

**Auftrag:** Client- und serverseitige Validierung realisieren.

**Hilfsmittel:** Präsentation und Template index.php auf BSCW, Internet

**Sozialform:** Einzelarbeit

**Zeit:** **60 Minuten**

Für die Registrierung eines neuen Users in Ihrer Webapplikation, wurde die Struktur der Datenbank-Tabelle bereits erstellt:

	#	Name	Typ	Kollation	Attribute	Null
<input type="checkbox"/>	1	<b>id</b> 	bigint(20)		UNSIGNED	Nein
<input type="checkbox"/>	2	<b>username</b>	varchar(30)	utf8_unicode_ci		Nein
<input type="checkbox"/>	3	<b>fitstname</b>	varchar(30)	utf8_unicode_ci		Nein
<input type="checkbox"/>	4	<b>lastname</b>	varchar(30)	utf8_unicode_ci		Nein
<input type="checkbox"/>	5	<b>email</b>	varchar(100)	utf8_unicode_ci		Nein
<input type="checkbox"/>	6	<b>password</b>	varchar(255)	utf8_unicode_ci		Nein

**Auftrag:**

1. Speichern Sie das File index.php in Ihrer WebRoot und analysieren Sie die Funktionsweise.
2. Validieren Sie die Benutzereingaben clientseitig anhand der Tabellen-Struktur auf maximale Länge und Pflichteingabe. (siehe TODOS).
3. Validieren Sie die Benutzereingaben serverseitig anhand der Tabellen-Struktur auf maximale Länge und Pflichteingabe. (siehe TODOS).
4. Notieren Sie Ihre Erkenntnisse in Ihrem Lernjournal.

**Zusatzaufgaben:**

1. Validieren Sie die Email-Adresse client- und serverseitig auf Ihre Gültigkeit. (*pattern / preg\_match / filter\_vars*)
2. Validieren Sie den Benutzername client- und serverseitig auf seine Gültigkeit. Wie ein gültiger Benutzername aussieht, können Sie dem Attribut *placeholder* entnehmen. (*pattern / preg\_match*)
3. Validieren Sie das Passwort client- und serverseitig auf seine Gültigkeit. Wie ein gültiges Passwort aussieht, können Sie dem Attribut *placeholder* entnehmen. (*pattern / preg\_match*)
4. Lassen Sie den Benutzer sein Passwort ein zweites Mal eingeben. Prüfen Sie clientseitig, ob die beiden Passwörter übereinstimmen. Informieren Sie den Benutzer wenn die Passwörter nicht übereinstimmen.