

HTML og relasjonsdatabaser med PHP

Oppgaveveiledning
Kent Dahl <kentda@pvv.org>

Informasjonsbehandling – Brukersystemer
Orkdal videregående skole
(7. mars 2004)

Innholdsfortegnelse

1. Introduksjon.....	2
1.1. Grupper.....	2
1.2. Opplysninger.....	2
2. Oppsett.....	3
2.1. Surfe til webtjeneren.....	3
2.2. Koble opp mot en delt mappe på Windows-nettverket.....	3
2.3. Lage gruppens egen katalog på webtjeneren.....	3
3. HTML.....	4
3.1. Table, tr, th og td.....	4
3.2. Oppgaver.....	4
3.2.1. Gruppeliste.....	4
3.2.2. Rediger HTML.....	4
3.2.3. Hjemmeside.....	4
4. Redigere PHP.....	5
4.1. Kjapp titt på PHP-kode.....	5
4.2. Oppgave: Redigere PHP-kode.....	5
5. Koble oss til databasen fra PHP.....	6
5.1. Gjøre klar databasen.....	6
5.2. Oppgave 1: PoststedListe.php.....	6
5.3. Oppgave 2: Bratt klatreklubb - prisliste.....	6
6. Motta input fra brukeren.....	7
6.1. Skjemaer.....	7
6.2. Hente inputdata i PHP.....	7
7. Legge poster inn i databasen.....	8
7.1. Oppgave 1: NyttPoststed.php.....	8
7.2. Oppgave 2: Bratt poststed.....	8
7.3. Oppgave 3: Nye Bratt kunder.....	8
8. Vedlegg.....	9
8.1. Kildekode.....	9
8.1.1. enkel.php.....	9
8.1.2. PoststedListe.php.....	10
8.1.3. SkjemaInput.html.....	11
8.1.4. Input.php.....	11
8.1.5. NyttPoststed.php.....	12

Dette dokumentet, og filene, er tilgjengelig elektronisk på:
<http://www.pvv.org/~kentda/fag/follo/>

1. Introduksjon

Dette undervisningsopplegget forsøker å koble skriving av websider (HTML) med relasjonsdatabaser (Microsoft Access), ved hjelp av programmeringsspråket PHP.

1.1. Grupper

Gå sammen i grupper med 2-3 stykk per gruppe. Dere trenger minimum 2 PCer per gruppe. Minst en av dem må kjøre i Windows slik at dere kan bruke Access.

1.2. Opplysninger

Plass for å notere ned opplysninger som endrer seg i timene.

<i>Opplysning</i>	<i>Opprinnelig</i>	<i>Endringer / notater</i>
Gruppens navn		
Gruppekatalog	gruppe1	
Webtjenerens IP	192.168.0.46	
Webtjenerens navn	C13_03	
Passord webtjener	WebPhp	

2. Oppsett

2.1. Surfe til webtjeneren

For å kunne se websidene våre ordentlig, må vi kunne surfe til dem med nettleseren. IP-adressen til webtjeneren er **192.168.0.46**, så det skal være mulig å gå til <http://192.168.0.46/> i nettleseren.

Dersom det ikke fungerer, kan det enten være webtjeneren ikke er slått på (mas på veilederne) eller at innstillingene i nettleseren ikke lar dere surfe til maskiner på lokalnettet.

I Windows kan du endre disse innstillingene i nettleseren Internet Explorer ved å gå på menyen (**Vis->Alternativer for Internett**), gå til arkfanen **Tilkobling** i dialogboksen og klikk på **Avansert** nede i boksen Proxy-server . I den neste dialogboksen legger du til IP-adressen 192.168.0.46 i listen over adresser som ikke skal gå via webproxyen. (**Exceptions**) Det fungerer omtrent tilsvarende i Skolelinux dersom noen ønsker å bruke det.

a) Hva er en IP-adresse?

b) Hva er en (web-)proxy?

2.2. Koble opp mot en delt mappe på Windows-nettverket

Dere skal arbeide opp mot en katalog som ligger på webtjener maskinen. Derfor må dere koble dere opp mot denne. På Windows-nettverket heter webtjeneren **C13_03**, katalogen som deles heter **php** og derfor er `\\C13_03\php\` navnet på ressursen vi vil ha tak i.

Vi kan gå direkte til ressursen i Internet Explorer eller i filbehandleren (**Min Datamaskin**) ved å skrive inn stien til ressursen i adresselinjen. (Merk at skråstrekene går motsatt vei av det de pleier når man surfer til websider! Det er fordi vi nå går til noe på Windowsnettverket og ikke Internett.)

Ressursen har ett passord slik at dere får skrivetilgang og det er **WebPhp** (pass på store og små bokstaver).

2.3. Lage gruppens egen katalog på webtjeneren

Dersom forrige steg gikk greit skal dere nå kunne se en del filer under `\\C13_03\php\`. Kopier katalogen **mal** (høyreklikk, kopier, høyreklikk, lim inn) og gi kopien et unikt navn, f.eks. **gruppe1** hvor dere bytter ut tallet med deres gruppenummer, eller et kort navn på gruppen. **Dere skal bare redigere filer inne i denne katalogen. Pass på å ikke rote bort filene til de andre gruppene!!!**

Husk også å ta **sikkerhetskopi** av deres gruppekatalog med jevne mellom. Kopier gruppekatalogen til deres personlige hjemmekatalog (**H:**) i tilfelle noe går galt på webtjeneren eller noen andre sletter filene deres ved et uhell.

Sjekk deretter at katalogen deres er tilgjengelig via webtjeneren på:

<http://192.168.0.46/php/gruppe1/>

3. HTML

Lenker om HTML: <http://www.orkdal.vgs.no/pt/index.php?name=html-redigering>

3.1. Table, tr, th og td

Tabellstrukturen er en av flere måter å presentere informasjon på en oversiktlig måte i HTML.

```
<html>
  <head>
    <title>Tabell eksempel</title>
  </head>
  <body>
    <h1>Tabell eksempel</h1>
    <table border="1">
      <tr> <th>Produkt</th> <th>Pris</th> </tr>
      <tr> <td>Cola</td> <td>kr. 15,-</td> </tr>
      <tr> <td>Snickers</td> <td>kr. 8,-</td> </tr>
      <tr> <td>Osteloff</td> <td>kr. 25,-</td> </tr>
    </table>
  </body>
</html>
```

Figur 1 Kildekode for eksempel på tabell i HTML

Tabell-elementer:

- TABLE – toppnivå elementet som inneholder hele tabellen.
- TR – “table row” - en rad i tabellen.
- TH - “table header” - en overskrift i tabellen.
- TD - “table data” - en datacelle i tabellen.

3.2. Oppgaver

3.2.1. Gruppeliste

Lag et HTML-dokument med en tabell som inneholder informasjon om medlemmene i gruppen deres. Eksempel på innhold er fornavn, etternavn, fødselsdato, epostadresse, valgfag etc. Husk overskrifter i tabellen.

Legg HTML-dokumentet ut på gruppekatalogen deres (\\C13_03\php\gruppe1), slik at dere kan se websiden på <http://192.168.0.46/php/gruppe1/Gruppeliste.html> eller lignende.

3.2.2. Rediger HTML

Åpne filen \\C13_03\php\gruppe1\index.html i gruppekatalogen deres i en teksteditor (f.eks. Notisblokk) og legg til en lenke som peker til gruppe listen deres. Legg gjerne til flere lenker etter hvert som dere lager nye HTML- eller PHP-dokumenter.

3.2.3. Hjemmeside

Dersom dere har tid til overs, kan dere arbeide videre med hjemmesidene dere laget tidligere i vår. Er det kanskje noe der som hadde gjort seg bedre i en tabell?

4. Redigere PHP

For å kunne hente data i fra databasen må vi ha et programmeringsspråk som hjelper oss. Vi skal benytte PHP. PHP lar oss "innebake" programkode i en HTML-fil. Denne programkoden vil utføres på webtjeneren og resultatet av beregningene koden gjør, vil vises i HTML-filen som kommer opp i nettleseren.

Lenker med informasjon om PHP:

- <http://www.php.net/>
- <http://www.orkdal.vgs.no/pt/index.php?name=php-mysql> (IKT-driftsfag sine sider)

4.1. Kjapp titt på PHP-kode

Åpne filen \\C13_03\php\gruppe1\enkel.php i en teksteditor. (Bytt ut **gruppe1** i stien med deres valgte katalognavn.) Åpne samtidig den samme filen via en nettleser ved å gå til <http://192.168.0.46/php/gruppe1/enkel.php> slik at dere ser hva resultatet blir.

I nettleseren, høyreklikk et sted i websiden og velg **Vis kilde** for å se hvordan HTML-koden ser ut etter at PHP-koden har blitt kjørt. Sammenlign filen dere åpnet via gruppekatalogen og filen dere får opp i nettleseren; PHP-koden er borte og bare resultatet fra kjøringen er igjen.

a) Hvor kjøres PHP-programkoden? På hvilken maskin?

4.2. Oppgave: Redigere PHP-kode

Forsøk å rediger PHP-koden i **enkel.php** i teksteditoren. Endre navnet på gruppen, gruppemedlemmenes navn og hvilken gangetabell som skrives ut. Sjekk i nettleseren om det fungerer ved å trykke på oppdateringsknappen. Husk å lagre **enkel.php** og prøv å holde inne Shift-tasten mens dere trykker på oppdateringsknappen dersom siden ikke ser ut til å bli oppdatert ordentlig.

(Kildekoden finnes i vedlegget på side 9)

5. Koble oss til databasen fra PHP

Vi ønsker nå å koble oss til en Access-database fra PHP og lage dynamisk websider ut fra innholdet. Dere skal her bruke en ferdig database dere har laget tidligere, nemlig **Bratt klatreklubb**.

Eksempelkoden fungerer opp mot en bitteliten testdatabase som allerede ligger i gruppekatalogen deres (dersom dere kopierte fra **mal**).

5.1. Gjøre klar databasen

Databasefilen må ligge på webtjeneren i gruppekatalogen deres. Eksempeldatabasen **PersonDB.mdb** ligger allerede der. Deres egen databasefil for **Bratt klatreklubb** må kopieres inn.

Deretter må man sette opp slik at man kan få tak i databasen ved hjelp av ODBC, (Open DataBase Connectivity) som er en standardisert måte å koble seg opp til relasjonsdatabaser. Dette gjøres direkte på webtjeneren, så be veilederne om hjelp. Du får da et navn på databasen, et brukernavn å koble opp med og et passord. Disse brukes i kallet til **odbc_connect** i PHP-koden. I eksempelet under er navnet på databasen "**Person_gruppe1**", brukernavnet "**gruppe1**" og passordet "**6ru993_1**".

Skriv ned de navn og passord dere får tildelt for databasene:

<i>Database</i>	<i>Databasenavn</i>	<i>Brukernavn</i>	<i>Passord</i>
PersonDB.mdb			
Bratt klatreklubb			

5.2. Oppgave 1: PoststedListe.php

PoststedListe.php er et kort eksempel som kobler seg opp til **PersonDB.mdb** databasen og henter ut postnumre og stedsnavn fra en databasetabell **Poststed** og skriver det ut i en HTML-tabell. (Kildekoden finnes i vedlegget på side 10)

Oppgaven: Sett opp databasen på webtjeneren (be om hjelp av veileder) og rediger så deres kopi av **PoststedListe.php** med det databasenavn, brukernavn og passord dere får. Surf til <http://192.168.0.46/php/gruppe1/PoststedListe.php> for å se om det fungerer. Åpne deretter **PersonDB.mdb** i Access, legg til noen poststeder, lagre og lukk databasen. Trykk så oppdater på websiden. Kommer de nye poststedene opp?

5.3. Oppgave 2: Bratt klatreklubb - prisliste

Bratt klatreklubb ønsker å ha prislisten sin på Internett slik at potensielle kunder kan få opp en liste over aktiviteter, med beskrivelse og priser. For å unngå dobbeltarbeid ønsker Bratt klatreklubb at du lager en PHP-side som henter informasjonen rett fra Access-databasen.

Oppgaven: Kopier databasen til webtjeneren og sett den opp. Skriv deres egen **Prisliste.php** fil som kobler seg til databasen, henter ut aktivitetsinformasjonen og viser den i en webside.

Tips: Ta utgangspunkt i koden fra **PoststedListe.php** som løser et lignende problem.

6. Motta input fra brukeren

En webside som bare kan lese fra en database er litt kjedelig. Derfor er vi interessert i å hente inn data fra brukeren, slik at vi kan legge inn nye poster. For å hente data fra brukeren kan vi bruke HTML sine skjemaer (forms).

6.1. Skjemaer

Se kildekoden til eksempelet **SkjemaInput.html** i vedlegget på side 11. Her bruker vi en samling av spesielle HTML-tagger som lar oss definere inputfelt, valgmuligheter og hvor inputdataene skal sendes.

Oppgave: Åpne **SkjemaInput.html** både i nettleseren og i en teksteditor. Kanskje kan du endre på oppsettet slik at det ser bedre ut og er lettere å bruke?

Tips: Inputfeltene er litt rotete. Hva som vi fikk satt dem opp på rekke og rad, som i et rutenett?

6.2. Hente inputdata i PHP

Når brukeren så har lagt inn data i **SkjemaInput.html** og trykket på Submit-knappen, blir de videresendt til en annen side, **Input.php**, som du finner kildekoden til i vedlegget på side 11.

For å bruke dem til noe fornuftig må vi først hente inputdataene ut i PHP-koden. Til det bruker vi variabelen **\$_REQUEST** som inneholder informasjonen nettleseren sendte til webtjeneren.

Oppgave: Åpne **SkjemaInput.html**, legg inn noen verdier og trykk på Submit-knappen. Prøv med litt forskjellige verdier og se om du forstår hva som skjer i **Input.php**. Prøv å legge inn et nytt inputfelt for mellomnavn.

7. Legge poster inn i databasen

I eksempelet **NyttPoststed.php** (kildekode på side 12) kombineres skjema for input og PHP-koden i en og samme fil. Dersom den mottar et postnummer og poststed, forsøker den å legge til en ny post i tabellen `poststed` i databasen `PersonDB`.

For å legge til 7300 Orkanger, kjører vi f.eks. SQL-kommandoen

```
INSERT INTO Poststed (Postnummer, Poststed) VALUES (7300, 'Orkanger')
```

Merk at vi i PHP-koden bruker punktum (.) for å slå sammen flere tekststrenger når vi lager SQL-koden. Når vi setter inn poststedet brukeren ga inn, må vi huske å sette fnutter (!) på begge sider, da poststedet er et tekstfelt.

7.1. Oppgave 1: **NyttPoststed.php**

Legg til noen poststed via **NyttPoststed.php**. Klikk inn på listen over poststedene (`PoststedListe.php`) for å se at de faktisk har blitt lagt inn i databasen. Forsøk å fremprovosere feil. Hva skjer om du skriver inn et poststed med samme nummer som et eksisterende sted?

7.2. Oppgave 2: **Bratt poststed**

Lag en kopi av **NyttPoststed.php** kalt **BrattNyttPoststed.php** og tilpass denne til **Bratt klatreklubb** databasen deres. Sjekk at den fungerer ved å legge .

7.3. Oppgave 3: **Nye Bratt kunder**

(Vanskelig!)

Lag en PHP-side hvor nye kunder kan registrere seg hos **Bratt klatreklubb**, med navn, adresse og poststed.

Pass spesielt på hvordan dere håndterer kundens poststed. Hva skal skje dersom poststedet ikke eksisterer?

8. Vedlegg

8.1. Kildekode

8.1.1. enkel.php

```
<HTML>
  <HEAD>
    <TITLE>Litt enkel PHP</TITLE>
  </HEAD>
<BODY>
<H1>Litt enkel PHP</H1>
<?php
$gruppenavn = "Gruppe 1";
print "<P>Vi er gruppe $gruppenavn.\n";
$person1 = "Ola Nordmann";
$person2 = "Kari Nordmann";
$person3 = "Per Nordmann";
print "<P>Gruppen består av $person1, $person2 og $person3.\n";
?>
<H2>Gangetabell</H2>
<TABLE BORDER="1">
<TR><TH> x </TH><TH> y </TH><TH> x gange y </TH></TR>
<?php
$gange = 5;
for( $steller=1; $steller <= 10; $steller=$steller+1 )
  {
    $produkt = $steller * $gange;
    print "<TR>";
    print "<TD>$steller</TD>";
    print "<TD>$gange</TD>";
    print "<TD>$produkt</TD>";
    print "</TR>\n";
  }
?>
</TABLE>
</BODY>
</HTML>
```

Figur 2 Kildekode for enkel.php

8.1.2. PoststedListe.php

```
<HTML>
  <HEAD>
    <TITLE>Liste over poststed</TITLE>
  </HEAD>
  <BODY>
  <H1>Liste over poststed</H1>
  <!-- Her følger dynamisk kode som kobler oss til databasen,
        sender en spørring og skriver ut en rad for hvert
        poststed: -->
  <?php
  $databasekobling = odbc_connect( "Person_gruppel",
                                   "gruppel", "6ru993_1" );

  if( !$databasekobling )
  {
    print "Feil under kobling til databasen. ";
  }
  $sporringsqlkode = "SELECT Postnummer, Poststed FROM Poststed";
  $radpeker = odbc_exec( $databasekobling, $sporringsqlkode);
  if( !$radpeker )
  {
    print "Feil under spørring av databasen. ";
  }
  print "<TABLE BORDER=1> \n";
  print "<TR><TH>Postnummer</TH><TH>Poststed</TH></TR> \n";
  while( odbc_fetch_row( $radpeker ) )
  {
    $Postnummer = odbc_result( $radpeker, 1 );
    $Poststed   = odbc_result( $radpeker, 2 );
    print "<TR><TD>$Postnummer</TD><TD>$Poststed</TD></TR> \n";
  }
  print "</TABLE> \n";
  odbc_close($databasekobling);
  ?>
  <!-- Dynamisk kode ferdig og databasekoblingen avsluttet. -->
</BODY>
</HTML>
```

Figur 3 Kildekode for PoststedListe.php

8.1.3. SkjemaInput.html

```
<HTML>
  <HEAD>
    <TITLE>Skjema input</TITLE>
  </HEAD>
  <BODY>
    <H1>Skjema input</H1>
    <!-- et eksempel på et skjema (eller "forms") i HTML -->
    <FORM NAME="SkjemaInput" METHOD=post ACTION="Input.php">
      Vennligst skriv inn:<BR>
      Fornavn: <INPUT NAME="Fornavn" TYPE="text" SIZE="50"> <BR>
      Etternavn: <INPUT NAME="Etternavn" TYPE="text" SIZE="50"> <BR>
      Fødselsår:
      <INPUT NAME="Fodselsar" TYPE="text" SIZE="4" VALUE="1987"> <BR>
      Kjønn:
      <SELECT NAME="Kjonn" TYPE="text">
        <OPTION value="M">Mann</OPTION>
        <OPTION value="K">Kvinne</OPTION>
      </SELECT>
      <INPUT TYPE="submit" VALUE="Send inn data">
    </FORM>
  </BODY>
</HTML>
```

Figur 4 Kildekode for SkjemaInput.html

8.1.4. Input.php

```
<HTML>
  <HEAD>
    <TITLE>Input</TITLE>
  </HEAD>
  <BODY>
    <H1>Input</H1>
    <?php
      $fornavn = $_REQUEST["Fornavn"];
      $etternavn = $_REQUEST["Etternavn"];
      $fodselsar = $_REQUEST["Fodselsar"];
      $kjonn = $_REQUEST["Kjonn"];
      print "<P>Hei, $fornavn $etternavn.\n";
      if($kjonn=="M"){
        print "<P>Heisann kjekken!\n";
      }
      if($kjonn=="K"){
        print "<P>Heisann søta!\n";
      }
      $ar = 2004;
      $alder = $ar - $fodselsar;
      print "<P>Du blir $alder år gammel nå i $ar, ikke sant?\n";
    ?>
  </BODY>
</HTML>
```

Figur 5 Kildekode for Input.php

8.1.5. NyttPoststed.php

```
<HTML>
  <HEAD>
    <TITLE>Nytt poststed</TITLE>
  </HEAD>
  <BODY>
    <H1>Nytt poststed</H1>
<!-- Vi kombinerer lagringen og skjemaet i en og samme fil. -->
<?php
$postnummer = $_REQUEST["Postnummer"];
$poststed   = $_REQUEST["Poststed"];
if( $postnummer && $poststed )
{
  /* Prøv å lagre dersom vi mottok både et postnummer og et poststed */
  print "Fikk inn poststed $postnummer $poststed fra brukeren. \n";
  $databasekobling = odbc_connect( "Person_gruppel", "gruppel", "6ru993_1" );
  if( !$databasekobling ) { print "Feil under kobling til databasen. \n"; }

  $sqlkode = "INSERT INTO Poststed (Postnummer, Poststed) ".
    "VALUES ($postnummer, '$poststed)";
  $resultat = odbc_exec( $databasekobling, $sqlkode );
  if( $resultat )
  {
    print "Poststed ble lagt til i databasen. \n";
  }
  else
  {
    print "Poststed ble <STRONG>ikke</STRONG> lagt til i databasen. \n";
  }
}
?>
<!-- Skriv ut det tomme skjemaet. -->
<FORM NAME="NyttPoststed" METHOD=post ACTION="">
  Vennligst skriv inn:<BR>
  Postnummer: <INPUT NAME="Postnummer" TYPE="text" SIZE="4" VALUE=""> <BR>
  Poststed:   <INPUT NAME="Poststed"   TYPE="text" SIZE="50" VALUE=""> <BR>
  <INPUT TYPE="submit" VALUE="Lagre nytt poststed">
</FORM>
<A HREF="PoststedListe.php">Listen over poststed.</A>
</BODY>
</HTML>
```

Figur 6 Kildekode for NyttPoststed.php