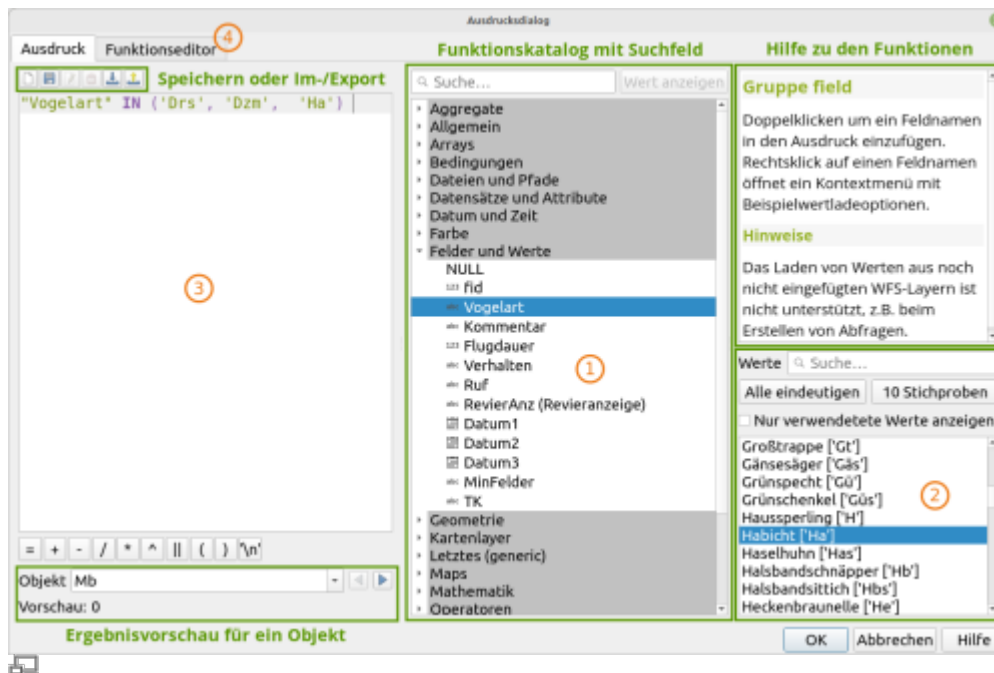


QGIS Ausdrücke (Expressions)

Ausdrücke (Expressions oder Regeln) sind überaus bedeutend, denn sie kommen in vielen Bereichen von QGIS vor: beim **Styling**, bei **Abfragen**, **Feldberechnungen**, zum **Filtern** und **Auswählen** und bei der **Erstellung von Karten oder Kartenserien**. Schenken wir diesem Thema also besondere Aufmerksamkeit.

Ausdrücke und Ausdrucksdialog



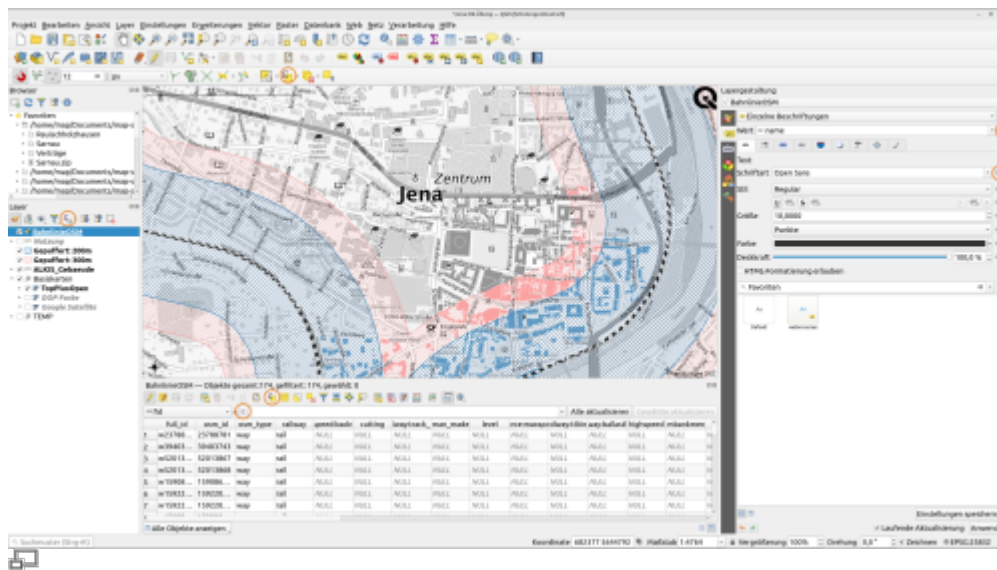
Der Ausdruckseditor mit einfachem Ausdruck (QGIS 3.24)

Ausdrücke sind im übertragenen Sinne **Formeln**, welche in einer QGIS-eigenen, **SQL-ähnlichen Syntax** geschrieben sind und in mehrere Ebenen verschachtelt sein können. Mit ihnen werden Bedingungen formuliert sowie damit verbundene Anweisungen, die bei Erfüllung oder Nichterfüllung der Bedingung eintreten. Wir können Ausdrücke **fortlaufend** schreiben oder mit Hilfe des **Ausdruckseditors**  per Mausklick erstellen. Es steht uns eine große Anzahl an **Funktionen** zur Verfügung, welche mit einer Suchmaske gefiltert werden können. Für fortgeschrittene Anwender besteht die Möglichkeit über den Funktionseditor, eigene **Python-Skripte** anstelle der QGIS-Ausdrücke zu verwenden.

1. Auswahl der in der Attributtabelle vorhanden Felder
2. ...und Werte
3. Funktionsfeld - hier wird die Funktion eingetrippt. Die Syntax wird hervorgehoben und Fehler mit einem roten Dreieck angezeigt
4. Fortgeschrittene Anwender:innen können auch die Python-Syntax verwenden

[Ausdrücke in der QGIS 3.22 Dokumentation](#)

...und wo finde ich den Ausdrucksdialog?



Such das Epsilon! (QGIS 3.24)

Überall in QGIS lassen sich mit Ausdrücken Dinge steuern: Beschriftungen, Symbolisierung, Berechnungen, Geometrien, Kartendarstellung etc... Betrachten wir die QGIS Oberfläche, finden wir auf Anhieb mehrere Stellen mit dem **Epsilon-Symbol** ⋈. Und dahinter verbirgt sich der **Ausdruckseditor oder Ausdrucksdialog**.

Aber **nicht nur da**, sondern auch an vielen weiteren Stellen. Zum Beispiel bei der *Datendefinierten Übersteuerung* mit diesem Symbol: ⋈ - finden Sie es?

Übersteuerung mit Ausdrücken, Variablen oder Feldwerten

Datendefinierte Übersteuerung

Beschreibung...

☐ Daten im Projekt speichern

Attributfeld

☐ Feldtype: Ganze Zahl, double, Zeichenkette >

Ausdruck

☐ Variable >

Bearbeiten...

Einfügen

Assistent...

Kontextmenü der Datendefinierten Übersteuerung

(Fast) überall da, wo Einstellungen am Stil, der Beschriftung oder in der Druckzusammenstellung vorgenommen werden können, befindet sich dieses kleine unscheinbare Symbol: ⋈ Es steht für **daten- oder regelbedingte Übersteuerung** und ermöglicht uns die statischen *per Hand* eingegebenen Werte durch **dynamische** zu ersetzen. Diese stammen aus

- **Regeln (Ausdrücke)** (siehe oben)
- **Variablen** (können in den Projekteigenschaften frei vergeben werden)
- **Feldwerte** (stammen aus der Attributtabelle des Layers) oder
- **Hilfsspeicher** (geometrielose Tabelle, welche mit der Projektdatei gespeichert wird, und als Datenspeicher dient)

Diese Funktion ist zugegeben etwas fortgeschritten, expandiert aber die Möglichkeiten der Visualisierung und Beschriftung ganz erheblich. Außerdem sind dynamische Werte Abhängig von den (Layer-)Daten und nicht von einer Nutzereingabe, wodurch sich beispielsweise gut mit Templates arbeiten lässt: eine Projektvorlage mit Datendefinierten Einstellungen, wird sich den Daten anpassen. Ein manuelles Nacharbeiten (bei dem oft Dinge übersehen oder falsch eingegeben werden) ist nicht notwendig!

mehr dazu (in Englisch) im [QGIS Handbuch](#)

Häufige Ausdrücke an praktischen Beispielen



Attribute (Spalte in der Attributtabelle) werden immer in **doppelte Anführungszeichen** gesetzt: „ATTRIBUT“; **Werte** (Zelle der Attributtabelle) in **einfachen Anführungszeichen**: 'WERT'.



Groß- und Kleinschreibung, Zeilenumbrüche oder aufeinanderfolgende Leerzeichen spielen bei der QGIS-SQL-Syntax **keine Rolle** und können verwendet werden

Einfache Ausdrücke mit häufigen Operatoren

Ausdruck	Syntax	Ergebnis	Erläuterung
"Vogelart" = 'Mäusebussard' OR "Vogelart" = 'Rotmilan'	a OR b	Alle Mäusebussarde und alle Rotmilane werden angesprochen	Abfrage an einem Attribut mit zwei Möglichkeiten
"Vogelart" IN ('Mäusebussard', 'Rotmilan')	a AND b	Alle Mäusebussarde und alle Rotmilane werden angesprochen	Elegantere Alternative zu oben
"Vogelart" = 'Mäusebussard' AND "Verhalten" = 'kreisend'	a = b	Alle kreisenden Mäusebussarde werden angesprochen	Abfrage an zwei Attributen . Zwei Bedingungen müssen erfüllt sein
"Vogelart" = 'Mäusebussard' AND "Verhalten" != 'kreisend'	a != b	Alle nicht-kreisenden Mäusebussarde werden angesprochen	Abfrage an zwei Attributen . Zwei Bedingungen müssen erfüllt sein

Einfache Ausdrücke mit Zahlenwerten

Ausdruck	Syntax	Ergebnis	Erläuterung
"Vogelart" = 'Rotmilan' AND "Hoehe" > 500	$a <> b$	Alle Rotmilane, welche höher als 500m fliegen werden angesprochen	Alle bekannten mathematischen Zeichen können bei Zahlenwerten eingesetzt werden
"Vogelart" = 'Rotmilan' AND "Hoehe" > 100 AND "Hoehe" <= 500	$a <= b$	Alle Rotmilane in einer Höhe zw. 100m und 501m werden angesprochen	...natürlich auch in Kombination
"Vogelart" = 'Rotmilan' AND "Hoehe" > 100 AND "Hoehe" <= 500	$a <= b$	Alle Rotmilane in einer Höhe zw. 100m und 501m	...natürlich auch in Kombination
round("Höhe", 2)	round(VALUE[, places=0])	Die Höhe wird auf 2 Nachkommastellen gerundet	Rundungs-Funktion mit Angabe der Nachkommastellen
rand(1, 20)	rand(MIN, MAX[, seed=NULL])	Ein zufälliger Zahlenwert zwischen 1 und 20 wird ausgegeben	Zufalls-Funktion mit Angabe eines Zahlenbereichs

sonst. Ausdrücke

Ausdruck	Syntax	Ergebnis	Erläuterung
"Vogelart" IS NOT NULL	$a \text{ IS NOT } b$	Alle Zeilen, in welchen bei der „Vogelart“ etwas drin steht	Das Spiel mit den Leerwerten: 'NULL' steht stellvertretend für <i>nichts</i>
"Vogelart" IS NULL	$a \text{ IS } b$	Alle Zeilen, in welchen bei der „Vogelart“ nichts drin steht	Das Spiel mit den Leerwerten: 'NULL' steht stellvertretend für <i>nichts</i>
COALESCE("Vogelart_A", "Vogelart_B")	COALESCE(expression1, expression2...)	Wenn in Vogelart_A nichts steht, aber in Vogelart_B so wird Vogelart_B verwendet	Liefert den ersten nicht-NULL-Wert der Ausdrucksliste.

Geometrie-Ausdrücke

Ausdruck	Syntax	Ergebnis	Erläuterung
\$length	\$length	Gibt die Länge einer Linie aus	Gilt nur für Linienobjekte . Es werden die aktuellen Ellipsoid- und Abstandseinheiteneinstellung berücksichtigt
\$area	\$area	Gibt die Fläche eines Polygons aus	Gilt nur für Polygonobjekte . Es werden die aktuellen Ellipsoid- und Abstandseinheiteneinstellung berücksichtigt
\$perimeter	\$perimeter	Gibt den Umfang eines Polygons aus	Gilt nur für Polygonobjekte . Es werden die aktuellen Ellipsoid- und Abstandseinheiteneinstellung berücksichtigt
\$x	\$x	x-Koordinate eines Punktes	Gilt nur für Punktobjekte .
\$y	\$y	y-Koordinate eines Punktes	Gilt nur für Punktobjekte .

Die QGIS Wenn-Dann-Sonst-Fälle

Ausdruck	Syntax	Ergebnis	Erläuterung
IF("Hoehe" > 500, 'hoch', 'niedrig')	IF(condition, result_when_true, result_when_false)	Ab einer Höhe von über 500 schreibe 'hoch' ansonsten schreibe 'niedrig'	Wenn-Dann-Sonst -Syntax. Kann auch in mehreren Zeilen geschrieben werden
CASE WHEN "Hoehe" > 300 THEN 'hoch' ELSE 'niedrig' END	CASE WHEN bedingung THEN ergebnis [...n] [ELSE ergebnis] END	Ab einer Höhe von über 500 schreibe 'hoch' ansonsten schreibe 'niedrig'	Wie oben, mit dem entscheidenden Unterschied, dass beliebig viele „WHEN“-Fälle kombiniert werden können s.U.
CASE WHEN "Hoehe" > 300 AND "Hoehe" < 500 THEN 'hoch' WHEN "Hoehe" > 500 THEN 'sehr hoch' ELSE 'niedrig' END	CASE WHEN bedingung THEN ergebnis [...n] [ELSE ergebnis] END	Bei einer Höhe von 300-500 schreibe 'hoch', bei über 500 schreibe 'sehr hoch', in allen anderen Fällen schreibe 'niedrig'	Beliebig häufig kann das Spiel mit WHEN-THEN wiederholt werden.

-  Ausdrücke: zu vervollständigen! (zugewiesen an map)
-  Ausdrucksdialog Screenshot erneuern (zugewiesen an map)
-  Sammlung an Beispielen an Übungsdatensatz (zugewiesen an map)

From:
<https://lernplattform.map-site.de/> - **Lernplattform für OpenSource GIS**

Permanent link:
https://lernplattform.map-site.de/doku.php/qgis/basic/d_layer_und_daten/cl06_qgis_ausdruecke_expression

Last update: **2023/05/23 15:27**

