

Kurzanleitung D-GRAM

D-GRAM ist ein browserbasiertes Tool, mit dem du eigene digitale Grammatiken erstellen kannst. Das Tool läuft direkt im Browser unter **bubenhofer.com/dilaits** (Reiter **D-Gram**). Es besteht aus zwei Eingabefeldern: **Rules** (Syntaxregeln) und **Vocabulary** (Lexikon). Mit diesen kannst du korrekte Sätze generieren, Sätze auf Korrektheit prüfen, Syntaxbäume zeichnen und digitale Poesie erstellen.

1 Syntaxregeln

Syntaxregeln beschreiben, wie sich sprachliche Einheiten zusammensetzen. In D-GRAM haben sie immer dieselbe Form:

```
ÜbergeordnetesElement -> UntergeordnetesElement1 UntergeordnetesElement2
```

Links vom Pfeil steht genau *ein* Element (das übergeordnete); rechts stehen ein oder mehrere Elemente, getrennt durch Leerzeichen. Der Pfeil wird aus -> (Minus + Grösser-als-Zeichen) gebildet.

Beispiele:

```
S -> SUBJ Verb OBJ
SUBJ -> Artikel Nomen
OBJ -> Artikel Nomen
```

Alle Namen kannst du frei wählen – aber schreib sie immer gleich! D-GRAM unterscheidet zwischen Gross- und Kleinschreibung.

Konvention (nicht Pflicht): Satzglieder und Phrasen in GROSSBUCHSTABEN, Wortarten mit Grossbuchstaben am Anfang:

```
NP -> Artikel Nomen      (Nominalphrase)
VP -> Verb NP             (Verbalphrase)
```

2 Lexikon (Vocabulary)

Das Lexikon weist Wortarten konkrete Wörter zu. Die Syntax ist dieselbe wie bei Regeln – nur stehen rechts Wörter in Anführungszeichen:

```
Nomen -> "Hund" | "Katze" | "Vogel"
Artikel -> "der" | "die" | "das"
Verb -> "sieht" | "jagt"
```

Mit dem senkrechten Strich | (Pipe, auf der Tastatur: AltGr + 7 oder ~ + 7) kannst du mehrere Wörter auf eine Zeile schreiben. Wörter setzt du am besten in *doppelte* Anführungszeichen – damit gibt es keine Verwechslungsprobleme mit Apostrophen (z.B. "l'eau", "Peter's").

3 Kongruenz und Eigenschaften

Wenn du sicherstellen möchtest, dass z.B. Artikel und Nomen im gleichen Kasus oder Numerus stehen, kannst du Elementen Eigenschaften in eckigen Klammern zuweisen:

```
NP[kasus=?k, numerus=?n] -> Artikel[kasus=?k, numerus=?n] Nomen[kasus=?k, numerus=?n]
```

Eigenschaften werden innerhalb der Klammer mit Kommas getrennt. Das **Fragezeichen + Buchstabe** (z.B. **?k**, **?n**) ist eine **Wildcard**: Zwei Elemente mit derselben Wildcard müssen denselben Wert haben – egal welchen. So erzwingt **?n** bei Artikel und Nomen denselben Numerus (beide Singular oder beide Plural), ohne festzulegen, welcher es sein muss.

Im Lexikon gibst du den Wörtern die passenden Eigenschaften mit:

```
Artikel[kasus=nom, numerus=sg] -> "der" | "die" | "das"
Artikel[kasus=nom, numerus=pl] -> "die"
Nomen[numerus=sg] -> "Hund" | "Katze"
Nomen[numerus=pl] -> "Hunde" | "Katzen"
```

Es gibt drei verschiedene Trennzeichen in D-GRAM: Leerzeichen (zwischen Elementen in Regeln), Pipe | (zwischen Varianten im Lexikon), Komma (zwischen Eigenschaften in eckigen Klammern).

4 Grenzen von D-GRAM

D-GRAM ist bewusst einfach gehalten. Folgende Einschränkungen solltest du kennen:

- Keine automatische Grossschreibung am Satzanfang – was du im Lexikon schreibst, erscheint 1:1 in der Ausgabe.
- Keine automatische Interpunktion – Satzzeichen müssen als eigene Elemente in der Grammatik modelliert werden.
- Zwischen jedem Element wird automatisch ein Leerzeichen gesetzt – auch vor Satzzeichen oder Affixen. D-GRAM gibt also z.B. **"Peter 's car ."** aus, nicht **"Peter's car."**

Challenges

Challenge 1: Erste Schritte ★☆☆

Öffne D-GRAM unter bubenhofer.com/dilais. Gib folgende Mini-Grammatik im Feld Rules ein und das Lexikon im Feld Vocabulary. Drücke dann auf «Update Grammar» und probiere alle Funktionen (Sätze generieren, prüfen, Syntaxbaum) aus.

```
Rules:
S -> SUBJ Verb
SUBJ -> Artikel Nomen

Vocabulary:
Artikel -> "der" | "die" | "das"
Nomen -> "Hund" | "Katze" | "Vogel"
Verb -> "schläft" | "rennt" | "fliegt"
```

Beantworte: Welche Sätze erzeugt die Grammatik? Sind sie alle grammatisch korrekt? Welche fehlen oder wirken seltsam?

Challenge 2: Grammatik erweitern ★☆☆

Erweitere die Grammatik aus Challenge 1, sodass auch Sätze mit einem Objekt möglich sind (z.B. "der Hund jagt die Katze"). Füge dazu ein OBJ-Element ein und ergänze das Lexikon mit passenden Verben.

Tipp: Beginne damit, die Regel **S -> SUBJ Verb** zu erweitern.

Bonus: Kannst du verhindern, dass "der Hund jagt der Hund" als korrekt gilt (Kasus-Kongruenz)? → Schau dir Abschnitt 3 an.

Challenge 3: Grammatik für dein Fach ★★☆☆

Erstelle eine Grammatik für die Sprache, die du unterrichtest oder lernst (Deutsch, Englisch, Französisch, Italienisch, ...). Ziel: D-GRAM soll mindestens 10 verschiedene, grammatisch korrekte Sätze erzeugen können. Baue Schritt für Schritt – fang mit 2–3 Regeln und 5–6 Wörtern an.

Halte fest:

- Welche Regeln hast du geschrieben?
- Welche Schwierigkeiten sind aufgetaucht?
- Was konntest du mit D-GRAM (noch) nicht abbilden?

Challenge 4: Ambiguität (Teleskop-Challenge) ★★★

Erstelle eine Grammatik, die die strukturelle Mehrdeutigkeit des folgenden Satzes korrekt abbildet:

Deutsch: die Dame sieht den Herrn mit dem Teleskop
English: the lady sees the gentleman with the telescope
Français: la dame voit le monsieur avec le télescope
Italiano: la signora vede il signore con il telescopio

Der Satz ist strukturell mehrdeutig: Das Teleskop kann dem *Sehen* (VP-Adjunkt) oder dem *Herrn* (NP-Adjunkt) zugeordnet werden. D-GRAM soll beide Lesarten als separate Sätze oder Syntaxbäume ausgeben können.

Tipp: Zwei verschiedene Regeln für S (oder VP) können beide Strukturen erzeugen.

Challenge 5: Digitale Poesie ★★☆☆

Erstelle eine Grammatik, die mit dem «Digital Poet» von D-GRAM überraschende, poetische Sätze erzeugt. Denke dabei weniger an grammatische Korrektheit als an Klang, Rhythmus und unerwartete Bildkombinationen.

Inspirationen:

- Kombiniere belebte und unbelebte Nomen als Subjekte/Objekte.
- Verwende Adjektive oder Adverbien für mehr Variation.
- Lass «semantisch unmögliche» Kombinationen zu (z.B. "das Schweigen tanzt").

Stelle uns dein schönstes Gedicht vor 😊