

# 1 Klassen- und Paketbeschreibung

Die Klasse `uhb-inf.cls` stellt die Weiterentwicklung der Klasse `pi1.cls` dar, die wiederum eine von Thomas Röfer adaptierte Version aus `pi1-muster.cls` und `defs.tex` war. Nachdem ich mich nun zwei Jahre intensiv mit  $\text{\LaTeX}$  beschäftigt habe, habe ich mir mal die Mühe gemacht einige Dinge auszumerzen, die man so (nicht) mehr macht,<sup>1</sup> oder anders besser lösen kann. Da mir allerdings die `pi1.cls` eine große Hilfe beim Lernen von  $\text{\LaTeX}$  war, denke ich ist es nur konsequent, diese Verbesserungen auch an die Erstsemester weiter zu geben.

Herausgekommen ist dabei eine gut anpassbare Dokumentklasse, die sich nicht nur für PI 1–3, sondern auch für Mathe/Theo/etc. einsetzen lässt.

Im Folgenden seien die Klassenoptionen sowie die bereitgestellten Befehle vorgestellt. Zuvor möchte ich aber die vorausgesetzten und empfohlenen Pakete auflisten.

## (1.a) Paketvoraussetzungen

Es empfiehlt sich immer, ein komplettes und aktuelles  $\text{\TeX}$ -System zur Verfügung zu haben, und nicht nur den *required*-Teil zu installieren. Letzterer erlaubt eher „nur“ rudimentären Schriftsatz, abgefahrenere Kommandos sind aber zum Großteil eben nicht enthalten. Bei heutigen Festplattenpreisen sollten 1,5 GB für Texlive 2008<sup>2</sup> nicht schmerzen.

Wer dennoch ein Minimalsystem sein eigen nennt, muss wenigstens diese  $\text{\LaTeX}$ -Pakete installiert haben:

- `amssymb`<sup>3,4</sup>
- `geometry`<sup>5</sup>
- `listings`<sup>6</sup>
- `fancyhdr`<sup>7</sup>
- `enumitem`<sup>8</sup>
- Fonts `mathpazo` und `beramono`
- oder Fonts `mathptmx`, `helvet` und `courier`
- oder Font `lmodern`

Daneben werden diese Standardpakete eingebunden (die sowieso immer eingebunden werden sollten):

- `ifthen`,
- `inputenc` mit passender Kodierung (siehe unten),
- `babel` mit neuer deutscher Rechtschreibung,
- `fontenc`,
- `graphicx`,
- `calc`,
- `amsmath`,
- `textcomp`,
- `verbatim`

Wie man die einzelnen Pakete nachinstalliert ist sowohl betriebssystem- als auch  $\text{\TeX}$ -distributionsspezifisch. Ist ein komplettes  $\text{\TeX}$ -System eingerichtet, sind die erforderlichen Pakete ebenfalls dabei...

<sup>1</sup><http://www.ctan.org/tex-archive/info/l2tabu/>

<sup>2</sup><http://www.tug.org/texlive/>

<sup>3</sup><http://tug.ctan.org/tex-archive/fonts/amsfonts/latex>

<sup>4</sup><http://www.ams.org/tex/amsfonts.html>

<sup>5</sup><http://tug.ctan.org/cgi-bin/ctanPackageInformation.py?id=geometry>

<sup>6</sup><http://tug.ctan.org/cgi-bin/ctanPackageInformation.py?id=listings>

<sup>7</sup><http://tug.ctan.org/cgi-bin/ctanPackageInformation.py?id=fancyhdr>

<sup>8</sup><http://tug.ctan.org/cgi-bin/ctanPackageInformation.py?id=enumitem>

---

## (1.b) Klassenoptionen

Zur schnellen Anpassung gibt es eine Reihe von Optionen, die in der Form

```
\documentclass[option-1,...,option-n]{uhb-inf}
```

angegeben werden können. Einige sind obligatorisch, ein Fehlen wird durch eine Warnung beim Kompilervorgang bekannt gegeben.

### (1.b.1) Eingabekodierung

Zur Wahl stehen **utf**, **mac**, **win** und **unix**, die entsprechende Zeichenkodierungen (*utf8*, *applemac*, *latin15* bzw. *latin1*) nachladen. Wird keine dieser Optionen angegeben, erscheint der Hinweis, dass man eine eigene Eingabekodierung angeben sollte. Es gibt hier also keinen Default-Wert.

### (1.b.2) Schriften

Es stehen drei Schrift-Themen zur Auswahl:

1. Das klassische L<sup>A</sup>T<sub>E</sub>X-Erscheinungsbild mit „Computer Modern“-Fonts, bzw. der verbesserten Variante „Latin Modern“. Hierzu ist keine Option nötig.
2. Eine modernere Variante mit Palatino als Brotschrift und Bera Mono als dicktengleiche Schrift für den Quelltext (wie in diesem Dokument). Hierzu gibt es die Option **palatino**.
3. Als letztes gibt es noch ein „sparsames“ Thema, das native PDF-Schriften verwendet und so die Dateigröße der PDF signifikant verkleinert. Allerdings sind die mit **times** eingestellten Schriften (Helvetica, Times und Courier) nicht sonderlich schön.

## (1.c) Anzeige verschiedener Variablen

Wie später noch gezeigt wird, lassen sich einige definierte Variablen entfernen. Im einzelnen sind dies:

- **tutor-aus** – unterbindet die Ausgabe des Tutors im Titel und den Kopfzeilen
- **gruppe-aus** – „Gruppe“ plus Nummer werden nicht ausgegeben
- **abgabe-aus** – das Abgabedatum erscheint nicht
- **semester-aus** – die Semesterangabe wird unterdrückt
- **beschreibung-aus** – die Angabe der Beschreibung („Aufgabenblatt“ bzw. „Lösungsblatt“) wird unterbunden
- **veranstaltung-aus** – dies schaltet die Ausgabe der Veranstaltungsbezeichnung aus.

### (1.c.1) Verschiedenes

Per Default wird unter der Veranstaltungsbezeichnung die Zeichenkette „Lösungsblatt“ ausgegeben, mit der Option **aufgabenblatt** wird diese zu „Aufgabenblatt“ geändert. Primär ist dies also zur Unterscheidung zwischen Aufgaben und Lösungen zu verstehen.

---

Die bei diesem Dokument aktiven Optionen lauten:

```
\documentclass[
  nicefonts,
  gruppe-aus,semester-aus,veranstaltung-aus,tutor-aus,abgabe-aus,
  utf
]{uhb-inf}
```

## (1.d) Befehle und Umgebungen

Die wichtigsten Befehle zum Einrichten des Titels und der Kopfzeilen sind die folgenden:

- `\veranstaltung{<kurz>}{<lang>}` – speichert die Bezeichnung der Veranstaltung; `<lang>` sollte soetwas wie „Praktische Informatik 1“ sein, während `<kurz>` nur „PI1“ bekommt.
- `\gruppe{<nummer>}{<teilnehmer>}` – speichert die Gruppenbezeichnung `<nummer>` sowie die Teilnehmer `<teilnehmer>`. Die Teilnehmer müssen durch `\\` getrennt sein, z.B. `\gruppe{$\alpha}{Jane Doe\\Max Mustermann}`.
- Mit `\zettel{<nummer>}{<abgabedatum>}` kann die Zettel-Bezeichnung gespeichert werden.
- `\semester{<bezeichnung>}` speichert das Semester, z.B. `\semester{WS 2009/10}`.
- `\tutorin{<name>}` oder `\tutor{<name>}` speichert den Namen der Tutorin bzw. des Tutors.
- Zur Anpassung der „Beschreibung“ (also je nach Option „Aufgabenblatt“ oder „Lösungsblatt“) kann `\beschreibung{<beschreibung>}` verwendet werden.
- Im Gegensatz zu den Standardklassen wird das Makro `\maketitle` automatisch ausgeführt und kann daher nicht vergessen werden.

Daneben gibt es die Umgebung `Java`, welche Java-Quellcode nicht nur wiedergibt, sondern auch ein wenig aufhübscht. Prinzipiell handelt es sich dabei um eine `lstlisting`<sup>9</sup>-Umgebung, eben speziell auf Java zugeschnitten. Dieser Umgebung können verschiedene Optionen mitgegeben werden (siehe dazu die Doku), per Default haben alle Listings diese Optionen aktiv:

```
1 \lstset{
2   numbers=left,
3   basicstyle=\footnotesize\ttfamily,
4   numberstyle=\tiny\sffamily,
5   tabsize=4,
6   xleftmargin=30pt,
7   breaklines=true,
8 }
```

Für die Java-Umgebung ist noch zusätzlich `language=Java` gesetzt.

## (1.e) Bezug

Zu dieser Klasse wurde ein *Google Code*-Projekt mitsamt SVN-Repository angelegt. Das Projekt ist unter <http://code.google.com/p/uhb-inf-tex> zu erreichen, auf das Repository kann via

```
svn checkout http://uhb-inf-tex.googlecode.com/svn/trunk/ ./uhb-inf-tex
```

anonym zugegriffen werden. Weiter sind auf der Projektseite ein Wiki mit Installationsanleitungen und weitere Ressourcen verlinkt.

## (1.f) Minimalbeispiele

Im Repository finden sich unter `trunk/doc/minimalbeispiele` zwei kleine Beispiele, einmal ein Pseudo-Aufgabenzettel und einmal eine dazugehörige Abgabe.

---

<sup>9</sup>Siehe dazu `listings`-Paket, <http://tug.ctan.org/tex-archive/macros/latex/contrib/listings/>

## (1.g) Ausblick

Einige Dinge, die noch auf der Todo-Liste stehen sind:

- Quellcodedokumentation ausbauen

## 2 Ein Beispieldokument

WS2009/10

### Praktische Informatik 1

Tutor niemand

Gruppe dmke

Dominik Menke

#### Übungszettel 1

Abgabe: 04.11.2009

#### Aufgabe 1 Foo

Dies hier ist ein Blindtext zum Testen von Textausgaben. Wer diesen Text liest, ist selbst schuld. Der Text gibt lediglich den Grauwert der Schrift an. Ist das wirklich so? Ist es gleichgültig ob ich schreibe: »Dies ist ein Blindtext« oder »Huardest gefburn«? Kjift – mitnichten! Ein Blindtext bietet mir wichtige Informationen. An ihm messe ich die Lesbarkeit einer Schrift, ihre Anmutung, wie harmonisch die Figuren zueinander stehen und prüfe, wie breit oder schmal sie läuft. Ein Blindtext sollte möglichst viele verschiedene Buchstaben enthalten und in der Originalsprache gesetzt sein. Er muß keinen Sinn ergeben, sollte aber lesbar sein. Fremdsprachige Texte wie »Lorem ipsum« dienen nicht dem eigentlichen Zweck, da sie eine falsche Anmutung vermitteln.

##### (1.a) bar

Dies hier ist ein Blindtext zum Testen von Textausgaben. Wer diesen Text liest, ist selbst schuld. Der Text gibt lediglich den Grauwert der Schrift an. Ist das wirklich so? Ist es gleichgültig ob ich schreibe: »Dies ist ein Blindtext« oder »Huardest gefburn«? Kjift – mitnichten! Ein Blindtext bietet mir wichtige Informationen. An ihm messe ich die Lesbarkeit einer Schrift, ihre Anmutung, wie harmonisch die Figuren zueinander stehen und prüfe, wie breit oder schmal sie läuft. Ein Blindtext sollte möglichst viele verschiedene Buchstaben enthalten und in der Originalsprache gesetzt sein. Er muß keinen Sinn ergeben, sollte aber lesbar sein. Fremdsprachige Texte wie »Lorem ipsum« dienen nicht dem eigentlichen Zweck, da sie eine falsche Anmutung vermitteln.

```
1 public class HalloWelt {
2     public static final void main(String[] args) {
3         System.out.println("Hallo_Welt!");
4     }
5 }
```

##### (1.b) baz

- Erster Listenpunkt, Stufe 1
- Zweiter Listenpunkt, Stufe 1
- Dritter Listenpunkt, Stufe 1
- Vierter Listenpunkt, Stufe 1
- Fünfter Listenpunkt, Stufe 1

Dies hier ist ein Blindtext zum Testen von Textausgaben. Wer diesen Text liest, ist selbst schuld. Der Text gibt lediglich den Grauwert der Schrift an. Ist das wirklich so? Ist es gleichgültig ob ich schreibe: »Dies ist ein Blindtext« oder »Huardest gefburn«? Kjift – mitnichten! Ein Blindtext bietet mir wichtige Informationen. An ihm messe ich die Lesbarkeit einer Schrift, ihre Anmutung, wie harmonisch die Figuren zueinander stehen und prüfe, wie breit oder schmal sie läuft. Ein Blindtext sollte möglichst viele verschiedene Buchstaben enthalten und in der Originalsprache gesetzt sein. Er muß keinen Sinn ergeben, sollte aber lesbar sein. Fremdsprachige Texte wie »Lorem ipsum« dienen nicht dem eigentlichen Zweck, da sie eine falsche Anmutung vermitteln.

1

Der Quellcode hierzu findet sich ebenfalls im Repository.