


Inhaltsverzeichnis

1 Wie kann ich eine neue eigene Fragensammlung erstellen?.....	2
2 Wie kann ich diese Fragensammlung dann in ein eigenes, im Webbrowser lauffähiges Quizprogramm umwandeln?.....	3
3 Wie kann ich eine bereits erstellte Fragendatei erneut bearbeiten, etwa um inhaltliche Fehler zu beheben?.....	4
4 Warum gibt es überhaupt zwei Wege (über das Vorlagenarchiv oder über den direkten MOODLE-HTML-Export), um ein Quizprogramm zu erhalten?.....	5
5 Wie kann ich die Fragendatei oder die HTML-Export-Datei auch ohne den Quizeditor verändern?.....	6
6 Wie kann ich überzählige Frage löschen?.....	7
7 Wie kann ich Bilddateien aus dem Internet in mein Quiz einbinden? [Achtung: Dies erfordert grundlegende HTML-Kenntnisse!].	10
8 Wie kann ich meine Fragentexte und die Lösungsmöglichkeiten formatieren (z.B. mit Fett- oder Kursivdruck hervorheben)? [Achtung: Dies erfordert grundlegende HTML-Kenntnisse!].	12
9 Hilfe – Mein erzeugtes Programm läuft nicht, wie kann ich es „reparieren“?.....	13

1 Wie kann ich eine neue eigene Fragensammlung erstellen?

Auf der Seite chemie-lernprogramme.de/daten/utills/quizeditor kann man ganz einfach in die entsprechenden Formularfelder die eigenen Texte eingeben (oder hineinkopieren, dies allerdings ohne Formatierung, vgl. Frage 8).


Sobald man mit einer Frage fertig ist, kann man durch den Weiter-Pfeil unten zur nachfolgenden Frage gehen. Über den zurück-Pfeil kann man auch zu vorherigen Fragen der aktuell bearbeiteten Fragensammlung zurückkehren.

Die an der vorher bearbeiteten Frage vorgenommenen Änderungen werden in beiden Fällen automatisch übernommen, allerdings muss man die gesamte Fragensammlung nach Abschluss der Bearbeitung über das Symbol  erst lokal abspeichern.

Die Quizeditorseite darf während der Bearbeitung keinesfalls neu geladen oder geschlossen werden, da sonst alle bisherigen Änderungen (also die komplette Fragensammlung) verloren gehen!

2 Wie kann ich diese Fragensammlung dann in ein eigenes, im Webbrowser lauffähiges Quizprogramm umwandeln?

Variante A:

Speichert man die selbst erstellte Fragedatei *questions.js* ab, so benötigt man zusätzlich die als ZIP-Archiv unter dem Symbol  zum Download angebotene Quizprogrammvorlage *quiz.zip*. Diese wird entpackt und man erhält u.a. die folgenden **Verzeichnisse** und *Dateien*:

```
quiz
| -daten
|   | -css
|   | -fonts
|   | -js
|   | -questions.js
|   | ...
| -index.html
```

Als Speicherort wird immer das **Desktop** (=Arbeitsflächen)-Verzeichnis empfohlen, da sowohl die Fragedatei als auch das Archiv dort leicht gefunden wird und auch der frisch entpackte Ordner dann direkt sichtbar ist.


Nun muss man nur noch die bereits vorhandene Beispielfragedatei *questions.js* mit der selbst erzeugten neuen gleichnamigen Datei (also ebenfalls *questions.js*) ersetzen. [Alternativ kann man natürlich auch das Vorlagenarchiv schon im Vorfeld herunterladen und entpacken und die selbst erzeugte Fragedatei dann gleich aus dem Quizeditor heraus an die entsprechende Stelle speichern, wobei dann die in der Vorlage bereits vorhandene überschrieben wird.]

Der Speicherpfad für die eigene Fragedatei muss also lauten:
quiz/daten/js/questions.js

Startet man nun die Datei *index.html* mit einem Webbrowser (empfohlen werden **Firefox**, **Chrome**, **Safari** oder auch **Opera**, nicht jedoch der Internet Explorer), so sollte das Quiz mit den eigenen Fragen starten.

Falls nicht, könnte Frage 9 einen Lösungsweg aufzeigen.

Variante B:


Über das Speichern-Symbol  kann man auch direkt eine einzige HTML-Datei in einem neuen Tab erzeugen und diese dann lokal mit der Dateiendung *.html* abspeichern, also z.B. als *mein_quiz.html*.

Die Datei *mein_quiz.html* wird dann anschließend zum Testen einfach direkt im Webbrowser geöffnet. Alternativ kann man diese Datei auch in einen MOODLE-Kurs hochladen. Bei älteren MOODLE-Versionen (1.9) muss man sie dazu zunächst ins Datei-Backend hochladen und dann z.B. von der Kursseite aus verlinken. Bei neueren MOODLE-Versionen (> 2.0) reicht es aus, die Datei per Drag&Drop einfach in den Kurs „fallenzulassen“ und sie später zum Starten anzuklicken.

3 Wie kann ich eine bereits erstellte Fragendatei erneut bearbeiten, etwa um inhaltliche Fehler zu beheben?

Jede mit dem Quizeditor erzeugte JavaScript-Datei mit der Dateierdung .js und der folgenden Struktur, kann auch wieder mit dem Quizeditor eingelesen und weiterbearbeitet werden.


```
var units=[
  {
    "answer": "b",
    "support": "Gib hier den Erklärungstext ein, der nach einem Quizdurchlauf
bei falscher Beantwortung dieser Frage angezeigt wird.",
    "question": "Gib hier die Fragestellung ein, für die nur eine der unten
stehenden vier Antwortmöglichkeiten richtig ist.",
    "a": "Antwortmöglichkeit A",
    "b": "Antwortmöglichkeit B; diese ist in diesem Beispiel durch die Auswahl
auf der linken Seite als richtig gekennzeichnet",
    "c": "Antwortmöglichkeit C",
    "d": "Antwortmöglichkeit D"
  }, // ← Achtung, bei allen anderen Fragen muss hier ein Komma stehen!
  {
    "answer": "a",
    "support": "Erklärungstext für die 2. Frage",
    "question": "Fragestellung für die 2. Frage",
    "a": "Antwortmöglichkeit A2; diese ist in diesem Beispiel oben durch die
Angabe b unter 'answer' als richtig gekennzeichnet",
    "b": "Antwortmöglichkeit B2",
    "c": "Antwortmöglichkeit C2",
    "d": "Antwortmöglichkeit D2"
  } // ← Achtung, bei der letzten Frage darf hier kein Komma mehr stehen!
]
```


Zum Einlesen der lokal vorhandenen Datei wird über das Symbol  ein Dateiauswahl-Dialog aufgerufen, über den man diese dann komfortabel auswählen kann.

In der ersten Zeile darf kein zusätzliches Leerzeichen vorkommen, so wären etwa die folgenden Varianten fehlerhaft und könnten nicht geladen werden:

```
var units =[
var units = [
```

4 Warum gibt es überhaupt zwei Wege (über das Vorlagenarchiv oder über den direkten MOODLE-HTML-Export), um ein Quizprogramm zu erhalten?

Die **Variante A** über  (vgl. Frage 2) hat den Vorteil, dass man die Fragendatei *questions.js* bei aufwändigeren Projekten und längeren Tests immer wieder einfach direkt im Quizeditor neu weiterbearbeiten kann. [Das Abspielprogramm ist optisch etwas aufwändiger gestaltet als in der Einzeldateivariante B.] Dafür kann man als Nachteil die komplette Quizprogrammverzeichnisstruktur nicht in MOODLE einbetten.

Die **Variante B** über  (vgl. Frage 2) hat den Vorteil, dass man die erzeugte HTML-Datei sofort öffnen und so direkt starten kann, ohne diese erst an eine bestimmte Stelle kopieren zu müssen. Außerdem kann diese HTML-Datei auch einfach in Lernplattformen wie MOODLE, aber z.B. auch in andere Content-Management-Systeme wie das der Schulhomepage (aber auch z.B. JOOMLA,...) eingebunden werden.

Nachteilig ist hier die erschwerte Weiterbearbeitung, da die sehr große Datei nicht mehr im Quizeditor weiterbearbeitet werden kann und insgesamt aufgrund der enthaltenen Stilvorlagen und anderen Skripte recht unübersichtlich ist. Eine Weiterbearbeitung der HTML-Datei in einem einfachen Texteditor (vgl. Frage 5) ist aber grundsätzlich möglich.

5 Wie kann ich die Fragendatei oder die HTML-Export-Datei auch ohne den Quizeditor verändern?

Sowohl die Fragendatei `questions.js` als auch die exportierte HTML-Datei (z.B. `mein_quiz.html`) können einfach mit jedem beliebigen Texteditor weiterbearbeitet werden. Dieser sollte allerdings das sogenannte [Syntaxhighlighting](#) beherrschen, damit Syntaxfehler die sonst zu einem Programmabbruch führen würden, sofort farbig hervorgehoben werden und so vermieden werden können. Eine mögliche Auswahl ist je nach Betriebssystem z.B.

- **Gedit** (projects.gnome.org/gedit),
- **Geany** (www.geany.org) oder
- **Notepad++** (notepad-plus-plus.org).

In allen Fällen sollte man in den Einstellungen aller Texteditoren darauf achten, dass man beim erneuten Abspeichern als Zeichenkodierung wieder UTF-8 auswählt. Als Zeilenumbruch sollte immer „Newline(\n)/Unix“ gewählt werden.

Ansonsten könnte es dazu kommen, dass z.B. Umlaute oder andere Sonderzeichen als fehlerhafter „Zeichensalat“ angezeigt werden.

Eine typische Frage besteht dabei aus einem solchen Block (im [JSON](#)-Format):

```
{
  "answer": "b",
  "support": "Gib hier den Erklärungstext ein, der nach einem Quizdurchlauf
bei falscher Beantwortung dieser Frage angezeigt wird.",
  "question": "Gib hier die Fragestellung ein, für die nur eine der unten
stehenden vier Antwortmöglichkeiten richtig ist.",
  "a": "Antwortmöglichkeit A",
  "b": "Antwortmöglichkeit B; diese ist in diesem Beispiel durch die Auswahl
auf der linken Seite als richtig gekennzeichnet",
  "c": "Antwortmöglichkeit C",
  "d": "Antwortmöglichkeit D"
}, // ← Achtung, bei allen außer der letzten Frage muss hier ein Komma stehen!
```

Die hier hellmagenta gekennzeichneten Bereiche können **beliebig abgeändert** werden, allerdings muss man vor jedes eingeschlossene **doppelte Hochkomma** jeweils einen **Backslash** schreiben, also z.B. in der Form:

```
"Bla \"blubb\" nochwas"
```

Zusätzlich zu reinem Text können hier auch einfache HTML-Tags verwendet werden, mehr dazu bei den Fragen 7 und 8.

6 Wie kann ich überzählige Frage löschen?

Für „normale“ Fragendateien *questions.js* ist es direkt im Quizeditor möglich, die aktuell angezeigte Frage zu löschen.

Es können einfach alle überzähligen Textblöcke (Schreibweise vgl. Frage 5) gelöscht werden, jedoch ist darauf zu achten, dass alle Fragen bis auf die letzte von einem Komma abgeschlossen werden. Bei der letzten darf an dieser Stelle kein Komma stehen.

7 Wie kann ich Bilddateien aus dem Internet in mein Quiz einbinden? [Achtung: Dies erfordert grundlegende HTML-Kenntnisse!]

Die Eingabe von HTML ist bereits im Quizeditor selbst möglich. Komfortabler geschieht dies aber nach dem Export in einer gespeicherten `questions.js` oder einer HTML-Datei unter Verwendung eines einfachen Texteditors (vgl. Frage 5).

Falls man die folgenden HTML-Beispiele direkt im Quizeditor ausprobieren möchte, so muss von den eingeschlossenen doppelten Hochkommata den führenden Backslash weglassen, da dieser beim Export automatisch hinzugefügt wird.

Eine typische Frage besteht dabei aus einem solchen Block:

```
{
  "answer": "b",
  "support": "Gib hier den Erklärungstext ein, der nach einem Quizdurchlauf
bei falscher Beantwortung dieser Frage angezeigt wird.",
  "question": "Gib hier die Fragestellung ein, für die nur eine der unten
stehenden vier Antwortmöglichkeiten richtig ist.",
  "a": "Antwortmöglichkeit A",
  "b": "Antwortmöglichkeit B; diese ist in diesem Beispiel durch die Auswahl
auf der linken Seite als richtig gekennzeichnet",
  "c": "Antwortmöglichkeit C",
  "d": "Antwortmöglichkeit D"
}, // ← Achtung, bei allen außer der letzten Frage muss hier ein Komma stehen!
```

In alle hier hellmagenta gekennzeichneten Bereiche kann **einfaches HTML** eingegeben werden. Für eine Bilddatei z.B. in der Form:

"Welche der folgenden Aussagen trifft auf dieses Bild einer Katze: `` zu?"

Vor jedes **doppelte Hochkomma** muss man jeweils einen **Backslash** schreiben.

Lokal gespeicherte Bilder können natürlich ebenfalls verwendet werden, jedoch muss man dann die Pfade zur Bilddatei entsprechend anpassen.

Variante A (vgl. Frage 2)

Hat man lokal eine Datei `questions.js` in ein Vorlagenarchiv gespeichert und hätte man darin die Bilddatei aus diesem Beispiel direkt in den Ordner **quiz** gespeichert in der Form:

```
quiz
| - daten
|   | - css
|   | - fonts
|   | - js
|   | - questions.js
|   | ...
| - index.html
| - EK_Huggy.jpg
```

So würde sich der **Pfad** ändern in:

"Welche der folgenden Aussagen trifft auf dieses Bild einer Katze: `` zu?"

Der Doppelpunkt bedeutet das Wechseln in das übergeordnete Verzeichnis also erst aus *js* in *daten* und dann von *daten* in *quiz*. Darin liegt dann die Datei *EK_Huggy.jpg*.

Variante B (vgl. Frage 2)

Hat man lokal eine HTML-Datei *mein_quiz.html* in Moodle hochgeladen, so muss man auch das Bild zuerst einzeln hochladen und dann dessen URL direkt angeben.

Diese erfährt man in älteren MOODLE-Installationen (<=1.9) dort im Datei-Backend beim Überfahren der Datei oder ggfs. auch über die Dateieigenschaften.

"Welche der folgenden Aussagen trifft auf dieses Bild einer Katze: `` zu?"

Hierbei wäre **yourdomain** die Adresse der MOODLE-Instanz und **1234** die jeweilige Kurs-ID.

Bei neueren MOODLE-Installationen (>2.0) ist das Bild über seine eigene ID ansprechbar in der Form:

"Welche der folgenden Aussagen trifft auf dieses Bild einer Katze: `` zu?"

Hierbei wäre **yourdomain** die Adresse der MOODLE-Instanz und **1234** die spezielle Datei-ID.

Um sich solche Verrenkungen zu ersparen, können Fortgeschrittene alternativ auch eine PNG-Datei als [DataURI](#) direkt in die HTML-Datei mit hineinschreiben. Dafür könnte man z.B. einen online [Base64-Encoder](#) benutzen:

quizdidaktik.de/bild2html/index.html

Dem Source-Attribut des Image-Tags wird dann in der folgenden Form der erzeugte, je nach Bildgröße sehr lange **Textstring** zugewiesen:

```

```

8 Wie kann ich meine Fragentexte und die Lösungsmöglichkeiten formatieren (z.B. mit Fett- oder Kursivdruck hervorheben)? [Achtung: Dies erfordert grundlegende HTML-Kenntnisse!]

Die Eingabe von HTML ist bereits im Quizeditor selbst möglich. Komfortabler geschieht dies aber nach dem Export in einer gespeicherten `questions.js` oder einer HTML-Datei unter Verwendung eines einfachen Texteditors (vgl. Frage 5).

Falls man die folgenden HTML-Beispiele direkt im Quizeditor ausprobieren möchte, so muss von den eingeschlossenen doppelten Hochkommata den führenden Backslash weglassen, da dieser beim Export automatisch hinzugefügt wird.

Eine typische Frage besteht dabei aus einem solchen Block:

```
{
  "answer": "b",
  "support": "Gib hier den Erklärungstext ein, der nach einem Quizdurchlauf
bei falscher Beantwortung dieser Frage angezeigt wird.",
  "question": "Gib hier die Fragestellung ein, für die nur eine der unten
stehenden vier Antwortmöglichkeiten richtig ist.",
  "a": "Antwortmöglichkeit A",
  "b": "Antwortmöglichkeit B; diese ist in diesem Beispiel durch die Auswahl
auf der linken Seite als richtig gekennzeichnet",
  "c": "Antwortmöglichkeit C",
  "d": "Antwortmöglichkeit D"
}, // ← Achtung, bei allen außer der letzten Frage muss hier ein Komma stehen!
```

In alle hier hellmagenta gekennzeichneten Bereiche kann **einfaches HTML** eingegeben werden.

Für **Fettdruck** z.B. in der Form:

"Hier steht erst ganz normaler Text, gefolgt von einem **fettgedruckten Fachbegriff**."

Für *Kursivdruck* z.B. in der Form:

"Hier steht erst ganz normaler Text, gefolgt von einem *<i>kursivgedruckten Fachbegriff</i>*."

Für Unterschreitungen, Hintergrundfarben oder andere komplexere Style-Angaben, muss man wieder vor jedes **doppelte Hochkomma** jeweils einen **Backslash** schreiben.

"Hier steht erst ganz normaler Text, gefolgt von einem **Fachbegriff mit Unterstreichung**."

9 Hilfe – Mein erzeugtes Programm läuft nicht, wie kann ich es „reparieren“?

- Falls nur einzelne Zeichen wie z.B. Umlaute falsch dargestellt werden, so kann ein erneutes Abspeichern der Datei unter UTF-8 helfen (vgl. Frage 5).
- Falls eigenes HTML im Quizeditor direkt mit eingegeben wurde, so darf vor einem eingebundenen doppelten Hochkomma kein führender Backslash stehen. Falls man jedoch HTML in den Textdaten einer questions.js Datei oder einer exportierten HTML-Datei eingibt, so muss dort der Backslash immer vorangestellt werden.

Schritt 1: Kontrolle: Bin ich überhaupt online? Kann ich z.B. auf anderen Webseiten ganz normal surfen?

Das Quiz läuft nur dann ab, wenn man online ist!*

*In der HTML-Einzeldateiversion wird jQuery dynamisch aus dem Internet nachgeladen.

Schritt 2 für **Variante A** (vgl. Frage 2):

Befindet sich die Fragendatei mit dem korrekten Dateinamen *questions.js* im korrekten Vorlagenordner **quiz/daten/js/** ?

Kontrolliere, ob der Speicherort und der Dateiname stimmen!

Wurde das Vorlagenarchiv korrekt entpackt?

Schritt 2 für **Variante B** (vgl. Frage 2) und Schritt 3 für **Variante A**:

- Firefox-Benutzer:

Öffne die Datei `index.html` (Variante A) bzw. die Datei `mein_quiz.html` mit **Firefox** und öffne unter „Extras“ > „Web-Entwickler“ > „Fehlerkonsole“ (auch mit Strg+Umschalt+J) die Fehlerkonsole.

Wähle im neu erscheinenden Fenster Fehlerkonsole die Ansicht Fehler und verfolge die unten aufgelisteten und beim Laden der Seite dann neu aktualisierten Fehlermeldungen. (Diese können für eine bessere Übersicht zwischendurch auf gelöscht werden, wenn man die Seite dann neu aktualisiert.)

- Chrome-Benutzer:

Öffne die Datei `index.html` (Variante A) bzw. die Datei `mein_quiz.html` mit **Chrome** und öffne unter „Tools“ > „JavaScript-Konsole“ (auch mit Strg+Umschalt+J) die Console.

Lade nun die Seite neu und verfolge die Meldungen. Durch Anklicken der angezeigten Zeilennummern der vermutlich fehlerhaften Zeilen kann man so schnell zur entsprechenden Stelle in der jeweiligen Skriptdatei gelangen.