

## 6 Der QGIS-Ausdruckseditor

### 6.1 Einführung

Mit dem QGIS-Ausdruckseditor können Sie Abfragen und Berechnungen auf Grundlage von Attributwerten und Geometrieeigenschaften durchführen. Die Syntax basiert auf der Datenbank-abfragesprache SQL und orientiert sich an PostgreSQL sowie deren räumliche Erweiterung PostGIS. Die Abfragesyntax ist somit Standardisiert und im Wesentlichen nicht QGIS-spezifisch. Die Abfragen entsprechen der SQL-WHERE-CLAUSEL zur Auswahl einzelner Zeilen. Seit QGIS 2.16 gibt es Aggregatfunktionen zur Gruppierung von Zeilen. Vollständiges SQL lässt sich in virtuellen Layern, über die SQL Editoren in der Toolbox oder im Datenbankmanager umsetzen. Der Ausdruckseditor ist im QGIS nicht nur ein einfacher Editor für Attributabfragen sondern auch ein zentrales Werkzeug zur Feinsteuerung von Symbolisierungen und Beschriftungen. Der direkte Zugriff auf Geometriefunktionen erlaubt es, verschiedenste räumliche Kriterien in Abfragen zu integrieren.

Der Ausdruckseditor erscheint in drei Formen

x **Mit einem Ausdruck wählen**

Objekte werden über Abfragen, die auch Bedingungen oder Berechnungen enthalten können ausgewählt. Es können über Abfragen auch Objekte in einer Auswahl abgewählt werden.

x **Der Feldrechner:**


Die Inhalte vorhandener oder neuer Spalten werden auf Grundlage von Berechnungen oder Abfragen neu erstellt. Es können einfache Werte eingetragen werden oder auch komplexe Berechnungen durchgeführt. Dabei ist es möglich, in einem Ausdruck auch unterschiedliche Berechnungen für unterschiedliche Bedingungen durchzuführen

x **Ausdruckseditor in Symbolisierung und Beschriftung**

Mit Hilfe von Ausdrücken, lassen sich spezielle Bedingungen für Darstellung und Beschriftung definieren. z.B. kann die Schriftgröße über die Flächengröße von Polygonen gesteuert werden. In der regelbasierten Darstellung erfolgt die gesamte Steuerung der Symbolisierung über sql-basierte Ausdrücke. So lassen sich Symbolkategorien abfragebasiert aus den Beziehungen beliebig vieler Attributspalten generieren


**Objekte auswählen über Abfragen nach Attributwerten mit Hilfe eines Abfrageeditors**

Es gibt in QGIS zwei Abfrageeditoren:

**Ansicht > Auswahl > Nach Ausdruck wählen** 

**Ergebnis der Abfrage wird als Auswahl markiert.**  
Abfragesyntax lässt sich als Vorlage speichern.

**Achtung: Abfrage vor O.K. immer testen oder kopieren! Weil bei Syntaxfehler wird die Abfrage geschlossen und der eigene Abfrageentwurf ist verschwunden!**

Ergebnisse lassen sich über 'Layer > speichern'  
(Kreuz bei nur gewählte Objekte..) in ein neues Shapefile speichern.  
Die Auswahl ist über **Ansicht > Auswählen aufheben** zu löschen  
Viele Prozesse lassen sich auf „ausgewählte Objekte“ beschränken 

**Layer > Eigenschaften > Allgemein > Abfrageerstellung**  
**Layer > Filter**

**Alle Objekte auf die die Abfrage nicht zutrifft werden ausgefiltert**  
Die Abfragezeile wird auch nach Verlassen des Dialogs beibehalten, kann in mehreren Schritten editiert werden und muss manuell gelöscht werden, um die Filterung aufzuheben. Es handelt sich nicht um eine Auswahl, die über 'Ansicht > Auswählen aufheben' aufgehoben werden kann .

---

Mit Hilfe des Abfrageeditors lassen sich komplexe Abfragen in SQL-ähnlicher Syntax erstellen.  
Dennoch lohnt sich der Blick auf die Seite  
<http://www.sql-und-xml.de/sql-tutorial/where-auswahl-von-daten-zeilen.html>

Abb. 42: Abfrage-Editor

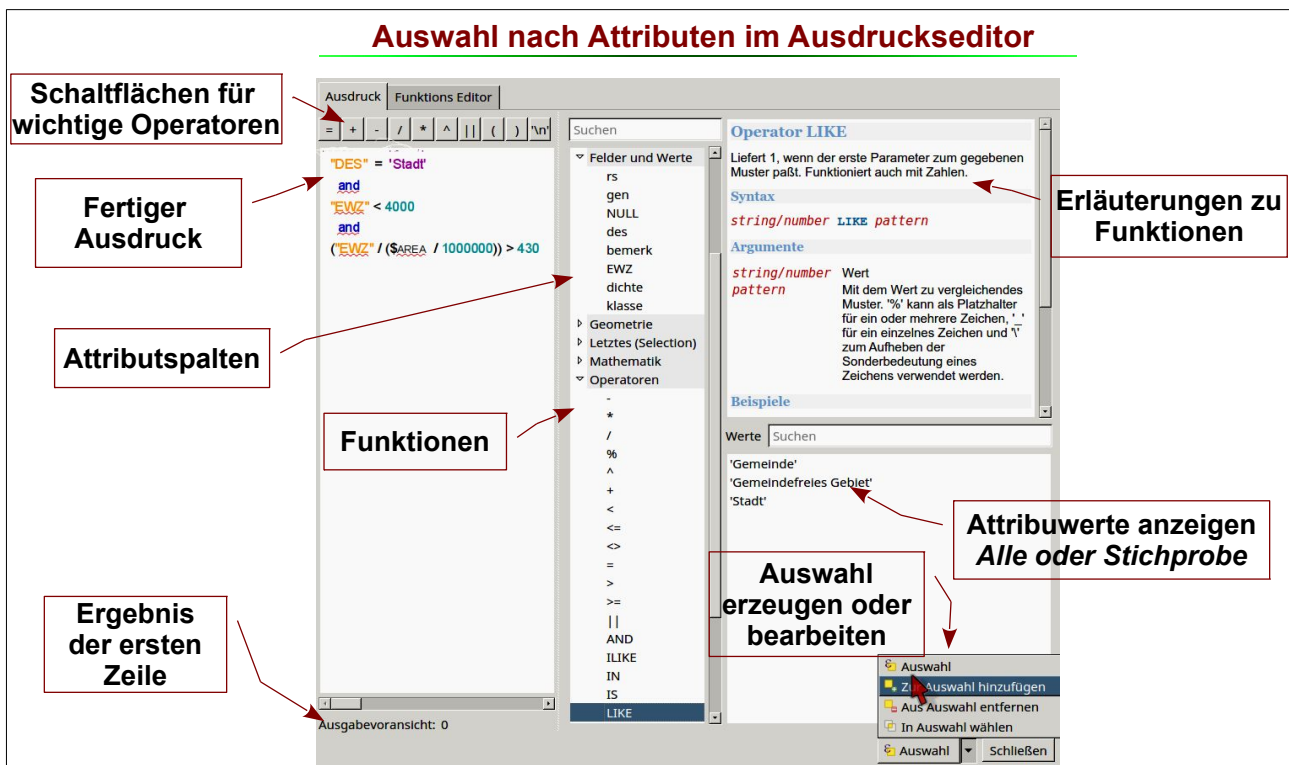


Abb. 43: Aufbau Abfrageeditor

Die Funktionen des Ausdruckseditors sind nach Kategorien geordnet, wobei sich die Basis-Abfragewerkzeuge unter „OPERATOREN“ befinden. Unter „FELDER UND WERTE“ haben Sie Zugriff auf die Spalten der Attributtabelle. Unter „LETZTES“ findet sich ein Archiv der letzten verwendeten Ausdrücke.

Zu jeder Funktion gibt es gute deutschsprachige Erläuterungen.

In der folgenden Abbildung finden Sie Syntax der wichtigsten Grundfunktionen.

## Operatoren und Syntax-Regeln

### Grundsätzlich

Textwerte sind in Hochkommas (') einzurahmen / Abfragen lassen sich mit **Klammern ()** strukturieren / Als **Platzhalter** gelten **\_** für ein beliebiges Zeichen und **%** für eine unbestimmte Anzahl beliebiger Zeichen

### Größenvergleiche

Die Operatoren **=**, **>**, **<**, **<=**, **>=**, **<>** (ungleich) sind selbsterklärend

**Bsp:** `Nutz = 'Wald'` (Alle Objekte, für die als Attribut der Spalte *Nutz* der Wert *'Wald'* zu finden ist)

**Bsp:** `FL_ha > 12` (Alle Objekte für die Attribut der Spalte *FL\_ha* ein Wert größer als 12 zu finden ist)

### Vergleich auf Ähnlichkeiten

Mit dem Operator **LIKE** Ähnlichkeiten finden

**Bsp:** `NAME LIKE '%Fulda%'` (Findet alle Objekte mit *'Fulda'* im Namen, z.B. Fulda, Fuldabrück, Rothenburg a. d. Fulda) (ILIKE ohne Berücksichtigung von groß- und Kleinschreibung)

### Suche über Aufzählungen

Mit dem Operator **IN** gesuchte Werte aufzählen

**Bsp:** `OBJEKTART IN ('Wohnbaufläche', 'Gewerbefläche', 'Fläche gemischter Nutzung')`

Findet alle Objekte, bei denen einer der aufgezählten Werte in der genannten Spalte vorkommt.

### Fragen verknüpfen

Mit dem Operator **OR**, **AND** und **NOT** Abfragen über verschiedene Attribute verknüpfen

**Bsp:** `GEMNR >= 611000 AND NOT N_KREIS = 'Landkreis Fulda' AND (EINWO_ZA > 10000 OR EinwQKM > 303)`

Findet alle Gemeinden mit einer höheren Gemeindenummer als 611000 (RP Kassel), die gleichzeitig nicht zum Landkreis Fulda gehören und entweder mehr als 10000 Einwohner oder über eine höhere Bevdichte als 303 Einw je Qkm verfügen.

### NoData\_Felder finden-Werte finden mit IS NULL

**Bsp:** `GEMNR IS NULL`

Abb. 44: Die wichtigsten Operatoren im Ausdruckseditor

## 6.2 Beispiele

### EINDEUTIGE WERTE FINDEN

Mit dieser Syntax können Sie eine ganze Reihe von Suchaufgaben im Abfrageditor umsetzen. Nehmen Sie den Beispieldatensatz GEMEINDEN und suchen Sie die Stadt SCHNEVERDINGEN.

In der Spalte GEN sind die Namen der Gemeinden verzeichnet. Mit

```
x "gen" = 'Schneverdingen'
```

findet man SCHNEVERDINGEN, jedoch nur wenn SCHNEVERDINGEN genau richtig geschrieben worden ist.

Die Attributspalte wird in Anführungszeichen "..." gesetzt. Auch wenn eine Abfolge von Buchstaben ohne Kennzeichnung erfolgt z.B gen wird dies als Attributspalte interpretiert. Eine Zeichenkette muss immer in Hochkomma ' ' stehen.

Innerhalb der Zeichenkette sind Leerzeichen Zeichen, so dass auch nach Leerzeichen gesucht werden kann.

Der Ausdruck "gen" = Schneverdingen

bedeutet: Wähle sämtliche Objekte, in denen die Spalten gen und die Spalte Schneverdingen den gleichen Wert haben

```
x "gen" = 'Schneverdingen'
```

bedeutet: Wähle sämtliche Objekte, in denen in der Spalten gen die Zeichenkette Schneverdingen steht.

### ÄHNLICHKEIT

Was aber tun, wenn der genaue Inhalt einer gesuchten Zeichenkette nicht bekannt ist. Sie suchen z.B. Rotenburg an der Wümme. Nur Sie sind sich nicht sicher, ob Rotenburg nicht vielleicht Rothenburg heist. Und was steht genau in der Spalte? Rotenburg a.d. Wümme oder Rotenburg (Wümme) oder was auch immer.

Die Funktion **like** hilft hier weiter. Mit **like** lassen sich Platzhalter in der Suche verwenden. \_ steht für ein beliebiges Zeichen und % steht für eine beliebige Anzahl beliebiger Zeichen.

```
x "gen" like 'Rot%burg%'
```

findet sowohl ROTENBURG als auch ROTHENBURG unabhängig davon, was noch nach dem BURG steht.

Wenn die Groß- und Kleinschreibung nicht berücksichtigt werden soll, kann statt **like** der operator **ilike** verwendet werden. So lassen sich z.B.sämtliche Gemeinden finden, die ein BURG enthalten, also Burgdorf genauso ebenso wie Friedeburg:

```
x "gen" ilike '%burg%'
```

Sie können Abfragen mit **and**, **or**, und **not** verknüpfen

x **Auf beiden Seiten dieser Operatoren muss jeweils eine vollständige Abfrage stehen!**

In der Spalte des des Gemeindelayers sind die Kategorien Stadt, Gemeinde, gemeindenfreies Gebiet aufgeführt. Suchen wir nach allen Städten, die gleichzeitig Bad sind, so lässt sich dies mit dieser Abfrage umsetzen:

```
x "gen" like 'Bad%' and "des" = 'Stadt'
```

### VERGLEICHE MIT ZHALENWERTEN

Interessanter wird es, wenn Zahlenwerte und Geometrien ins Spiel kommen. Laden Sie die Tabelle EWZ.XLSX ins Projekt. Die Tabelle besitzt eine Spalte mit dem Regionalschlüssel RS, welchen jede Gemeinde eindeutig kennzeichnet. Dieser Regionalschlüssel findet sich auch im Gemeinde-Layer, so dass die Tabelle über dieses GEMEINSAME SCHLÜSSELFELD angejoint werden kann. (Siehe Kapitel 5 S. 34)

Suchen Sie **sämtliche Gemeinden die Stadt sind und über weniger 3000 Einwohner verfügen**  
(Leerzeichen und Umbrüche zwischen den Operatoren, Spalten und Werten haben keine Bedeutung).

"des" = 'Stadt'

and

"EWZ" < 3000

### GEOMETRIEWERTE EINBEZIEHEN

In der Rubrik GEOMETRIE finden sich die Grundfunktionen zur Berechnung von Flächen, Strecken, Umfang und Koordinaten, nebst weiteren Funktionen. Ausgabereinheit ist die Einheit des Koordinatenbezugssystems, bei ETRS89 / UTM32 und GK 3 ist dies der Meter.

x \$area Fläche

x \$length Länge einer Strecke

x \$perimeter Umfang einer Fläche

x \$x und \$y Koordinaten eines Punktes, des Geometriemittelpunktes

**Jetzt können wir folgendes herausfinden:**

**Finde sämtliche Städte mit weniger als 4000 Einwohner, die jedoch über eine höhere Bevölkerungsdichte als 430 EW je qkm verfügen.**

x Mit des = Stadt und "EWZ" < 3000 haben wir zwei Wert, die wir benötigen, die Einwohnerdichte steht jedoch in keiner Spalte.

x des = Stadt und "EWZ" < 3000 and dichte > 430 kämen wir zum Ergebnis, wenn es die Spalte dichte mit den Einwohner je qkm gäbe.

die Dichte wäre "EWZ" / qkm

\$AREA gibt Quadratmeter aus, mit \$area / 1000000 bekommen wir qkm

somit können wird die Einwohnerdichte mit diesem Ausdruck berechnen

("EWZ" / (\$area / 1000000)) > 430

Die Lösung der Fragestellung sieht folgendermaßen aus:

"DES" = 'Stadt'

and

"EWZ" < 4000

and

("EWZ" / (\$AREA / 1000000)) > 430

Bitte denken Sie daran sinnvolle Klammern zu setzen.

### REGELBASIERTE DARSTELLUNG IM LAYER

In Layer DLM250 ist eine regelbasierte Darstellung über mehrere Attributspalten und teilweise mit mehreren Attributen mit Hilfe von Abfragen umgesetzt.

Regelbasierend	
Beschriftung	Regel
<input checked="" type="checkbox"/> Wald	"Veget" = 'AX_Wald'
<input checked="" type="checkbox"/> Siedlung	"Siedlung" in ('AX_FlaecheBesondererFunktionalerPraegung', 'AX_Industrie...
<input checked="" type="checkbox"/> Wasser	"Wasser" IS NOT NULL
<input checked="" type="checkbox"/> Heide	"Veget" = 'AX_Heide'

**Abb. 45: Regelbasierte Darstellung einfach**

Der Bereich Siedlung wird über folgende Abfrage dargestellt:

"Siedlung" in ('AX\_FlaecheBesondererFunktionalerPraegung', 'AX\_IndustrieUndGewerbeflaeche', 'AX\_Siedlungsflaeche')

Mit in lassen sich mehrere exakte Werte einer Spalte auswählen, ohne das eine Vielzahl von Abfragen mit OR zu verknüpfen ist.

## NEUE SPALTEN BERECHNEN MIT DEM FELDRECHNER ÜBER BEDINGTE ABFRAGEN MIT CASE

Ein weiteres Erscheinungsbild des AUSDRUCKSEEDITORS ist der FELDRECHNER, der vorhandene und neue Spalten berechnet. Aufrufbar aus der ATTRIBUTTABELLE oder im FELDER-TAB der LAYER-EIGENSCHAFTEN. Der Layer muss über den gelben Stift bearbeitbar geschaltet werden (BEARBEITUNGSSTATUS UMSCHALTEN) – Speichern durch erneutes Umschalten mit dem Stift.



So lässt sich z.B. die zu Ganzzahl gerundete Einwohnerdichte berechnen:

`to_int("EWZ" / ($AREA / 1000000))`      Zu Ganzzahl umwandeln mit `to_int()`

`round(($AREA / 1000000),2)`      Quadtrakilometer berechnen auf auf zwei Stellen runden

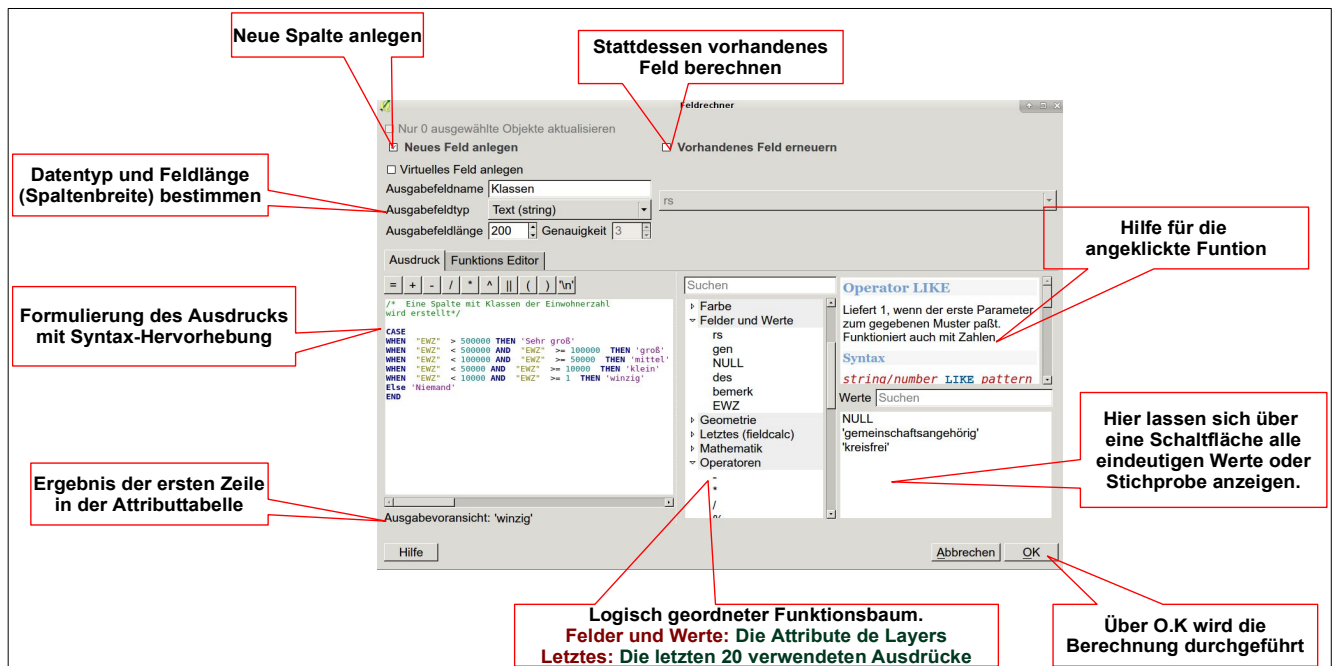


Abb. 46: Feldrechner Überblick

Neben einfachen Berechnungen wir z.B. `"EWZ" / ($AREA / 1000000)`, um die Einwohnerdichte zu berechnen oder einfachen Textfunktionen wie z.B. `"PLZ" || "ORT"` um, den Inhalt einer Spalte mit der Postleitzahl mit dem Inhalt der Orts-Namen-Spalte zu verketteten ( `||` ), können komplexe bedingte und verschachtelte Ausdrücke formuliert werden.

Insbesondere Bedingungen mit der **Case**-Abfrage sind sehr nützlich. Die Funktion findet sich im Ausdruckseditor unter „BEDINGUNGEN.“

**CASE** (Anweisung beginnt)  
**When** (Es folgt die erst Bedingung)  
 SQL-Bedingung wie Abfrageeditor  
**then** (Es folgt der Wert oder der Inhalt einer Spalte oder eine Berechnung)  
 Ein Text- oder Zahlenwert, eine Spalte oder eine Berechnung  
**When....** (weitere Bedingungen)  
**then....** (Weitere Anweisungen)  
**Else** (alle übrigen Felder)  
**END** (Anweisung zu Ende)

Abb. 47: Bedingungen mit Case

Hinter **when** und hinter **then** ist jeder Ausdruck möglich, den die Syntax des Ausdruckseditors zulässt. Soll im Layer GEMEINDEN eine neue Spalte angelegt werden, in welcher **Stadt** dem Städtenamen vorausgestellt wird, ist folgendermaßen vorzugehen:

## CASE

```
when "DES" = 'Stadt' then "DES" || ' ' || "GEN"
when "DES" = 'Gemeindefreies Gebiet' then 'Jagdrevier' || ' ' || "GEN"
Else "GEN"
END
```

|| verknüpft zwei Zeichenketten wenn in der Spalte "DES" die Zeichenkette 'Stadt' steht und in der Spalte "GEN" die Zeichenkette 'Schneeverdingen', werden beide über || verknüpft. Zwischen beiden || steht ein Leerzeichen in Hochkomma || ' ' ||. also steht in der resultierenden Spalte 'Stadt Schneeverdingen'

Soll eine neue Attributspalte angelegt werden, bei der für die Gemeinden, Klassen der Einwohnerzahl entsprechend folgende Kriterien angelegt werden,

Sehr groß:	500000
groß:	100000
mittel:	50000
klein	10000
winzig:	1000

ist folgendermaßen vorzugehen:

--Spalte im Textformat anlegen, das ein Textwert eingetragen wird.

## CASE

```
when "EWZ" >= 500000 then 'sehr groß'
when "EWZ" < 500000 and "EWZ" >= 100000 then 'groß'
when "EWZ" < 100000 and "EWZ" >= 50000 then 'mittel'
when "EWZ" < 50000 and "EWZ" >= 10000 then 'klein'
when "EWZ" < 10000 and "EWZ" >= 1 then 'winzig'
else 'niemand'
END
```

## KREISIDE MIT SUBSTR() ERZEUGEN

Aus den Gemeinden sollen Landkreis-Geometrien erzeugt werden. Die Tabelle **lkr.csv** enthält die Landkreisnamen und eine Kreis-ID (krid), deren Zeichenfolge im Regionalschlüssel des Gemeindelayers enthalten ist: Mit der krid **3151** korrespondiert der Regionalschlüssel **'031510040040'**. Somit entspricht die Stelle 2-5 des Regionalschlüssels der Krid.

Um die 3. bis 5. Stelle zu extrahieren, kann die Funktion **substr("SPALTE",starstelle, Länge)** verwendet werden: **substr("rs", 2, 4)** ergibt **'3151'**: Über **to\_int()** kann der Wert in eine Ganzzahl verwandelt werden.

```
to_int(
  substr("rs", 2, 4)
)
```

ergibt: **3151**

Jetzt kann die Kreis-Namen-Tabelle angejoint werden. Anschließend sind in der Attributtabelle für jede Gemeinde die korrespondierenden Kreis-ID und Landkreis-Namen zu sehen. Mit Hilfe der Kreis-ID können die Gemeindegeometrien über ein Dissolve-Werkzeug zu landkreis-Geometrien verschmolzen werden.