

Laboration 3 – HTML och PHP

Denna text är ett extra stöd till laboration 4, och ger förslag på lösningar till vissa av uppgifterna. Se de här förslagen som just förslag, exempel på hur man kan göra men inte måste göra. Lösningarna är inte heltäckande eller färdiga.

Tips för formuläret

Formuläret ska innehålla tre fält enligt uppgiften (se Stefan Gustavsons instruktion till laborationen): namn på författaren, rubrik på inlägget, samt innehållet i inlägget. Men, gränssnittet behöver ytterligare ett textfält för att mata in lösenord, samt två knappar för att spara inlägget samt för att rensa formuläret. Gränssnittet skapas lämpligen med ett antal divar eller med en tabell.

För att ta emot inmatning från användaren kan input-element användas. Läs mer här:

http://www.w3schools.com/tags/tag_input.asp

Formulär med input-element kan ta emot olika typer (type) av input, till exempel text eller knapptryckningar. Det är viktigt att namnge (name) varje input-element korrekt eftersom PHP sedan kommer att ta emot informationen baserat på HTML-elementens namn. Som vanligt när man namnger variabler och klasser etc så ska man använda bra namn som förklarar funktionen och/eller innehållet eftersom detta underlättar kodningen och felsökningen. Läs mer här:

http://www.w3schools.com/html/html_form_input_types.asp

Läs mer om attribut och formulär här:

http://www.w3schools.com/html/html_form_attributes.asp

Storleken (size) på dessa textfält ska sättas direkt i HTML-koden. Detta kan tyckas strida mot regeln att HTML är innehåll och CSS är utseende, men size sätter storleken till antalet tecken som ska rymmas och visas i textfältet. Size anpassar storleken på fältet efter teckenstorlek etc. Läs mer här:

http://www.w3schools.com/tags/att_input_size.asp

Det går också att ändra textfältens storlek på skärmen (width och height) och utseende i CSS-koden på samma sätt som för andra HTML-element. Om alla input-element av typen text ska ändras lika är detta enkelt löst i CSS-koden, om bara ett element av de tre ska ändras ska det elementet få ett id som sedan ändras i CSS, om två av de tre elementen ska ändras tilldelas dessa element en klass. Det är lite knöligare att skriva denna typ av element i CSS. Läs mer här:

http://www.w3schools.com/css/css_attribute_selectors.asp

Rutan med själva blogginlägget bör var större än de övriga input-elementen eftersom blogginlägget med stor sannolikhet är längre än titel och författarens namn, därför ska textarea användas i stället för input. Storleken på textrutan sätts med cols och row i HTML-koden. Både cols och row är nödvändiga att ange i HTML, saknas definitionen av dem i koden sätts cols till 20 och rows till 3 av browsern. Därför kan en HTML-sida valideras som ok även om dessa inte definierats. Elementets storlek på skärmen kan sedan ändras i CSS med height och width. Cols och rows är beroende på teckenstorleken, radhöjden (line-height) och hur stort utrymme varje tecken har (letter-spacing), till skillnad mot height och width. Läs mer här:

http://www.w3schools.com/tags/tag_textarea.asp

Blanda HTML och PHP

Det är tämligen enkelt att blanda PHP och HTML i en och samma fil. Det viktiga är att man håller koll på alla ' och alla ". Använd genomgående samma sätt att skriva inom en HTML-fil eller en JS-fil, till exempel för att skriva strängar i JS, eller värden, klasser och idn etc i HTML. Rekommenderas är att köra dubbelfnuttar ". Det innebär att ni måste använda enkelfnutt ' i PHP i en del fall. Till exempel om PHP-koden innehåller en print("text som ska printas") och texten som ska printas är HTML-kod som exempelvis är en bild () måste ena paret av dubbelfnuttar bytas mot enkelfnuttar. Jag rekommenderar att ni därför kör genomgående med enkelfnuttar i PHP och dubbelfnuttar i HTML, men ni väljer hur ni gör. Så här kan resultatet se ut:

```
print('');
```

Läs mer här:

http://www.w3schools.com/php/func_string_print.asp

Ett annat sätt att skriva HTML i en PHP-fil är genom att använda echo. Echo skriver ut en eller flera strängar, till exempel:

```
echo "Hello world!";
```

Eller:

```
echo '';
```

Läs mer här:

http://www.w3schools.com/php/func_string_echo.asp

Det är rekommenderat med att skriva PHP-koden i ett HTML-dokument, snarare än tvärt om. Man får exempelvis bättre stöd av editorn för att balansera taggarna i HTML-koden och kontrollera syntaxen. Deklarera HTML-dokumentet som vanligt. I <body> där hela eller delar HTML-innehållet ska alstras av PHP så deklarerar PHP som vanligt med <?php varpå PHP-koden följer och sedan avslutas med ?>. HTML-koden kan innehålla HTML-element före eller efter PHP-koden. Flera omgångar PHP-kod kan användas och blandas med HTML-kod så länge PHP-koden deklarerar och avslutas. Slutligen avslutas </body> och </html> som vanligt.