PHP
Auswertung von HTML-Formularen

Auswertung von HTML-Formularen

- Ein typischer Anwendungsfall für PHP-Skripte liegt in der Auswertung von ausgefüllten HTML-Formularen, die via HTTP-GET oder HTTP-POST an den Webserver gesendet wurden.
- In dem nun folgenden Beispiel wird ein einfaches Login-System realisiert.
- Dieses Beispiel zeigt, wie
  - ein Formular den Login einleitet.
  - die Eingabedaten überprüft werden.
- Wichtig: Nutzen Sie dieses Beispiel niemals in einem realen Projekt!
  - Es wurde aus Verständnisgründen stark vereinfacht und wichtige Sicherheitsaspekte nicht berücksichtigt.

Auswertung von HTML-Formularen
formular.html

- Wir verwenden ein einfaches HTML-Formular mit zwei Textfeldern und einem Button:

```
<form action="auswertung.php" method="POST">
    <fieldset>
        <legend>Anmeldeseite</legend>
        <label for="user">Benutzername:</label>
        <input type="text" id="user" name="username" placeholder="Mustermann" />
        <br/>
        <label for="pass">Passwort:</label>
        <input type="password" id="pass" name="password" />
        <br/>
        <input type="submit" value="Absenden" />
    </fieldset>
</form>
```

Auswertung von HTML-Formularen

- Als Zieladresse haben wir im Formular die Datei auswertung.php angegeben. Sobald das Formular abgesendet wird, wird der Client von seinem Browser auf diese Datei weitergeleitet.
- In dieser Datei müssen wir nun unser PHP-Skript realisieren, das die Formulardaten auswertet und prüft, ob die angegebenen Daten tatsächlich stimmen.
- Relevant hierfür sind die name-Attribute im HTML-Formular, also **username** und **password** in unserem Beispiel:

```
<input type="text" id="user" name="username" placeholder="Mustermann" />

<input type="password" id="pass" name="password" />
```

- Das name-Attribut dient uns serverseitig als eindeutige Kennzeichnung für ein bestimmtes input-Element im Formular.

Auswertung von HTML-Formularen

- Das assoziative Array $_GET beinhaltet
  die per HTTP-GET vom Client an den Server gesendeten Daten:
  https://www.php.net/manual/de/reserved.variables.get.php

- Analog dazu gibt es ein assoziatives Array $_POST für Daten,
  die via HTTP-POST übertragen wurden:
  https://www.php.net/manual/de/reserved.variables.post.php

- Das assoziative Array $_REQUEST beinhaltet sowohl per HTTP-GET,
  als auch per HTTP-POST übertragene Daten sowie auch
  die Daten der client-seitigen Cookies:
  https://www.php.net/manual/de/reserved.variables.request.php

Auswertung von HTML-Formularen
auswertung.php

- Da wir unsere Formulardaten via HTTP-POST versenden, greifen wir also mit $_POST[“username“] und $_POST[“password“] auf die Daten zu.

```php
<?php

// Hole Daten von Client
$username = $_POST["username"];
$password = $_POST["password"];

// Prüfe, ob die Daten stimmen
if(!($username === "Mustermann" && $password === "web-hs-ma")) {
    exit("Benutzername oder Passwort ist nicht korrekt!");
} else {
    echo "Herzlich willkommen, " . $username . "!";
}

?>
```

Auswertung von HTML-Formularen

- Unser bisheriges System funktioniert nun wie folgt:

- Nach der Eingabe von Benutzername und Kennwort und dem Klick auf Absenden wird das Formular an die auswertung.php gesendet.
  - Diese antwortet
    - entweder mit „Benutzername oder Passwort ist nicht korrekt“
    - oder mit „Herzlich willkommen Mustermann“.

Auswertung von HTML-Formularen

- Im Fehlerfall erfolgt die Ausgabe mit einem harten exit, das PHP-Skript wird dann sofort unterbrochen.
- Im Erfolgsfall erfolgt die Ausgabe einfach mit einem echo.

- Als nächstes wären folgende Erweiterungen sinnvoll:
  - Die Prüfung von Benutzername und Passwort sollte gegen registrierte Benutzer erfolgen, die in einer Datenbank abgelegt sind. Diese sind natürlich normalerweise nicht hart codiert.
  - Die Antwort sollte in allen Fällen als gültige HTML-Datei zurück gegeben werden.
  - Der erfolgreich eingeloggte Benutzer sollte ab jetzt in einer PHP-Session gespeichert werden mit automatischer Übergabe der Session-ID an den Client, damit dieser bei weiteren Aufrufen automatisch authentifiziert ist bis zu seinem Logout.