|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Visualisering  V3.1 | | | |  |
| **Emne:** | PHP | **Opgave:** | Login funktionalitet |
| **Formål:** | * Jeg kan anvende online dokumentation | | |
| * Jeg kan anvende phps indbyggede funktioner til at | | |
| validere de indtastede data i formularfelter | | |

**Opgave**

Du skal i denne opgave validere de felter, som brugeren indtaster i sine formularfelter – herunder bl.a. sørge for, at der er blevet indtastet noget i påkrævede felter og udskrive en fejlmeddelelse til brugeren, hvis der ikke er indtastet.

1. Du skal arbejde videre på dit CMS website
2. I de næste punkter skal du arbejde på at validere inputtene i din login-formular
3. Følg vejledningen på w3schools i forhold til at validere brugernes input i formularfelterne: <https://www.w3schools.com/php/php_form_validation.asp>
4. I eksemplet her lægges data fra formularens $\_POST array over i selvstændige variabler. Du skal i stedet lægge dem over i $\_SESSION arrayet, for vi skal bruge dem i næste opgave til at sørge for, at brugeren bliver ved med at være logget ind, indtil de logger sig ud igen. Det betyder, at koden  
   $name = test\_input($\_POST["name"]); *// Hvor ”name” skal ændres til navnene på hhv dit brugernavn-felt og password-felt*skal ændres til:  
   $\_SESSION[”name”] = test\_input($\_POST["name"]);
5. Du skal nu teste på, om brugeren har indtastet noget i både brugernavn og password. Dette gør du ved at følge denne vejledning på w3schools: <https://www.w3schools.com/php/php_form_required.asp>   
   (husk, at det samme gør sig gældende som i punkt 4: værdierne fra $\_POST skal lægges over i $\_SESSION og ikke i selvstændige variabler)
6. Herudover skal du også flytte værdien fra din submit knap over i din $\_SESSION variabel. Dette skal kun gøres, hvis både brugernavn OG password er udfyldt. Dette gør du med endnu en if-sætning, hvor du tester på om brugernavn og password felterne ikke er tomme (med !empty)
7. Brug også w3schools HTML-eksempel til at få udskrevet fejlmeddelelsen på brugerens skærm, hvis han/hun ikke har indtastet noget i et af felterne. Overvej hvordan du vil style på teksten, så det fremgår tydeligt, at det er en fejlmeddelelse

Det vigtigste i denne opgave er at få punkt 1 - 7 til at virke, for det skal du bruge i den næste opgave.

Hvis du ikke har så meget tid tilbage, skal du gå videre med opgave 8.

Hvis du har god tid tilbage, kan du fortsætte med næste punkt, hvor du skal validere, om brugeren har indtastet korrekte data i din bruger oprettelses formular. Denne validering er ikke så vigtig i en login formular, da brugeren ikke skal indtaste nye data (brugernavn og password er på dette tidspunkt allerede oprettet i en database, og hvis brugeren indtaster ugyldige eller forkerte tegn, bliver han/hun ikke logget ind, så det sker sjældent). Men i en brugeroprettelses formular er valideringen af de indtastede tegn vigtig, for der skal oprettes nye data i databasen.

1. Følg vejledningen på w3schools til at validere, om brugeren har indtastet korrekte data, når han/hun opretter sig selv som bruger på dit website: <https://www.w3schools.com/php/php_form_url_email.asp>   
   Vejledningen her viser både, hvordan man validerer almindelige data, emails og URLs.