

Übung zur Benutzung der TUDaExercise-Klasse

Marei Peischl



TECHNISCHE
UNIVERSITÄT
DARMSTADT

Sommersemester 2042 – v3.00 (2020/09/08)
Übungsblatt 5

Aufgabe 5.1: Titelsei

Im Gegensatz zu den ganzen Titelseiten bei TUDaPub benötigt TUDaExercise eine platzsparende Möglichkeit für die Titelerzeugung. Daher verfügt TUDaexercise nur über einen Titelkopf. Für die Übergabe der Daten werden die folgenden Makros ausgewertet:

```
\title[Kurztitel für die Kopfzeile]{Titel}  
\subtitle{Untertitel}  
\author{Autor*in/Dozent*in}  
\term{Semester}  
\date{Datum}  
\sheetnumber{Nummer des Übungsblatts}
```

Mit Ausnahme des Titels können alle Felder auch leer bleiben. Sofern keine Blattnummer übergeben wird, werden die Übungsaufgaben beginnend mit 1, fortlaufend nummeriert.

Aufgabe 5.2: Farbgebung

Die Farbgebung nutzt die selben Mechanismen wie die übrigen Klassen des TUDa-CI-Bundles. Die Optionen `color`, `accentcolor`, `textaccentcolor`, `identbarcolor` funktionieren wie gewohnt (vgl. TUDaPub). Darüber hinaus ist es auch hier möglich den farbigen Titelblock zu deaktivieren. Die Option `colorback=false` ermöglicht es für Versionen zum Ausdrucken die Tonersparende Variante einzustellen.

Aufgabe 5.3: Eine Aufgabe erstellen

Aufgaben werden bei `tudaexercise.cls` durch die Umgebung `task` definiert. Das Notwendige Argument übernimmt dabei eine Überschrift. Es kann leer bleiben.

```
\begin{task}[<Optionen>]{Überschrift}  
Text der Aufgabe  
\end{task}
```

Querverweise zwischen den Aufgaben sind wie üblich mit `\label` und `\ref` möglich.

Neben der Umgebungsstruktur, die zur besseren Übersichtlichkeit zu empfehlen ist, verfügt `tudaexercise` auch über die Möglichkeit eine Aufgabe nur mit `\task` einzuleiten.

```
\task{Überschrift der Aufgabe}
```

In diesem Dokument wird lediglich die empfohlene Variante unter Nutzung der Umgebung gezeigt.

Falls die Makro-Version benutzt und damit Optionen gesetzt werden, ist zu beachten, dass diese nicht automatisch wieder beendet werden kann, da ja das Ende der Umgebung fehlt. Dies ist der Hauptgrund für die Empfehlung der Umgebung.

Aufgabe 5.4: Kopfzeilenanpassungen

Der Mechanismus für die Kopfzeilen wurde angepasst, sodass sein Inhalt frei wählbar ist.

```
\ConfigureHeadline{
  headline={Inhalt der Kopfzeile}
}
```

Es existiert nun eine Unterscheidung zwischen linken und rechten Seiten, sofern `twoside=true`.

```
\ConfigureHeadline{
  even={Inhalt der Kopfzeile für linke Seiten},
  odd={Inhalt der Kopfzeile für rechte Seiten},
  onside={Inhalt der Kopfzeile im einseitigen Modus}
}
```

Einstellungen, die gleiche Felder betreffen, überschreiben sich gegenseitig.

5.4a)

Die Inhalte sind frei wählbar und werden automatisch in einer Box mit der gleichen Breite, wie der Text gesetzt. Der Inhalt wird linksbündig platziert. Es ist möglich Grafiken oder Tabellen innerhalb dieser Box zu platzieren.

5.4b)

Für die Kopfzeilen existieren einige vorgefertigte Elemente, die dort platziert werden können.

```
\ShortTitle
\StudentID
\StudentName
```

Darüber hinaus kann auch jeglicher \LaTeX -Content dort eingefügt werden.

Zusätzlich sind einige Werte vorbelegt und platzieren die Einträge nach dem selbsterklärenden Namensschema: `title-name-id`, `title-name`, `title`, `name-id`, `name`

5.4c)

Da die Kopfzeile häufig für die Angabe von Studentendaten genutzt wird, ist es mit der Klassenoption `headtotitle` auch möglich die Kopfzeile ebenfalls auf der Titelseite einzufügen.

Aufgabe 5.5: Lösungen

tudaexercise verfügt über einen Mechanismus um Lösungsvorschläge innerhalb der Dateien mit zu verwalten. Hierfür existiert die Umgebung `solution`.

Die Ausgabe der Lösungen wird entweder global

```
\documentclass[...,  
  solution=true,  
{tudaexercise}
```

oder lokal für einzelne Aufgaben konfiguriert:

```
\begin{task}[solution=true]{Überschrift}  
Aufgabentext  
  \begin{solution}  
    Lösungstext  
  \end{solution}  
\end{task}
```

Die Voreinstellung ist `solution=false`.

Lösungsvorschlag:

Beispiel für einen Lösungsvorschlag

Aufgabe 5.6: Punkte (10)

Punkteangaben sind über den Optionsschlüssel `credit` möglich. Der Wert wird hinter den Aufgabentitel in runde Klammern gesetzt. Dies geschieht mithilfe des Makros `\creditformat`

Alternativ existiert der Schlüssel `points` dieser ergänzt zusätzlich zur Angabe auch den Text `\PointName` bzw. `\PointsName` je nachdem ob der Wert größer 1 oder nicht. Der Schlüssel `points` akzeptiert daher nur numerische Werte.

5.6a) Beispiel für den Schlüssel `points` (1 Punkt)

Da ältere Versionen von TUDaExercise keine Optionsverarbeitung für `subtask` erlaubten, ist hier ein zusätzlicher Mechanismus enthalten. Falls sowohl ein Titel, als auch spezifische Angaben notwendig sind, existiert hier zusätzlich der Schlüssel `title`. Geprüft wird darauf, ob das Argument ein Gleichheitszeichen enthält, ist dies der Fall, so wird davon ausgegangen, dass Optionen nach Schlüssel=Wert-Struktur übergeben werden.

5.6b) Fallback-Modus, falls ein `=` im Titel detektiert wird, aber keine Zuweisung möglich ist.

Falls bei der Optionsverarbeitung Unstimmigkeiten festgestellt werden, so wird das gesamte Argument als Titel verarbeitet.

Aufgabe 5.7: Beta-Feature: Einheitliche Syntax für Task/Subtask

TUDaExercise unterscheidet in der Standardeinstellung zwischen `task` und `subtask`. Dies ist dadurch begründet, dass `subtask` keine Angabe eines Titels erfordert.

Es wurde der Bedarf nach einer einheitlichen Syntax geäußert (siehe https://github.com/tudace/tuda_latex_templates/issues/189).

5.7a) Die gesternten Varianten der task/subtasks-Umgebungen

Die mit Version 3.0 implementierten Varianten `task*` und `subtask*` ermöglichen dies.

Beide Varianten stehe nur als Umgebung und nicht als Makro zur Verfügung:

```
\begin{task*}{Aufgabentitel}
  \begin{subtask*}{Titel der Teilaufgabe}
    Beschreibung
  \end{subtask*}
\end{task*}
```

5.7b) Vertauschen der beiden Varianten

Zusätzlich erlaubt die Klassenoption `match-task-subtask-syntax` das Vertauschen von `subtask` und `subtask*`.

Die Umgebungen `task*` und `task` sind identisch.

Aufgabe 5.8: Kompatibilitätsmodus

Die Klasse `tudaexercise` verfügt über einen Kompatibilitätsmodus, um den Umstieg von zuvor existierenden Templates, wie `TUDexercise` zu erleichtern. Um den Kompatibilitätsmodus zu aktivieren, existiert die Option `compat=true`.

Damit können Strukturen wie `examheader`, `examheaderdefault` und die Erzeugung von Aufgaben mithilfe von `\subsection` wie gewohnt verwendet werden. Da einige dieser Mechanismen jedoch der Philosophie der semantischen Auszeichnung widersprechen wird ihre Verwendung hier nicht genauer erklärt. Nutzer, denen diese Mechanismen nicht vertraut sind, wird empfohlen die durch TUDaExercise zur Verfügung gestellten Varianten zu nutzen.

Beim zugehörigen Release im Development Repository findet sich auch eine Beispieldatei, die die alten Mechanismen verwendet und spezielle Hinweise für Umsteiger enthält.