

WebTech-Probeklausur

Vorbereitung:

- Laden Sie **Klausur1.zip** aus Moodle herunter und entpacken Sie das Verzeichnis in Ihrem workspace/Document Root
- Der Ordner enthält:
 - o Eine Datei **klausur1.php**
 - o Einen Ordner **sources**, welcher eine Datei **text.txt** enthält
 - o (und eine originalText.txt, falls Sie text.txt wiederherstellen wollen)
- Sollten Sie php verwenden, dann erstellen Sie am besten alles innerhalb der Datei **klausur1.php** (auch die JavaScript-Skripte und die CSS-Styles) und laden nur diese am Ende hoch; ansonsten zippen Sie Ihr Projekt und laden es dann hoch

Aufgabe: Erstellen Sie folgende Webseite:

WebTech-Klausur

Datei laden

Suche

Original:

Kopie :

Kopieren

Text speichern

Suchbegriffe :

Jörn Freiheit

Abbildung 1: Anzeige am Desktop-PC

HTML+CSS:

- 2-Spalten-Layout (für Viewport-Breite größer als 768px) mit Kopf- und Fußzeile
- Kopfzeile hellgrauer (#f5f5f5) Hintergrund über die gesamte Breite, Überschrift WebTech-Klausur zentriert; Überschrift fett und in dunkelgrau (darkgrey, #A9A9A9)
- Fußzeile hellgrauer (#f5f5f5) Hintergrund über die gesamte Breite; Link „Ihr Name“ (Link auf die Landingpage für Ihre Aufgaben), zentriert, Link nicht unterstrichen, Schriftfarbe bleibt immer gleich (dunkelgrau (darkgrey, #A9A9A9) egal, ob Link angeklickt wurde oder nicht)

- „Datei laden“ ist ein Hyperlink, der wie ein Button aussieht (**Tipp:** Klassen `.btn`, `.btn-info` und `.btn-block` von Bootstrap verwenden)
- „Original“ ist ein mehrzeiliges Textfeld mit Label (es soll 15 Zeilen anzeigen)
- „Kopieren“ ist ein Submit-Button
- „Text speichern“ ist ebenfalls ein Submit-Button
- Label „Original“, das mehrzeilige Textfeld und die beiden Submit-Button gehören zu einem Formular
- Unter dem Wort „Suchbegriffe“ ist ein nicht sichtbares `<div>`
- „Suche“ ist ein einzeliges Eingabefeld
- unter dem Wort „Kopie“ ist ein `<div>` mit hellgrauem Untergrund
- Wird Ihr Browser auf eine Breite unter 768px geschoben, ergibt sich folgendes 1-Spalten-Layout (**Tipp:** wenn Sie das Bootstrap-Grid-System verwenden, beziehen sich die `-xs`-Klassen auf Viewport-Breiten unter 768px, die anderen Klassen `-sm`-, `-md`-, ... auf breitere Viewports):

WebTech-Klausur

[Datei laden](#)

Suche

Original:

[Kopieren](#) [Text speichern](#)

Kopie :

Suchbegriffe :

Jörn Freiheit

Abbildung 2: Anzeige bei Viewport-Breite kleiner als 768px

PHP:

- Der Hyperlink „Datei laden“ ruft das Script „klausur1.php“ wieder auf. Nach Aufruf des Hyperlinks wird die Datei „sources/text.txt“ in das mehrzeilige Textfeld „Original“ eingelesen → es erscheint der Text aus der Datei im Textfeld
- Das Formular wird durch Drücken eines der beiden Submit-Buttons abgesendet. Die Daten werden per POST-Methode übertragen. Geben Sie den Buttons mithilfe des

„name“-Attributs verschiedene Namen (z.B. 'copy' und 'save') – je nachdem durch welchen Button das Formular abgesendet wurde, ist nur der entsprechende Eintrag im POST-Array vorhanden (z.B. entweder `isset($_POST['copy'])` oder `isset($_POST['save'])`)

- „Datei speichern“: wird das Formular durch Drücken dieses Buttons abgesendet, wird der Inhalt des mehrzeiligen Textfeldes in die Datei „sources/text.txt“ geschrieben → der alte Inhalt der Datei wird überschrieben
Achten Sie darauf, dass die Datei Schreibrechte besitzt!
Der gespeicherte Text soll auch nach dem Speichern noch im Textfeld zu sehen sein!
- „Kopieren“: wird das Formular durch Drücken dieses Buttons abgesendet, wird der Text aus dem mehrzeiligen Textfeld in das „Kopie“-div kopiert:

Abbildung 3: Anzeige nach Drücken des Buttons "Kopie"

Tipps für das Kopieren:

- später (siehe JavaScript) werden Buchstabenfolgen in das obere Suchfeld eingegeben und alle Wörter, die mit diesen Buchstabenfolgen beginnen, sollen gelb markiert werden
- damit das möglich ist, müssten Sie für jedes Wort die Hintergrundfarbe einzeln setzen können
- dazu wäre es sinnvoll, wenn Sie jedes Wort z.B. in ein eigenes ``-Element einbetten würden (wenn Sie einen Text `$text` haben, dann entsteht durch `explode(" ", $text)` ein Array aus den einzelnen Wörtern des Textes – Leerzeichen ist der Trenner) → weisen Sie diesem Span am besten auch gleich eine Klasse für eine Hintergrundfarbe zu

JavaScript:

- behandeln Sie ein Ereignis für die Eingabe im Suchfeld derart, dass nach jedem eingegebenen Zeichen die Wörter im Kopie-Feld gelb markiert sind, die mit der eingegebenen Zeichenkette übereinstimmen, z.B.:

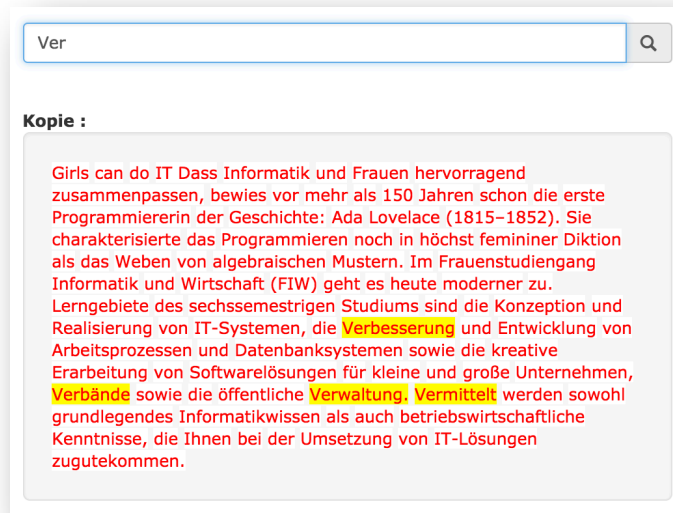


Abbildung 4: Markierte Wörter nach Eingabe von "Ver"

- jede Zeicheneingabe soll sofort behandelt werden
- wird die Enter-Taste im Suchfeld gedrückt, so sollen unterhalb des Wortes „Suchbegriffe:“ die durch die Eingabe markierten Wörter in Form eines Buttons erscheinen

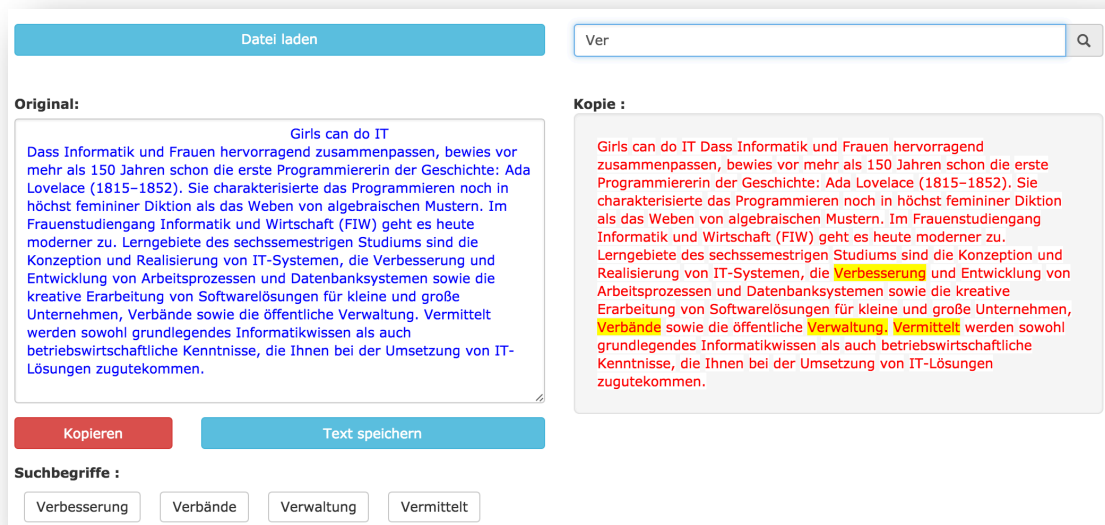


Abbildung 5: Nach Eingabe von "Ver" und Drücken der Enter-Taste

Tipp:

- dem Aufruf einer Funktion wird als erster Parameter das Ereignis übergeben. Mit `event.keyCode == 13` können Sie ermitteln, ob die Taste die Enter-Taste war