

Texte einfach formatieren mit Zettlr

Mela Eckenfels

14. Februar 2024

Inhaltsverzeichnis

Markdown und YAML	2
Markdown	2
YAML	2
Snippets	6
Bilder und das Bildverzeichnis	8
Bilder	8
Abbildungsverzeichnis	10
Weiteres	10
Einen Seitenumbruch erzwingen	10
PDF-Einstellungen	11
Die PDF-Export-Einstellungen anpassen	11
Ausgabe mit Inhaltsverzeichnis	11
Ein Schritt weiter: Die Dokumentenklasse scrartcl	12
Fazit oder tl;dr	12

Markdown und YAML

Markdown

Zettlr zuerst einmal ein **Markdown**-Editor (von »Markup«, übersetzt »Textauszeichnung«). Das bedeutet, mit einer überschaubaren Anzahl Markup-Zeichen, weist man Bereiche des Texts als Überschrift aus, als Zitat, setzt sie fett oder bindet Bilder ein.

Die Dokumente sind im Prinzip erst einmal nur Text. Das bedeutet, wenn man bisher nur mit Word gearbeitet hat, muss man etwas umdenken, denn das Aussehen des Endprodukts ist vom Schreibvorgang getrennt. (Persönlich halte ich das für eine gute Sache.)

Typische Markdown-Syntax sieht folgendermaßen aus:

```
# Überschrift 1. Ebene
## Überschrift 2. Ebene
### Überschrift 3. Ebene

* Nicht-nummerierte Liste

1. Nummerierte
2. Liste

**Fett**
_Kursiv_
```

Mehr Markdown-Informationen finden sich hier:

- <https://spec.commonmark.org/0.30/>
- <https://www.markdownguide.org/cheat-sheet/>
- [Meine Markdown-Ressourcen-Sammlung](#)

YAML

Um Markdown-Dokumenten Metadaten hinzuzufügen, wird »Yet Another Markup Language« verwendet, kurz YAML. Ein YAML-Header gibt mir selbst Hinweise darauf, wann ich einen Text erstellt habe, wie er heißt, welche Inhalte er hat (ohne ihn erst noch einmal lesen zu müssen) und enthält andere Angaben, die ich selbst dort sehen möchte.

Beispielsweise sieht der YAML-Block meiner Aufgabe zur Kollokationsanalyse folgendermaßen aus:

```
--- YAML Frontmatter
title: "Kollokationsanalyse und Frequenzanalyse: autistisch"
keywords: [Begriffsgeschichte, Kollokationsanalyse, Frequenzanalyse, DiaCollo, Korpus, Korpora, Bundestagsprotokolle]
subtitle: "Seminar: Begriffsgeschichte 2.0"
subject:
colorlinks: true
urlcolor: MidnightBlue
lang: de-DE
date: 05. Februar 2024
author: Mela Eckenfels
id: 20240205200246
email: "mela@eckenfels.net"
reference-section-title: Verweise
header-includes: |
  \usepackage{fancyhdr}
  \pagestyle{fancy}
  \fancyhead[R]{ }
  \fancyhead[C]{Kollokationsanalyse und Frequenzanalyse}
  \fancyhead[L]{ }
  \fancyfoot[R]{05. Februar 2024}
---
```

Abbildung 1: Der ausgefüllte YAML-Block meines Textes zur Kollokationsanalyse

Der von mir eingetragene Titel wird von Zettlr verwendet, um den Text im Programm anzuzeigen ...

```
--- YAML Frontmatter
title: "Kollokationsanalyse und Frequenzanalyse: autistisch"
keywords: [Begriffsgeschichte, Kollokationsanalyse, Frequenzanalyse, DiaCollo, Korpus, Korpora, Bundestagsprotokolle]
subtitle: "Seminar: Begriffsgeschichte 2.0"
subject:
```



Abbildung 2: Der Titel im YAML-Block, der wird sowohl für von Zettlr intern verwendet, als auch bei der Ausgabe

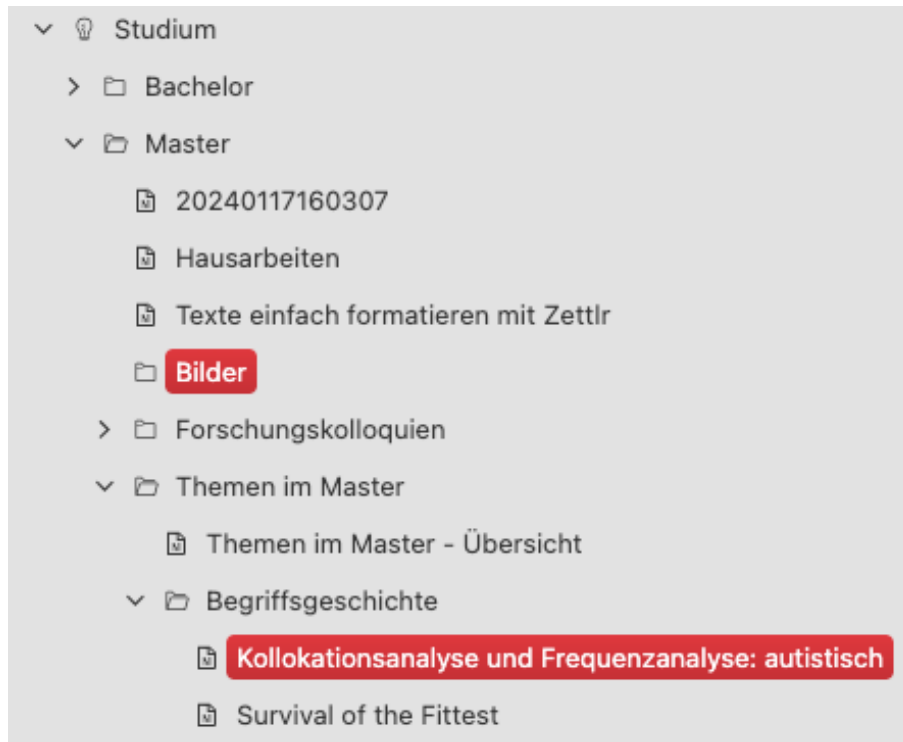


Abbildung 3: Zettlr Seitenleiste mit der Übersicht des Arbeitsbereichs und der enthaltenen Dokumente

... aber bei der Ausgabe als PDF automatisch als Überschrift angezeigt.

Kollokationsanalyse und Frequenzanalyse: autistisch

Seminar: Begriffsgeschichte 2.0

Mela Eckenfels

05. Februar 2024

Abbildung 4: Überschrift, Name und Datum (sowie Untertitel) in der PDF-Ausgabe

Auch Name und Datumsangabe werden aus dem YAML-Block übernommen.

```
---
YAML Frontmatter
title: "Kollokationsanalyse und Frequenzanalyse: autistisch"
keywords: [Begriffsgeschichte, Kollokationsanalyse, Frequenzanalyse, DiaCollo, Korpus, Korpora, Bundestagsprotokolle]
subtitle: "Seminar: Begriffsgeschichte 2.0"
subject:
colorlinks: true
urlcolor: MidnightBlue
lang: de-DE
date: 05. Februar 2024
author: Mela Eckenfels
id: 20240205200246
email: "mela@eckenfels.net"
reference-section-title: Verweise
header-includes: |
  \usepackage{fancyhdr}
  \pagestyle{fancy}
  \fancyhead[R]{ }
  \fancyhead[C]{Kollokationsanalyse und Frequenzanalyse}
  \fancyhead[L]{ }
  \fancyfoot[R]{05. Februar 2024}
---
```

Abbildung 5: Weitere Metadaten im YAML-Block

Der etwas kryptisch wirkende untere Block definiert das Aussehen von Header & Footer:

```
---
YAML Frontmatter
title: "Kollokationsanalyse und Frequenzanalyse: autistisch"
keywords: [Begriffsgeschichte, Kollokationsanalyse, Frequenzanalyse, DiaCollo, Korpus, Korpora, Bundestagsprotokolle]
subtitle: "Seminar: Begriffsgeschichte 2.0"
subject:
colorlinks: true
urlcolor: MidnightBlue
lang: de-DE
date: 05. Februar 2024
author: Mela Eckenfels
id: 20240205200246
email: "mela@eckenfels.net"
reference-section-title: Verweise
header-includes: |
  \usepackage{fancyhdr}
  \pagestyle{fancy}
  \fancyhead[R]{ }
  \fancyhead[C]{Kollokationsanalyse und Frequenzanalyse}
  \fancyhead[L]{ }
  \fancyfoot[R]{05. Februar 2024}
---
```

Abbildung 6: Header und Footer der PDF-Ausgabe festlegen

Die Variable `\fancyhead[C]` definiert, dass der Titel zentriert gesetzt wird. Hätte ich den Titel stattdessen in `\fancyhead[R]` oder `\fancyhead[L]` eingegeben, wäre er, je nachdem rechts- bzw. linksbündig gesetzt.

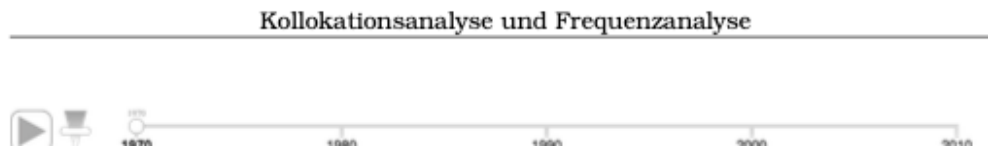


Abbildung 7: So sieht der Header auf Seite 2 in der PDF-Ausgabe aus

`\fancyfoot [R]` setzt das Datum des Dokuments rechtsbündig. Die Seitennummer wird, in diesem Beispiel, nicht über den YAML-Header konfiguriert. Sie ist in der Konfigurationsdatei für die PDF-Ausgabe bereits vorkonfiguriert. Diese Voreinstellungen sind im Normalfall so sinnvoll, dass sie nicht verändert werden müssen.



Abbildung 8: So sieht der Footer in der PDF-Ausgabe aus. In meinem Beispiel wird mittig die Seitenzahl angegeben und rechts davon das Datum.

Solche Bereiche sehen auf den ersten Blick ‚schwierig‘ aus, wenn man gewohnt ist, das Aussehen eines Dokuments in Word zusammenzuklicken. Allerdings plane ich auch hier noch Vorlagen für Studierende der Fakultät zusammenzustellen, die man dann einfach übernehmen kann.

Snippets

Auf den ersten Blick mag der YAML-Block lästig und unhandlich wirken. Man muss diesen nicht aber jedes Mal von Hand tippen, denn Zettlr besitzt eine Funktion, die sich »**Snippets**« nennt.

Dort legt man kurze Schnipsel an, die man innerhalb Zettlr wiederverwendet. Man ruft dieses Schnipselverzeichnis auf, in dem man nur einen Doppelpunkt `:` nach einem Leerzeichen eintippt und wählt den benötigten Schnipsel aus Menü aus:

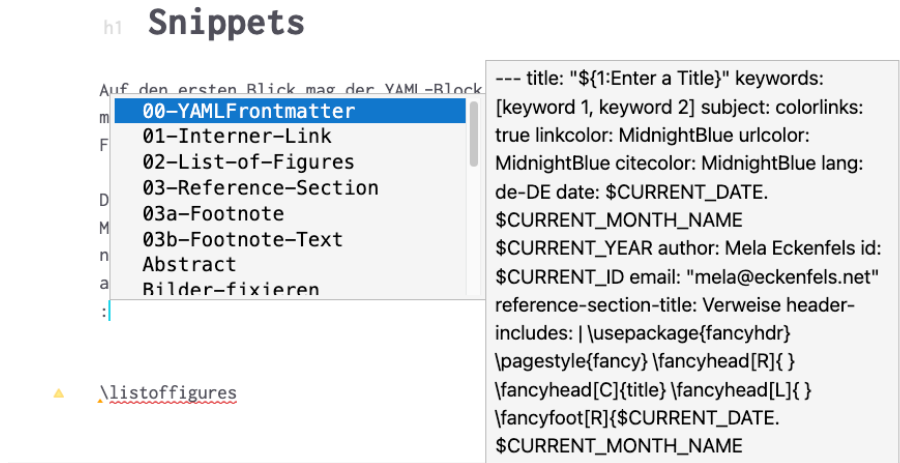


Abbildung 9: Das Snippets-Auswahlmenü

Der Schnipsel für den YAML-Block enthält beispielsweise einige Variablen¹, die beim Einfügen ins Dokument automatisch ergänzt werden:

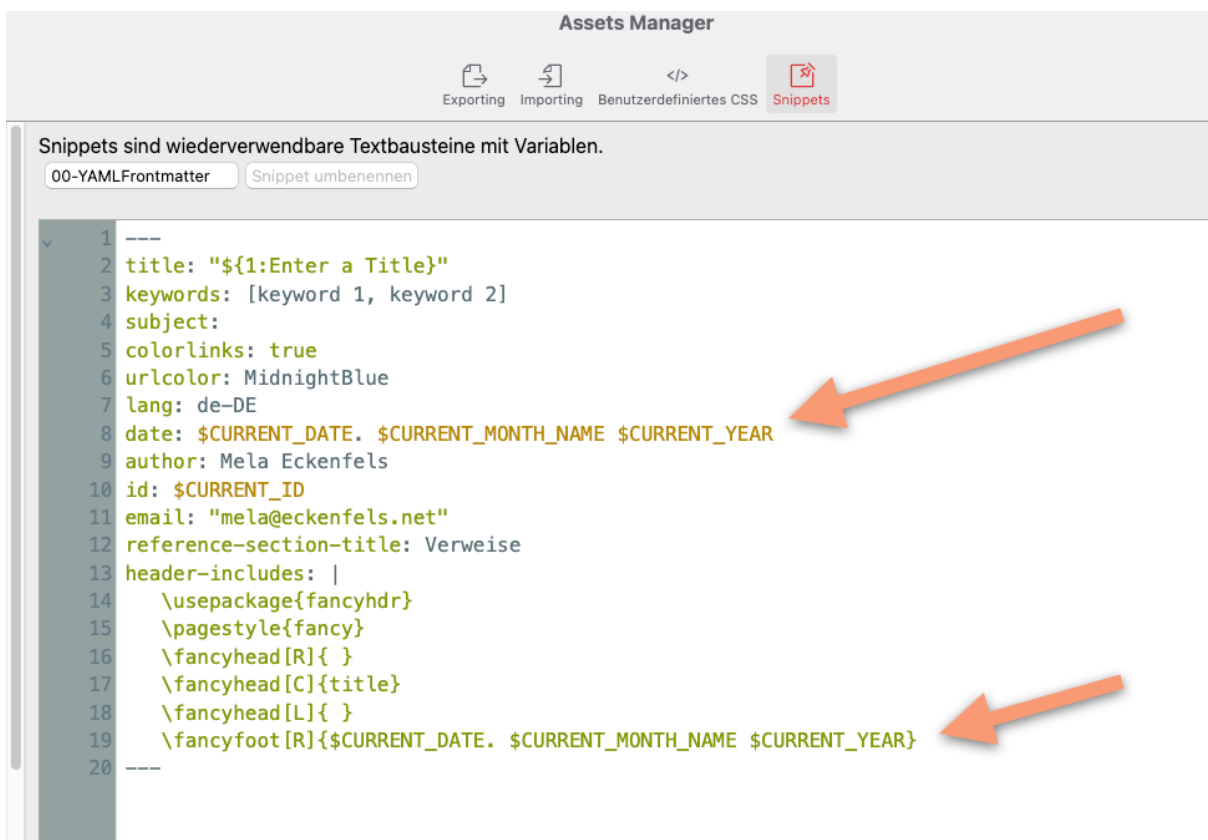


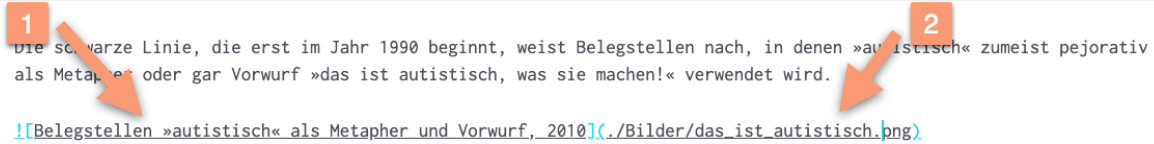
Abbildung 10: Die Variablen im Snippet werden beim Einfügen durch das aktuelle Datum und eine »ID« ersetzt

¹Die »ID« ist an dieser Stelle nicht wichtig. Sie kommt dann ins Spiel, wenn man Zettlr als Wissensmanagementtool, oder auch »Zettelkasten« nach Niklas Luhmann verwendet.

Bilder und das Bildverzeichnis

Bilder

Auch Bilder werden in Markdown-Editoren, so auch Zettlr, über einen kleinen Markup-Befehl repräsentiert und dargestellt. Er besteht aus zwei Teilen.



1 Die schwarze Linie, die erst im Jahr 1990 beginnt, weist Belegstellen nach, in denen »autistisch« zumeist pejorativ als Metapher oder gar Vorwurf »das ist autistisch, was sie machen!« verwendet wird.

2 `![Belegstellen »autistisch« als Metapher und Vorwurf, 2010](./Bilder/das_ist_autistisch.png)`

Abbildung 11: Bildauszeichnung

1. Die Beschreibung des Bildes. Sie wird bei der Ausgabe als PDF zur Bildunterschrift.
2. Der Speicherort des Bildes

Zettlr erstellt diese Zeile automatisch, wenn man ein Bild aus der Zettlr-Seitenleiste in den Editor zieht. Nur bei der Bildbeschreibung muss man dann noch von Hand nacharbeiten.

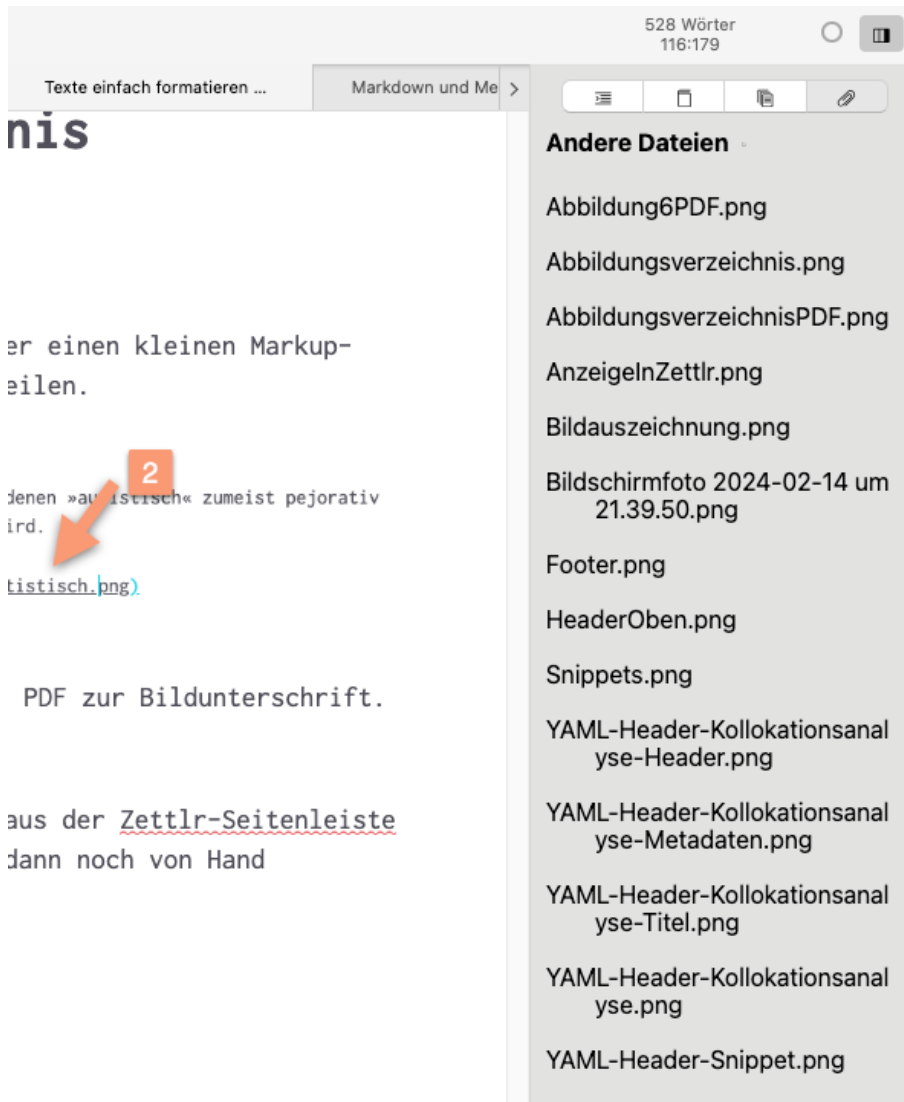


Abbildung 12: Die Bild-Seitenleiste von Zettlr zeigt die im derzeit ausgewählten Bildverzeichnis vorhandenen Dateien an

In der PDF-Ausgabe sieht das Ganze dann so aus:

Die schwarze Linie, die erst im Jahr 1990 beginnt, weist Belegstellen nach, in denen »autistisch« zumeist pejorativ als Metapher oder gar Vorwurf »das ist autistisch, was sie machen!« verwendet wird.

1: [bundestagert:17/17282:24546]	Das ist autistisch , was Sie machen!
2: [bundestagert:17/17246:11698]	... Passage der Entschließung zur Antiterrordatei fast autistisch angesichts der massiven Vorgaben aus Karlsruhe, die...

Abbildung 6: Belegstellen »autistisch« als Metapher und Vorwurf, 2010

Abbildung 13: Die im Dokument hinzugefügte Abbildung im PDF. Die Nummerierung der Bilder wird bei der Ausgabe automatisch vorgenommen.

Abbildungsverzeichnis

Das Abbildungsverzeichnis fügt man in Zettlr mit einem einzigen LaTeX-Befehl hinzu: Diesen setzt man an die Stelle des Dokuments, wo das Abbildungsverzeichnis erscheinen soll. Also zumeist am Ende des Dokuments.

Auch diesen kurzen Befehl habe ich mir als Snippet angelegt, damit ich ihn mir nicht merken muss.

`\listoffigures`

Abbildung 14: Der LaTeX-Befehl ‚listoffigures‘ fügt ein Abbildungsverzeichnis an genau der Stelle ein, an der er im Dokument steht. Der Backslash vor dem Begriff ist wichtig.

In der PDF-Ausgabe sieht das Ganze dann wie folgt aus:

Abbildungsverzeichnis

1	Kollokationsanalyse »autistisch« 1970 (Bubble)	2
2	Belegstelle: Lebenschancen autistischer Kinder, 1970	2
3	Kollokationsanalyse »autistisch« 2010 (Bubble)	2
4	Belegstellen: »autistische Erinnerung und autistische Weltsicht«, 2010	3
5	Frequenzanalyse »autistisch«	3
6	Belegstellen »autistisch« als Metapher und Vorwurf, 2010	3

Abbildung 15: Das Abbildungsverzeichnis in der PDF-Ausgabe

Weiteres

Einen Seitenumbruch erzwingen

Manchmal kann es für die Übersichtlichkeit sinnvoll sein, einen Seitenumbruch zu erzwingen. Zum Beispiel zwischen dem Ende des Texts, dem Abbildungsverzeichnis und der Quellenliste. Dies erreicht man, indem man an die gewünschte Stelle den Befehl ...

`\pagebreak`

... eingibt.

Auch für diesen Befehl habe ich mir ein [Snippet](#) angelegt.

PDF-Einstellungen

Das Aussehen des PDFs, z. B. die Breite der Ränder, oder Überschriften mit oder ohne Nummerierung und andere Details werden über eine Export-Konfiguration definiert. Auf diese werde ich hier nicht mehr eingehen, da auch die von Zettlr mitgelieferten Grundeinstellungen zumeist ausreichend sind. Aber auch sie findet man im »Assets Manager«, in dem sich auch die »Snippets« befinden, und zwar unter dem Reiter »Exporting«.

Für den Export als PDF sucht man nach dem Eintrag mit dem Hinweis »Markdown → PDF«.

Die PDF-Export-Einstellungen anpassen

Ausgabe mit Inhaltsverzeichnis

In den meisten Fällen sind die Grundeinstellungen völlig ausreichend. Möchte man aber zum Beispiel ein Inhaltsverzeichnis unter dem Titel-Block einbinden, muss man die Standard-Konfigurationsdatei im Zettlr-Asset-Manager kopieren und die Zeile ...

```
toc: false # Include a table of contents?
```

... abändern, zu:

```
toc: true # Include a table of contents?
```

Auch die nächste Zeile muss dann eventuell angepasst werden. Sie definiert, welche Überschriften im Inhaltsverzeichnis angezeigt werden.

Wir erinnern uns, in Markdown gibt die Zahl der »#« an, welche Ebene die Überschrift hat. Die Zeile:

```
toc-depth: 2 # 2 means to only include headings level 1 and 2 in the ToC
```

schließt die Überschriften der ersten und der zweiten Ebene ein, also:

```
# Überschrift 1. Ebene
```

```
## Überschrift 2. Ebene
```

Möchte man 3 Ebenen im Inhaltsverzeichnis anzeigen, ändert man die 2 zur 3.

```
toc-depth: 3 # 2 means to only include headings level 1 and 2 in the ToC
```

Eingeschlossen sind nun:

```
# Überschrift 1. Ebene
```

```
## Überschrift 2. Ebene
```

```
### Überschrift 3. Ebene
```

Meistens kommt es der Übersichtlichkeit zugute, wenn nur die ersten zwei, maximal die ersten drei Ebenen im Inhaltsverzeichnis angezeigt werden. Das hängt aber vom Thema und der Art des Dokuments ab.

Ein Schritt weiter: Die Dokumentenklasse `scrartcl`

Um Markdown-Dateien mit Zettlr exportieren zu können, muss man auf dem Rechner, den man zum Export nutzen möchte, die [LaTeX-Umgebung installieren](#). Diese installiert eine ganze Menge nützlicher Erweiterungen gleich mit. So auch das Paket namens »Koma-Script«.

Koma-Script bringt eine ganze Menge nützlicher Einstellungen für akademische Texte mit. Einfach nur die Zeilen ...

```
variables:  
  documentclass: scrartcl
```

... zur Konfigurationsdatei für den PDF-Export hinzuzufügen hat zur Folge, dass alle Überschriften in einem serifenlosen Font gesetzt werden, während der Textkörper weiterhin einen Serifen-Font nutzt — üblich und von der Fakultät vorgegeben ist Times New Roman. Das gibt dem Dokument ein sauberes, etwas moderneres Aussehen.

Alle meine [PDF-Export-Konfigurations-Beispiele](#) nutzen den Koma-Script-Dokumententyp `scrartcl`.

Fazit oder tl;dr

Zettlrs System aus Snippets und einem robusten Export mit LaTeX im Hintergrund, ermöglicht es, sich rein auf den Text zu konzentrieren.

Zu Beginn der Arbeit fügt man den YAML-Block mit einmal drücken des Doppelpunkts aus dem Snippets-Menü ein und füllt nur noch Titel (und Schlüsselwörter) aus, bzw. ergänzt jene Metadaten, die man im Endprodukt sehen möchte.

Bilder zieht man mit der Maus ins Dokument und ergänzt die Bildbeschreibungen.

Mit einem weiteren Drücken des Doppelpunkts fügt man, aus dem Snippets-Menü, ein Bildverzeichnis an der Stelle ein, an der es im Dokument auftauchen soll.

Das Endergebnis wirkt mit wenig Aufwand sauber und ordentlich formatiert.

Es sind auch komplexere Layouts möglich, z. B. mit einem ausführlichen Deckblatt, aber für den alltäglichen Bedarf ist dieses einfache Ausgabeformat ausreichend.

Gibt man im YAML-Header nicht nur seinen Namen und die E-Mail-Adresse, sondern auch sinnvolle Schlüsselwörter ein, werden diese beim PDF-Export übergeben, genau wie die Gliederung. Das PDF besitzt daher sinnvolle Metadaten und kann über das Inhaltsverzeichnis, sofern aktiviert, leicht navigiert werden.

Abbildungsverzeichnis

1	Der ausgefüllte YAML-Block meines Textes zur Kollokationsanalyse . . .	3
2	Der Titel im YAML-Block, der wird sowohl für von Zettlr intern verwendet, als auch bei der Ausgabe	3
3	Zettlr Seitenleiste mit der Übersicht des Arbeitsbereichs und der enthaltenen Dokumente	4
4	Überschrift, Name und Datum (sowie Untertitel) in der PDF-Ausgabe . .	4
5	Weitere Metadaten im YAML-Block	5
6	Header und Footer der PDF-Ausgabe festlegen	5
7	So sieht der Header auf Seite 2 in der PDF-Ausgabe aus	6
8	So sieht der Footer in der PDF-Ausgabe aus. In meinem Beispiel wird mittig die Seitenzahl angegeben und rechts davon das Datum.	6
9	Das Snippets-Auswahlmenü	7
10	Die Variablen im Snippet werden beim Einfügen durch das aktuelle Datum und eine »ID« ersetzt	7
11	Bildauszeichnung	8
12	Die Bild-Seitenleiste von Zettlr zeigt die im derzeit ausgewählten Bildverzeichnis vorhandenen Dateien an	9
13	Die im Dokument hinzugefügte Abbildung im PDF. Die Nummerierung der Bilder wird bei der Ausgabe automatisch vorgenommen.	9
14	Der LaTeX-Befehl ‚listoffigures‘ fügt ein Abbildungsverzeichnis an genau der Stelle ein, an der er im Dokument steht. Der Backslash vor dem Begriff ist wichtig.	10
15	Das Abbildungsverzeichnis in der PDF-Ausgabe	10