

## Anleitung zu den Buchungsaufgaben

Fred Neumann, 15.07.2019 00:01:00

### Neuerungen

#### Version 1.7.2 (ILIAS 5.3) und 1.8.1 (ILIAS 5.4)

- Definition von Variablen und ihre Verwendung in Fragentexten, Buchungen und Feedback.
- Festlegung der Genauigkeit beim Vergleich von Werten

#### Version 1.3 (ILIAS 5)

- Die Flash-Komponente der Aufgabe wurde durch eine HTML5-Darstellung ersetzt. Die Berechnung des Ergebnisses erfolgt nun in ILIAS statt in Flash und erlaubt eine Auswertung in der Vorschau. Entsprechend sind die Einstellungen **bestanden\_ab** und **debug** obsolet und werden nicht mehr interpretiert.
- **Mehrere Buchungssätze pro Tabelle werden nicht mehr unterstützt.** Die Oberfläche zeigt keinen Schalter zum Beginn eines neuen Buchungssatzes mehr an. Verwenden Sie stattdessen Teilaufgaben. Entsprechend sind die Einstellungen **bonus\_reihe** und **malus\_anzahl** beim Aufgabentyp Buchungssätze obsolet und werden nicht mehr interpretiert.

### Grundfunktion

Mit dem Aufgabentyp „Buchungsaufgabe“ können Buchungstabellen zweierlei Typs generiert werden (T-/Bilanzkonto oder Buchungssätze), die von den Studenten ausgefüllt werden müssen und die mit Vorgabewerten verglichen wird. Die Vorgabe und die erreichbaren Punkte werden in der Aufgabendefinition über XML konfiguriert.

Mit jeder Aufgabe wird ein Kontenrahmen definiert und es können zu einer textuellen Aufgabenstellung (z.B. Schilderung eines Geschäftsvorfalles) mehrere Buchungstabellen zum Ausfüllen dargestellt werden.

### Verwendung in ILIAS

Der Aufgabentyp kann in ILIAS als Plugin installiert werden. Dazu wird er in einer ILIAS-Installation unter folgendem Pfad abgelegt:

Customizing\global\plugins\Modules\TestQuestionPool\Questions\assAccountingQuestion

Danach muss das Plugin unter *Administration > Plugins* aktualisiert und aktiviert werden.

Buchungsfragen können dann wie alle anderen Fragetypen erzeugt und verwendet werden. Beim Anlegen einer Buchungsfrage müssen die folgenden Eigenschaften festgelegt werden:

- Titel
- Autor/In
- Fragestellung (z.B. Schilderung eines Geschäftsvorfalles, der verbucht werden soll)
- Definition eines Kontenrahmens in XML. Ein Beispiel ist als **kontenrahmen.xml** im Unterverzeichnis *examples* des Plugins zu finden.

- Definition mindestens einer Tabelle für einen Buchungssatz oder ein T-Konto in XML. Ein Beispiel ist als **buchungen.xml** oder **buchungen\_konto.xml** im Unterordner *examples* des Plugins zu finden.

Optional können in XML Variablen definiert werden, die ihre Werte per Zufallsauswahl erhalten. Sie lassen sich in den Fragetexten, Buchungen oder im Feedback verwenden.

Mit einer Aufgabe können mehrere Buchungstabellen definiert werden. Zu jeder Buchungstabelle lässt sich eine Teilfrage vorsehen, die über der Tabelle angezeigt wird.

### XML-Definition des Kontenrahmens

Die Kontenrahmen-Definition hat für beide Aufgabentypen (T-Konten oder Buchungssätze) den gleichen Aufbau. Mit ihr wird festgelegt welche Kontennummern und Namen in den Buchungen zur Auswahl stehen:

```
<?xml version="1.0" encoding="utf-8" ?>
<konten anzeige=" beide">
    <konto nummer= "800"    titel="AB" />
    <konto nummer= "801"    titel="SB" />
    <konto nummer= "300"    titel="Bank" />
    ...
</konten>
```

Jedes Element **konto** hat die Attribute **nummer** für die Konten-Nummer und **titel** für den Konten-Titel. Beide sollten eindeutig sein.

Es muss ein umfassendes Element **konten** vorhanden sein, das beliebig viele Konten enthalten kann. Mit dem Attribut **anzeige** wird festgelegt, was in den Auswahlfeldern angezeigt werden soll:

- nummer            nur die Konten-Nummer
- titel             nur der Konten-Titel
- beide            Nummer: Titel

### XML-Defintion der Buchungen

Das Wurzelement der XML-Definition bestimmt den Typ der Aufgabe:

```
<konto>            →            T-Konten/Bilanzkonten
<buchungssätze> →            Buchungssätze
```

### Import von XML-Dateien

Die XML-Definitionen des Kontenrahmens oder von Buchungen können im Editor der Frage direkt bearbeitet oder aus einer XML-Datei eingelesen werden. Wenn Sie XML-Definitionen importieren, müssen die XML-Dateien UTF8-codiert sein und sollten eine 2 Byte-lange UTF8-Signatur am Dateibeginn haben. Dann werden sie, auch wenn sie keine Umlaute enthalten vom Editor Notepad als UTF-8 erkannt und neu eingefügte Umlaute entsprechend codiert. Die Beispieldateien im Verzeichnis *examples* sind entsprechend codiert.

## Aufgabentypen

### T-Konten / Bilanzkonten

Aufgaben zu T-Konten haben folgende Struktur:

S	Kontentitel		H
[Auswahl Konto ▼]	[Eingabe Betrag]	[Auswahl Konto ▼]	[Eingabe Betrag]
...	...	...	...
Summe:	Anzeige Summe	Summe:	Anzeige Summe

Einzugeben sind die Buchungen auf der Soll- und Haben-Seite mit Auswahl des Kontos und freier Eingabe des Betrags. Die Seiten werden unabhängig voneinander betrachtet.

### XML-Definition von T-Konten

Die Datei buchungen.xml enthält die Lösungen/Vorgabewerte für die Buchungsaufgabe. Sie hat bei T-Konten oder Bilanzkonten den folgenden Aufbau:

```
<?xml version="1.0" encoding="utf-8" ?>
<konto    links="S"
          mitte="Kontentitel"
          rechts="H"
          zeilen="10"
          summen_zeigen="1"
          bonus_reihe_links="0"
          bonus_reihe_rechts="0"
          malus_anzahl_links="0"
          malus_anzahl_rechts="0"
          malus_summen="0"
>
    <links    konto="800"    betrag="100"    punkte="1" />
    ...
    <rechts    konto="801"    betrag="200"    punkte="1" />
    ...
</konto>
```

Es muss ein umfassendes Element **konto** vorhanden sein, das beliebig viele Vorgaben für die linke und rechte Seite als Unterelemente (**links**, **rechts**) enthalten kann. Das Element **konto** hat folgende Attribute:

<b>links</b>	Titel über der linken Spalte, z.B. "S" oder "Aktiva"
<b>mitte</b>	Titel in der Mitte, normalerweise der Name des Kontos
<b>rechts</b>	Titel über der rechten Spalte, z.B. "H" oder "Passiva"
<b>zeilen</b>	So viele Zeilen werden zur Eingabe generiert.
<b>summen_zeigen</b>	Anzeige der berechneten Summen in einer Zeile unterhalb der Eingabe ("0" oder "1").

<b>bonus_reihe_links</b>	Bonus für die korrekte Reihenfolge aller richtigen Buchungen auf der linken Seite, wenn davon mindestens zwei existieren.
<b>bonus_reihe_rechts</b>	Bonus für die korrekte Reihenfolge aller richtigen Buchungen auf der rechten Seite, wenn davon mindestens zwei existieren.
<b>malus_anzahl_links</b>	Malus für eine überschrittene Anzahl von Buchungen auf der linken Seite. Es wird aber maximal die links erreichte Punktezahl (inklusive Bonus) abgezogen, so dass das linke Ergebnis nicht negativ wird.
<b>malus_anzahl_rechts</b>	Malus für eine überschrittene Anzahl von Buchungen auf der rechten Seite. Es wird aber maximal die rechts erreichte Punktezahl (inklusive Bonus) abgezogen, so dass das rechte Ergebnis nicht negativ wird.
<b>malus_summen</b>	Malus, wenn die Summe der Buchungen (egal, ob korrekt oder nicht) links und rechts nicht übereinstimmt. Es wird aber maximal die bis dahin erreichte Punktezahl abgezogen, so dass das Gesamtergebnis nicht negativ wird.

Für die Buchungen werden je nach Seite die Unterelemente **links** und **rechts** verwendet. Diese haben folgende Attribute:

<b>konto</b>	Nummer oder Titel des Kontos.
<b>betrag</b>	Betrag. Cent-Beträge werden mit Komma abgetrennt. Tausenderpunkte werden ignoriert.
<b>punkte</b>	Anzahl der Punkte, falls die Buchung korrekt eingegeben wurde.

## Buchungssätze

Aufgaben zu Buchungssätzen haben folgende Struktur:

von	Kontentitel			an
[Auswahl Konto ▼]	[Eingabe Betrag]		[Auswahl Konto ▼]	[Eingabe Betrag]
...	...		...	...
<b>Summe:</b>	Anzeige der Summe		<b>Summe:</b>	Anzeige der Summe

Einzugeben sind die Konten und Beträge auf der linken und rechten Seite.

## XML-Definition von Buchungssätzen

Die Datei buchungen.xml enthält die Vorgaben für die Buchungsaufgabe. Sie hat bei Buchungssätzen den folgenden Aufbau:

```
<?xml version="1.0" encoding="utf-8" ?>
<buchungsaetze links="von"
               mitte="Kontentitel"
               rechts="an"
               zeilen="10"
               summen_zeigen="1"
>
  <satz      malus_anzahl_von="0"
```

```

        malus_anzahl_an="0"
        malus_summen="0"
    >
        <von konto="800"        betrag="100"        punkte="1" />
        ...
        <an  konto="801"        betrag="200"        punkte="1" />
        ...
    </satz>
</buchungsaetze>

```

Es muss ein umfassendes Element **buchungsaetze** vorhanden sein, das die Darstellung definiert. Das Element **buchungsaetze** hat folgende Attribute:

<b>links</b>	Titel über der linken Spalte, normalerweise Regel "Von".
<b>mitte</b>	Titel in der Mitte, normalerweise "Buchungen".
<b>rechts</b>	Titel über der rechten Spalte, normalerweise "An".
<b>zeilen</b>	So viele Zeilen werden zur Eingabe generiert.
<b>summen_zeigen</b>	Anzeige der berechneten Summen in einer Zeile unterhalb der Eingabe ("0" oder "1").

Innerhalb davon gibt es *genau ein* Element **satz**, das einen Buchungssatz definiert. Es hat folgende Attribute:

<b>malus_anzahl_von</b>	Malus für eine überschrittene Anzahl von Buchungen auf der linken "von"-Seite des Buchungssatzes.
<b>malus_anzahl_an</b>	Malus für eine überschrittene Anzahl von Buchungen auf der rechten "an"-Seite des Buchungssatzes.
<b>malus_summen</b>	Malus, wenn die Summe der Buchungen (egal, ob korrekt oder nicht) in diesem Buchungssatz links und rechts nicht übereinstimmt.

Im Gegensatz zu den T-Konten werden die Maluspunkte von Buchungssätzen erst nach der Punktesumme beider Seiten berechnet. Sie werden in der oben angegebenen Reihenfolge vom jeweiligen Zwischenergebnis so abgezogen, dass es nicht negativ wird.

Für die korrekten Buchungen werden je nach Seite die Unterelemente **von** und **an** verwendet. Diese haben folgende Attribute:

<b>konto</b>	Nummer oder Titel des Kontos.
<b>betrag</b>	Korrektter Betrag. Cent-Beträge werden mit Komma abgetrennt. Tausenderpunkte werden ignoriert.
<b>punkte</b>	Anzahl der Punkte, falls die Buchung korrekt eingegeben wurde.

## Variablen

Mit Variablen lassen sich zufallsgesteuert Varianten einer Aufgabe erzeugen. Es gibt Grundvariablen, die Werte per Zufallsauswahl festlegen, bedingte Variablen, die anhand von anderen Variablen ihren Wert auswählen und berechnete Variablen, die mit einer Formel aus anderen Variablen ihren Wert berechnen.

## XML-Definition von Variablen

Variablen werden im Editor der Aufgabe analog zum Kontenrahmen in einem neuen Textfeld per XML definiert. Jede Variable erhält einen eindeutigen Namen, mit dem sie verwendet werden kann. Die Namen dürfen aus alphabetischen Zeichen und Zahlen bestehen.

```
<?xml version="1.0" encoding="utf-8" ?>
<variables >
    <var name="var1" ... />
    <var name="var2" ... />
</variables>
```

### Grundvariablen

Mit Variablen vom Typ **range** wird eine beliebige Zahl aus einem Wertebereich ausgewählt. Mit „step“ kann eine dabei eine Schrittweite festgelegt werden, der Default-Wert ist 1.

```
<var name="var1" type="range" min="10" max="100" step="10" />
```

Mit Variablen vom Typ **select** wird zufällig eine von mehreren Möglichkeiten ausgewählt. Der Inhalt kann ein Text für die Frage oder eine Zahl zur weiteren Berechnung sein.

```
<var name="var2" type="select">
    <val>800</val>
    <val>900</val>
</var>
```

### Bedingte Variablen

Variablen vom Typ **switch** erlauben es, ihren Wert anhand des Wertes einer anderen Variablen zu bestimmen. Damit können z.B. Texte in der Beschreibung des Buchungsfalls abhängig von Variablen realistischer gestaltet werden, z.B. „verkauft Socken“ oder „verkauft einen Anzug“, je nach zufälligem Preis.

Die Fälle werden von oben nach unten abgearbeitet. Entspricht die Variable der Bedingung wird der Inhalt des Falls genommen und die Bearbeitung abgebrochen. Neben Text kann der Inhalt auch ein Zahlenwert für weitere Berechnungen sein.

```
<var name="var3" type="switch" check="{var2}">
    <case value="{var1}">Text A</case>
    <case max="50">Text B</case>
    <case max="150">Text B</case>
    <default>{var2}</default>
</var>
```

Mit dem Attribut *value* wird auf einem bestimmten Wert getestet, mit dem Attributen *max* auf eine obere Grenze. In beiden Fällen wird der Wert als Zahl verglichen.

## Berechnete Variablen

Berechnete Variablen vom Typ **eval** erlauben die Anwendung einer Formel.

```
<var name="calc1" type="eval">
  {var1}*({var2}+{var4})
</var>
```

In der Formel werden zunächst die Platzhalter der Variablen anhand ihrer geschweiften Klammern erkannt und ersetzt. Platzhalter für Variablen, die nicht definiert sind, erhalten den Wert 0.

Anschließend wird die Formel wie in der Formelfrage von ILIAS ausgewertet. Erlaubt sind das beliebige Klammern von Ausdrücken, die mathematischen Operatoren + (Addition), - (Subtraktion), \* (Multiplikation), / (Division), ^ (Potenzieren), die Verwendung der Konstanten 'pi' für die Zahl Pi und 'e' für die Eulersche Zahl, sowie die mathematischen Funktionen 'sin', 'sinh', 'arcsin', 'asin', 'arcsinh', 'asinh', 'cos', 'cosh', 'arccos', 'acos', 'arccosh', 'acosh', 'tan', 'tanh', 'arctan', 'atan', 'arctanh', 'atanh', 'sqrt', 'abs', 'ln', 'log'

Eine Variable darf in einer Berechnung nicht selbst verwendet werden, auch nicht indirekt über andere Variablen.

## Variablen-Verwendung

Zur Verwendung einer Variablen muss immer ihr Name in geschweiften Klammern als Platzhalter angegeben werden. Variablen können an den folgenden Stellen in der Aufgabe verwendet werden:

- im Fragentext
- im Text einer Teilfrage
- in einem Rückmeldungs-Text
- im *check*-Attribut einer bedingten Variable. Hier wird ihr Wert als Zahl verglichen.
- im *value* oder *max*-Attribut einer bedingten Variable. Hier wird ihr Wert als Zahl verglichen.
- im Wert der *case*- oder *default*-Option einer bedingten Variable
- in der Formel einer berechneten Variable. Hier wird ihr Wert als Zahl verglichen.
- als Attribute *konto* oder *betrag* in der XML-Definition von Buchungen. Für Konten wird ihr Textwert verwendet, für Beträge ihr Zahlenwert.

Werden Variablen mit Textwert **als Zahl benötigt**, wandelt das System automatisch so viele Zeichen um, wie als Zahl interpretiert werden können „10,50 EUR“ ergibt 10,50, „Firma X“ ergibt 0.

Wie bei den Beträgen in Buchungen werden Dezimalstellen mit Komma abgetrennt. Tausenderpunkte werden ignoriert.

## Verhalten in der Vorschau und im Test

Wie bei der Formelfrage werden die Variablenwerte bei der erstmaligen Anzeige der Frage in der Vorschau oder in einem Testdurchlauf berechnet und für den Benutzer gespeichert. In der Vorschau im Fragenpool bleiben sie bis Klick auf „Vorschau zurücksetzen“ erhalten. Im Test bleiben sie bis zum Start eines neuen Testdurchlaufs erhalten.

## Eingabe

Nach dem Aufruf einer Fragenseite in der Vorschau oder im Test kann mit [Tab] vorwärts und mit [Umschalt]+[Tab] rückwärts durch die Felder gesprungen werden.

Bei den **Kontenfeldern** ist eine direkte Eingabe der Kontennummer oder die Auswahl per Liste möglich. Das Listenfeld lässt sich mit einer beliebigen Taste aufklappen und mit [Esc] zuklappen. Bei den **Betragsfeldern** ist nur die Eingabe von Ziffern und dem **Komma für Dezimalzahlen** möglich.

## Auswertung

Zur **Bewertung** werden auf jeder Seite alle Buchungen, deren Konto und Betrag mit der Vorgabe übereinstimmt mit den Punkten aus der Vorgabe versehen.

Der **Bonus** für die richtige **Reihenfolge** wird vergeben, sobald auf einer Seite mindestens zwei korrekte Buchungen vorhanden sind. Dazu müssen alle korrekten Buchungen in der richtigen Reihenfolge sei (nur bei T-Kontenaufgaben).

Ein **Überzahl-Malus** kann abgezogen werden, wenn die **Anzahl** der Buchungen auf einer Seite die vorgegebene Anzahl überschreitet.

- Bei **T-Konten** wird der Überzahl-Malus für jede Seite separat von den erzielten Punkten plus Reihenfolge-Bonus so abgezogen, dass das Zwischenergebnis für diese Seite nicht negativ wird
- Bei **Buchungssätzen** werden zunächst die Punkte beider Seiten addiert und anschließend der Malus für links und rechts bestimmt und so abgezogen, dass das jeweilige Zwischenergebnis nicht negativ wird.

Ein **Summen-Malus** kann abgezogen werden, wenn die **Summe** der Buchungen auf einer Seite nicht mit der Summe auf der anderen Seite übereinstimmt. Es wird aber maximal die erreichte Punktezahl abgezogen.

## Berechnungs-Toleranz

Die **Genauigkeit** beim Vergleich von Werten wird in einem neuen Eingabefeld in Form der Anzahl von Nachkommastellen eingegeben. Zwei Werte müssen an diesen Nachkommastellen übereinstimmen, um als gleich zu gelten. Das betrifft die Überprüfung bei bedingten Variablen und die Betragsvergleiche in Buchungen.

## Rückmeldungen

Die Rückmeldung zur Auswertung einer Nutzereingabe kann Informationen zu den vergebenen Teilpunkten enthalten. Das hängt von der Situation und den Rechten des Nutzers ab:

- Beim Test der Frage in der Vorschau eines Fragenpools oder Tests werden die vergebenen Teilbewertungen angezeigt.
- Bei der Ergebniseinsicht durch Test-Teilnehmer (ohne Schreibrecht) werden keine Teilbewertungen angezeigt. Die Anzeige der erhaltenen Gesamtpunkte einer Frage hängt von den Testeinstellungen ab.
- Bei der Ergebniseinsicht durch Testautoren (mit Schreibrecht) werden die Teilbewertungen angezeigt und ein Hinweis darauf eingeblendet.