in a GIS this is the only query possible. Therefore, this variable must not be set in the GIS.

table: Name of the table, which contains the attributes needed for the query.

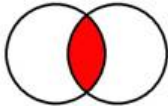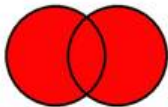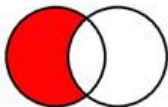**where-expression:** The conditions to filter the data.

Overview over the SQL expressions, which use the where-expression, and their meaning.

SQL Expression	Meaning
=	equal
<	smaller
<=	smaller or equal.
>	greater.
>=	greater or equal.

SQL Expression	Meaning
<>	not equal, different.
LIKE	Comparison of characters. The following characters can be used as placeholders: %: Placeholder for any number of characters. For example, LIKE "F%" for all entries that begin with the letter F. _ : Placeholder for exactly one character. E.g. LIKE "Me_er", if you are looking for a person and don't know if you want to use Meier or Meyer's last name. For LIKE "M__er" it could also be Mayer.
IS NULL	Cell is empty.
IS NOT NULL	Cell is not empty.
BETWEEN	between to values.

You can combine different requirements together and use logical Operators if *Condition 1* **and** (**AND**) *Condition 2* or *Condition 1* **or** (**OR**) *Condition 2* or *Condition 1* **and not** (**NOT**) *Condition 2* should be valid.

Here's a wrapup to illustrate everything as a venn diagram:

AND-Operator: Which element belongs both to set A and set B?	
OR-Operator: Which element belongs to either set A or set B	
NOT-Operator: Which element belongs to set A but not to set B	

❗ *Exercise 3*
Try it now for yourself and formulate the following query in SQL:

1. All lakes should be selected whose area is greater than 20 km^2 and which are deeper than 400 m.a.s.l..

2. *The search that you did as an introduction to theory as Search 1 on the homepage* http://tel.local.ch/de/.

QGIS Exercise

The "speciality" of a GIS should of course not only be thought in theory. In this chapter you will perform some thematic and spatial queries.

Learning objectives

At the end of this exercise, you've mastered the following points:

- Select elements based on thematic and spatial criteria.

- Calculate a new value for a field from existing values.

Thematic selection

At the beginning of this exercise we would like to perform a simple query. You already know it from the SQL exercise in the theoretical part:

All lakes should be selected whose area is larger than 20 km^2 and which are deeper than 400 m.a.s.l.

To select these lakes, you could now look through each row of the attribute table and determine which objects belong to them and which do not. But if you have a lot of objects, this kind of search would quickly take a lot of time.

It's easier if you use the predefined search masks. If you are only looking for a name or a number, the easiest way is to use the *Field Filter* in the attribute table.

Try this search with a small example. Open the attribute table of the lakes. In the lower left area you will see a line that says *Show All Features* and a menu that you can open. Search in this menu, in the *Field Filter*, for the attribute you want to sort by. Additionally, you have to enter the term you want to filter by. For example, you want to search for all lakes in which "Lago" appears in the name. Select the attribute "name" in the menu at the bottom left and search for the term "Lago".

To see immediately how many elements have been selected, you can activate the *Show Selected Features* option.

❗ *Exercise 4*
How many solutions Do you get?

Let's get back to the opening example. This query is a bit more complicated. Open the query editor by clicking the *Select features using an expression* button. Here the query can be formulated in the form of a *SQL* query. However, only the *where clause* of the *SQL* query has to be entered. From "*SELECT * FROM Seen WHERE max_depth > 100*" only the part "*max_depth > 100*" has to be typed into the query field.

The single components of the where-expression can be composed by double-clicking on the window to the right.
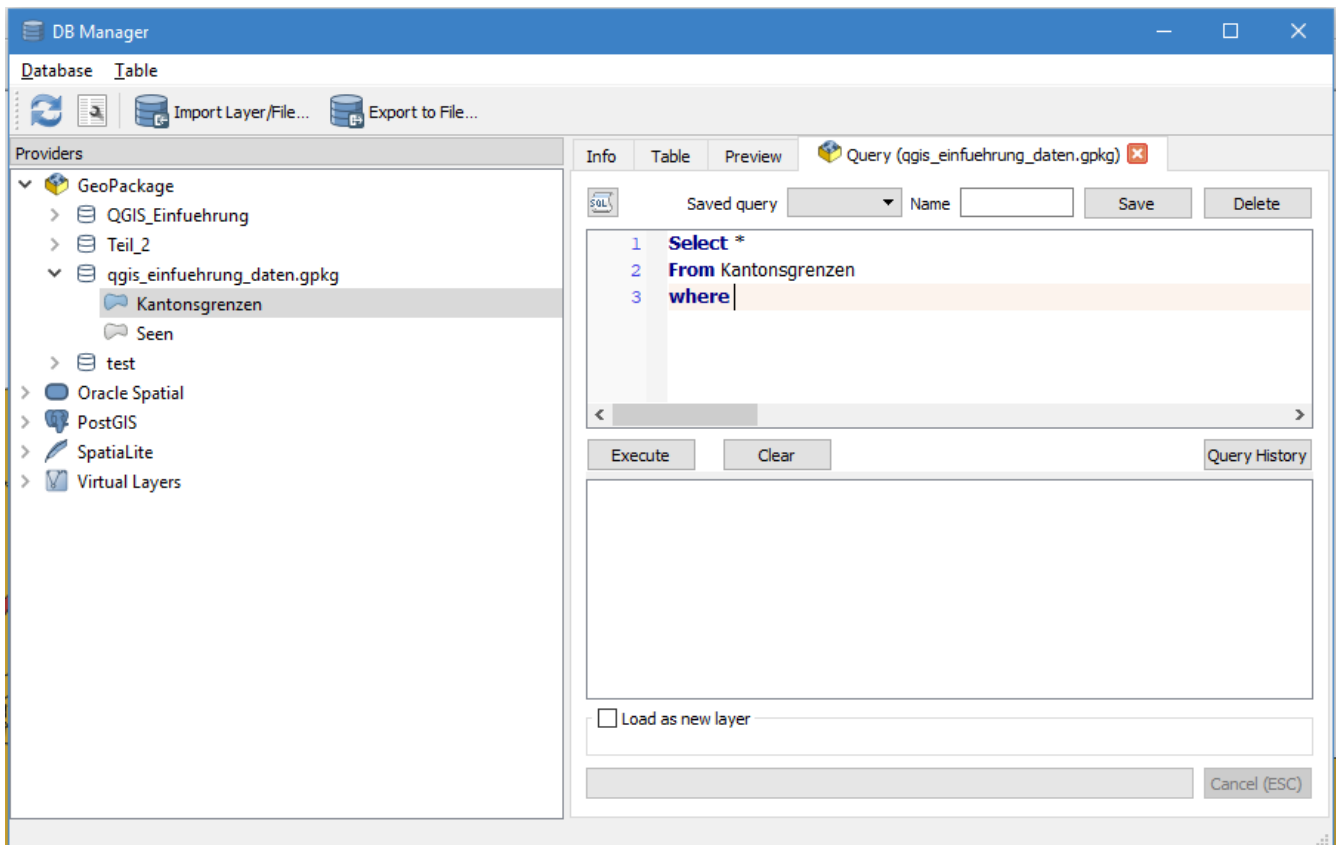
❗ *Exercise 5*
Now formulate the where clause of the SQL query for the entry task and execute the query. How many lakes meet these conditions?

In more demanding analyses you want to use the whole SQL language. The field filter is not sufficient for this. For your own SQL queries, you can use the *DB Manager* in QGIS, which you find in the QGIS menu under *Database*. Given that data is already loaded in the project, you can use *Select Virtual*

Layer. If you open it, you will see *Project Layer* and below it you see the queryable layers that are loaded into the project, i.e. in our case Seen (lakes) and Kantonsgrenzen (cantonal borders). To create your own SQL query, select *Project layer* and click at the top of the SQL window icon. A new, two-part dialog appears with an empty text field at the top and an output field at the bottom. In the upper field, the cursor is positioned at line 1 and you can start there to write the SQL query.

The SQL query can be executed by clicking the button *Execute*. If the query was successful, the result is being displayed in the output field. If you select the checkbox *Load as new layer* before executing the SQL query, the result will be loaded as a new layer.

❗ *Exercise 6*
Formulate the same query as in the previous task with the where-clause, but now as a standalone SQL query. What is the SQL query?

Spatial selection

In the first part of this section a query was created based on values from the attribute table. Now a query has to be written which takes the position of the object into consideration, also called spatial selection.

You can find the tools for the spatial queries when you click on the button *Toolbox* and select *Select by location* from *Vector selection*

All cantons are to be selected in which at least one of the lakes lies entirely within the canton (i.e. the canton doesn't share the lake with another canton or country).

The *Select features from* layer gives information from which layer we select the elements. In this case we select from cantons and because of that we choose the layer canton boarders. The lakes are

noted in the *By comparing to the features from* layer. The box *Selected features only* needs o e checked so that only the lakes with a depth of at least 100 meters are selected.

The lakes should be completely within the boarders of a canton. I.e. the cantonal areas fully contain the lake areas. Therefore the topological operation *contains*. By selecting topological operations one always starts from *Select_features from* and checks how it compares to *By comparing to the features from*.

❗ *Exercise 6*
Run the query. How many cantons fulfill the condition?

❗ *Exercise 7*
How would you need to modify the query if all lakes should be selected which are part of more than one canton or country?

❗ *Exercise 8*
How many lakes meet the condition?

Calculate fields

The next step is to create a new value using existing attributes. More precisely, the population density of the cantons is to be calculated. This is done using the following formula:

Population density per km^2 = Number of inhabitants / surface area in km^2

For this calculation, the attribute table of the cantons must be used. Select the option *Open field calculator*. If you want to write the density value into an already existing column, you can mark *Update existing field* in the mask and select your desired column in the dropdown menu. In our case, however, we would like to create a new field. Therefore enter under *Output field name* the term *Density*. Choose as *Output field type integer* and as *Output field length 10*.

If you want to calculate the density value for all and not only for the selected cantons, you have to make sure that the cross is not set for *Only update x selected features* .

Now you can formulate the *Field Calculator Expression* according to the population density formula. All you have to do is enter the expression after the equal sign.

We would also like to calculate the density as a whole rounded number without decimal places. Therefore the option to *Whole number* must be selected.

❗ *Exercise 9*
What do you type into the calculator?

After the calculation the editing mode must be switched off again. If you are satisfied with the calculation then save the changes. If not, reject them.

Combined selection

In a further query, determine the names of all the main towns whose associated cantons have a population density of more than 200 inhabitants per km^2 and which joined Switzerland between 1800

and 1900.

A query is now to be carried out in which both thematic and spatial components play a role.

🔴 *Exercise 10*
Think about which component is thematic and which is spatial.

For this query we'll do two steps; in the first one we'll run the thematic query. Open for that the attribute table for cantons and select with help of *Select by expression* all cantons with the population density greater than 200 people per km^2 and joined Switzerland between 1800 and 1900.

🔴 *Exercise 11*
What's the where-expression?

Now all cantons are selected which fulfill this condition.

After that we need to find the corresponding the main towns, this is done with the help of spatial selection. For that open the tool *Select by Location*.

🔴 *Exercise 12*
*Which layer is the **Select features from** layer and which is the **By comparing to the features from** layer? Which topological operation has to be chosen?*

Before the execution of the query, check that the options are set, so that from the cantonal borders only the selected ones are taken into account.

🔴 *Exercise 13*
Which main towns are the result?

Chapter test 4

Question 1

In chapter 4 you have learned to differentiate between to variants of analysis, namely queries and manipulations.

What's the difference between these two variants?

Write down your answer using your own words and add an example for each.

Question 2

You have learned to formulate thematic queries with SQL.

Create an SQL query for the following question:
"Which cantons have a population density greater than 200 inhabitants per km^2, joined Switzerland before 1600 A.D. and have a name that does not begin with the letter Z?"

Use the simplified version of the SQL query and mark the typical elements of it separately. The

names of the tables and attributes can be taken from the data given to you.

Question 3

Besides the thematic queries, there are other variants which can be combined.

For the following question create an SQL-Query:
"Select all cantons which joined Switzerland after 1600 and have contact with a lake greater than 50km^2"

When answering this question, only the lakes stored in the dataset qgis_einfuehrung_daten.gpkg must be considered. Perform the query and write down step by step how you proceed. Small hint: Think about the task from the exercise, in which the names of all main locations were selected, whose associated cantons had to fulfill certain thematic conditions.