

- **1. Übung HTML**

-

Webserver

Auch wenn wir für HTML und CSS noch keinen Webserver benötigen, so ist es ratsam, bereits jetzt einen Webserver zu verwenden, um später keine aufwändigen Anpassungen durchführen zu müssen.

Erstens sollten Sie sich einen Webserver lokal auf Ihrem Rechner einrichten.

- Mac: z.B. <https://coolestguidesontheplanet.com/get-apache-mysql-php-and-phpmyadmin-working-on-macos-sierra/> (zunächst Webserver und Document Root)
- Windows: z.B. <http://www.webmasterpro.de/coding/article/werkzeuge-xampp-lokaler-testserver-einrichtung-und-verwendung.html> (MySQL und phpMyadmin muss noch nicht)
- Linux: z.B. <https://wiki.ubuntuusers.de/Apache/>

→ Wenn Sie im Browser <http://localhost> eingeben, dann sollte so etwas wie „it works“ erscheinen!

Document Root

Unter Document Root verstehen wir den Ordner, in dem die Webseiten für Ihre Domäne gespeichert sind (dort, wo Ihr Webserver nach Ihren Webdokumenten sucht). Unter Windows kann das der **htdocs**-Ordner unter **xampp** sein und im Mac der Ordner **/Library/WebServer/Documents**. Es ist – zumindestens auf dem Mac ratsam, Ihr Document Root zu ändern, z.B. nach **/Users/ihrNutzername/Sites**

Auf dem studi-Server steht ein Webserver zur Verfügung. Ihr Document Root dort ist Ihr privates **public_html**-Verzeichnis. Sollten Sie noch kein **public_html**-verzeichnis haben, müssen Sie sich dieses zunächst erstellen.

A. Git einrichten

Damit Sie lokal auf Ihrem privaten Rechner entwickeln und testen und anschließend Ihre Lösungen einfach in das **public_html**-Verzeichnis des Fachbereichnetzes portieren können, empfiehlt sich der Einsatz von git (siehe z.B. <https://www.atlassian.com/git/> und/oder <https://studi.f4.htw-berlin.de/www/help/tutorial/git/>).

Einrichten von git (die Ordnernamen können Sie natürlich frei wählen – auch gleich benennen; unterschiedliche Ordner sind im folgenden extra unterschiedlich benannt):

1. Loggen Sie sich unter dem **studi.f4.htw-berlin.de** Server ein und erstellen Sie unter Git ein neues Repository (z.B. **WebTech16**)
2. Klicken Sie auf dieses Repository und merken Sie sich die **Clone-URL**
3. Am Beispiel von Eclipse (diejenigen, die nicht Eclipse nutzen, können dies überspringen; wichtig ist nur, dass Sie am Ende einen Ordner (z.B. **WT16**) in Ihrem **Document Root** haben; das kann z.B. auch Ihr Projekt sein):
 - a. Erstellen Sie ein neues PHP-Projekt, z.B. **WeTe16**
 - b. Rechtsklick auf **WeTe16** → Team → Share Project... → (falls Sie gefragt werden:

Git → Next) → bei **Repository** wählen Sie **Create** → browsen Sie zu Ihrem **Document Root** und erstellen dort einen Ordner (z.B. **WT16**)

- c. Ihr Projekt **WeTe16** im workspace ist nun mit dem Ordner **WT16** im Document Root synchronisiert.
4. Wechseln Sie **in** das Verzeichnis **WT16** im Document-Root im Terminal. Geben Sie im Terminal **git clone <CloneURL>** ein. In dem WT16-Ordner entsteht ein Ordner mit dem Repository-Namen (hier: **WebTech16**). → git clone müssen Sie hier nur einmal machen!!
5. Nun können Sie Ordner und Dateien in Ihrer Entwicklungsumgebung dem Ordner **WebTech16** hinzufügen. Achten Sie darauf, diese Ordner und Dateien immer dem git-Index hinzuzufügen. Das können Sie entweder über Ihre Entwicklungsumgebung (in Eclipse z.B. Team → Add Index) machen oder über die Kommandozeile **git add <Datei/Ordner>**. Viele Entwicklungsumgebungen erkennen auch beim Commit, dass neue Dateien getrackt werden und fragen Sie, ob Sie sie hinzufügen wollen, dann müssen Sie nur Häkchen setzen. Um die Dateien in das Repository auf dem Studi-Server zu schieben, benötigen Sie zunächst ein **git commit -m „Commitnachricht“** und dann ein **git push**. Schauen Sie sich ruhig mit **git status** auch mal den Status Ihres Repositories an.
6. Wenn Sie im Browser die Gitweb-URL Ihres Repositories auf dem Studi-Server aufrufen, können Sie überprüfen, ob Ihr commit+push erfolgreich war.

Ihre Lösungen nach public_html auf den Studi-Server laden

7. Loggen Sie sich auf einem Terminal mit Hilfe von **ssh IhrFB4Account@uranus.f4.htw-berlin.de** auf Uranus ein und wechseln in das **public_html**-Verzeichnis
8. Geben Sie **git clone <CloneURL>** ein. Testen Sie, ob die Synchronisation funktioniert hat, indem Sie die URL **studi.f4.htw-berlin.de/~s05xxxxx/Aufgabe1/aufgabe1.html** aufrufen (evtl ist die URL anders – je nachdem, welche Unterordner im Pfad zur aufgabe1.html sind). → git clone müssen Sie hier nur einmal machen!!
9. **alle weiteren** Änderungen, die Sie commiten (Commit & Push), können im public-html-Verzeichnis über **git pull** „herunterladen“

B. Aufgabe1 (wird noch erweitert):

1. Erstellen Sie in einem **Aufgabe1**-Ordner eine Datei **aufgabe1.html**. Das **body**-Element soll ein **header**-Element, ein **nav**-Element, ein **section**-Element und ein **footer**-Element enthalten.
2. Laden Sie die Datei **mockupdatatable** aus Moodle herunter. Diese Datei enthält eine Tabelle mit allen Teilnehmerinnen der Veranstaltung. Binden Sie den Inhalt der Datei so in Ihren html-Code, dass folgende Seite erscheint:

Teilnehmer_innen WebTech



[ALLE](#) [A-G](#) [H-L](#) [M-R](#) [S-Z](#) [Startseite](#)

Alle Teilnehmer_innen

Vorname	Nachname	E-Mail-Adresse	IP-Adresse
Adam	Anderson	aanderson8@google.fr	118.93.83.157
Susan	Andrews	sandrewsn@google.co.jp	228.214.9.251
Catherine	Andrews	candrewsp@noaa.gov	112.111.87.178
Alan	Bradley	abradley1c@globo.com	229.152.117.127
Anne	Brooks	abrooks16@bravesites.com	243.159.39.234
Russell	Brown	rbrownq@nifty.com	215.38.120.242
Ryan	Burton	rburton18@foxnews.com	159.60.107.14
Roy	Campbell	rcampbell1@geocities.com	237.232.34.20
Russell	Campbell	rcampbell17@eventbrite.com	251.2.92.63
Bonnie	Coleman	bcoleman11@fc2.com	109.150.122.102
Ernest	Coleman	ecoleman15@businessweek.com	213.173.4.7
Richard	Cruz	rcruz7@unc.edu	235.124.23.221
Sean	Cruz	scrutz10@answers.com	92.255.49.227
Rebecca	Cunningham	rcunninghamd@mac.com	65.79.191.52
Margaret	Evans	mevansh@pcworld.com	162.10.86.196
Jeffrey	Ford	jford14@cnet.com	210.216.54.14
Andrea	Gardner	agardnerv@woothemes.com	179.91.0.30
Deborah	George	dgeorge6@furl.net	201.76.47.162

- ← Überschrift im header-Bereich
- ← Bild (images-Ordner neben Aufgabe1)
- ← Links auf Unterseiten (in Aufgabe1/NN)
- ← Überschrift im section-Bereich

← Tabelle im section-Bereich

↙ Ganz unten noch Ihr Name im footer-Bereich

3. Es sollen 4 Unterseiten erstellt werden. Bei Klick auf diese Seiten soll die Tabelle jeweils nur die Teilnehmerinnen enthalten, deren Nachname mit dem entsprechenden Anfangsbuchstaben beginnt. Die Seiten **ag.html**, **hl.html**, **mr.html**, und **sz.html** sollen im Ordner **NN** abgelegt werden, der Unterordner von **Aufgabe1** ist.

4. Achten Sie darauf, dass man von jeder Unterseite auf jede andere Unterseite und auch auf die Hauptseite wechseln können muss.

5. Das Bild, das eingebunden wird, können Sie unter https://www.htw-berlin.de/files/Presse/Corporate_Design/Piktogramme_Studiengaenge/FB4_I_W.jpg herunterladen. Es soll in einen **images**-Ordner gespeichert werden, der in der Ordner-Hierarchie neben dem Aufgabe1-Ordner liegt. Um die Größe des Bildes festzulegen, können Sie mit Hilfe des style-Attributes die Höhe und die Breite bestimmen: **style="width:53px; height:48px"**

Nächste Woche wird Aufgabe1 um CSS erweitert.

Abgabe:

Die Lösung soll unter <http://studi.f4.htw-berlin.de/~s05xxxxxx/Aufgabe1/aufgabe1.html> sichtbar sein (evtl. anderer Pfad). Sehen Sie auch die Informationen zur Abgabe in Moodle.