

Sessions & Cookies, JSON encoding

Teil 1: Session-Management

1) Erweiterung Ü4: Login-Funktion

- a) Wenn der/die UserIn sich erfolgreich eingeloggt hat, ist das Formular nicht mehr zu sehen und der/die UserIn erhält eine entsprechende Meldung.
- b) Speichern Sie den Login-Zustand in der Session. Weiters sieht der/die UserIn nun beim Aufruf der Seite einen Link „Logout“. Wird dieser angeklickt, wird der aktuelle Eintrag des/r Users/In in der Session gelöscht und das Login-Formular ist wieder sichtbar.
 - i) Ebenso sollte die Session automatisch zerstört werden, wenn Sie den Browser schließen – Testen Sie diese verschiedenen Zustände und kontrollieren Sie die Werte in Ihrer Session-Datei am Webserver (Windows-Standard-Ordner: C:\xampp\tmp)
 - ii) Beobachten Sie das Senden und Empfangen von Session-Cookies über Ihre Browsertools (Netzwerk, Anfrage-/Antwort-Header)
- c) Erweitern Sie das Formular um eine Checkbox: „Angemeldet bleiben“. Wurde dieses aktiviert, erhält der/die UserIn eine Cookie, welches ihn automatisch an der Seite anmeldet, unabhängig davon, ob der Browser geschlossen wurde oder nicht.
 - i) Das Cookie läuft nach 10 Minuten ab, testen Sie dies mit kürzeren Zeiträumen.
 - ii) Bei Klick auf Logout, soll das Cookie ebenfalls gelöscht werden (Cookie löschen, Session-Inhalte entfernen → Formular ist wieder sichtbar)
- d) Einschränkung Menü: Der Punkt „Zimmerliste“ (siehe 2) soll nur eingeloggten UserInnen zur Verfügung stehen.

2) Zimmerdaten abrufen

- a) Erweitern Sie Ihr Menü um 2 Einträge: „Neue Raum“ & „Zimmerliste“
- b) Legen Sie eine neue Seite (in sites) an: „neuerRaum.php“. Die Datei nimmt die Daten für ein neues Zimmer entgegen (ID, Titel, qm, Licht Status, Temperatur Status, Hintergrundbild, Beschreibungstext, ...).

- c) Speichern Sie die im Formular eingegebenen Daten in der Session und listen Sie diese unter dem Punkt „Zimmerliste“ als HTML-Tabelle auf.
- i) Hinweis: Sie können im Session-Array auch komplexe Strukturen wie Objekte bzw. weitere (mehrdimensionale) Arrays abspeichern:

```
//Session-Variable als Array initialisieren
$_SESSION[,zimmerliste`] = array();
//Alle Inhalte des GET-Requests an Session-Array hängen
array_push($_SESSION[,zimmerliste `], $_GET);
//Das Auslesen d. Arrays ist nun bspw. via foreach möglich
bzw. erfolgt der Zugriff auf einzelne Variablen über ihre
Position sowie ihren Namen im Array
echo $_SESSION[,zimmerliste `][2][,inputname`];
```

- a) Jeder Eintrag ist mit einem x versehen (als Link). Klickt der/die UserIn darauf, wird der Eintrag aus der Session gelöscht.

Teil 2: PHP-Variablen in JSON-Formate konvertieren

1) JSON encoding

- a) Legen Sie einen neuen Folder in Ihrem Projekt an: „RESTServices“. Darin die Datei: „getPatientData.php“.
- b) Erweitern Sie Ihr Menü um einen Punkt „REST API“ (welcher ebenfalls nur für eingeloggte User/Innen sichtbar ist).
- c) Die Datei getPatientData.php gibt alle Patienteninformationen, welche in der Session gespeichert wurden, im JSON-Format aus.
- d) Konvertieren Sie Ihre Session-Inhalte in das JSON-Format mit dem simplen Aufruf `json_encode($var)` und geben Sie das Ergebnis mittels `echo` aus. Testen Sie das Ergebnis.

```
$myJsonString = json_encode($_SESSION[,patientendaten`]);
echo $myJsonString;
```

Freiwilliges Add-on: Greifen Sie auf die REST-Schnittstelle Ihres/r SitznachbarIn zu und geben Sie die Inhalte aus.

Teil 3: VitalServer – Vorbereitung

1. Inbetriebnahme VitalServer

- a. Starten Sie den VitalServer (siehe Moodle-Download) und öffnen Sie die `index.html`. Konfigurieren Sie bei Bedarf den Port (`start.bat` bzw. `start.sh`)

- b. Überprüfen Sie, ob Daten über das vorgegebene Formular angelegt werden können bzw. in der history verfügbar sind.
- c. FH-intern verfügbar unter (LAN/VPN): <http://aat.technikum-wien.at:8080/>

Allgemeines

1. Außerdem & Kontrolle

- a. Stellen Sie sicher, dass Sie weiterhin valides Markup erzeugen
 - i. Einhalten des HTML-Grundaufbaus, HEAD-BODY-Bereiche
 - ii. richtige Verschachtelung
- b. Validieren Sie Ihr Ergebnis (das im Browser als Quelltext ankommt) erneut (W3C-Validator)

Form der Abgabe

Laden Sie die gesamte Abgabe bis zum Abend vor der nächsten Übungseinheit als zip/rar File im Moodle hoch. Achten Sie darauf, dass ALLE Ressourcen vollständig enthalten sind (CSS-Files, Bilder, ...)

>>> FASTLANE >>>

1. Log-Levels:

- a. Geben Sie Zustandsänderungen und Fehler in Ihrer Applikation auf unterschiedlichen Log-Levels aus.
- b. Dokumentieren Sie auch User-Informationen (IP Adresse, Browser-Informationen, Username) sowie Zeitinformation.
- c. Abgabemodus: Screenshot

2. Log-File

- a. Schreiben Sie Zustandsänderungen und Fehlermeldungen Ihrer Applikation in ein separates Log-File.
- b. Abgabemodus: Screenshot

3. Session-Timeout

- a. Konfigurieren Sie Ihren Webserver dahingehend, dass eine Session automatisch nach 10 Minuten beendet wird.
- b. Abgabemodus: Screenshot

Referenzen

- Sessions: https://www.w3schools.com/php/php_sessions.asp
- Cookies: https://www.w3schools.com/php/php_cookies.asp
- PHP Arrays: https://www.w3schools.com/php/php_arrays.asp
- JSON: <http://php.net/manual/de/ref.json.php>
- ErrorLog: https://www.w3schools.com/php/func_error_log.asp
- LogLevels: <http://php.net/manual/de/function.syslog.php>
- LogFiles: <http://php.net/manual/de/function.syslog.php>
- Session Timeout:
<http://php.net/manual/de/session.configuration.php#ini.session.gc-maxlifetime>