

Nya webbservern Dvwebb.mah.se

Bakgrund:

BIT (Bibliotek och IT) beslutar att ta ner Novell-systemet 28/3 som är en katalogtjänst som styr bland annat alla studenter s.k. hemkataloger på Malmö högskola såväl som att ligga till grund till många stödsystem till verksamheten.

Några av dessa stödsystem som läste från hemkatalogerna kallades homeweb och dvwebb.

När Novell byttes ut mot Microsoft AD (Active Directory) så skapade man på BIT nya s.k. hemkataloger i det nya systemet och utlovade en likvärdig weblösning som kallas Webshare.

Webshare har ni tidigare i kursen ombetts att titta på och den fungerar att köra PHP på och arbeta fram till lab4. (Webbteknologier och databaser delkurs3, Webbprogrammering).

Webshare har dock inte databasstöd och därmed måste vi koppla till en annan databasserver för att lösa lab5.

Tekniker på Teknik och Samhälle har under våren fått i uppdrag att sätta upp en egen TS (Teknik och Samhälle) lösning för både PHP och databasstöd i en och samma server som skulle vara integrerad med högskolans övriga system.

Denna lösning är nu någorlunda färdig och därför uppmanar jag er att testa denna.

Den har fått namnet dvwebb.mah.se (samma som tidigare) men har inget med den gamla dvwebb.mah.se att göra.

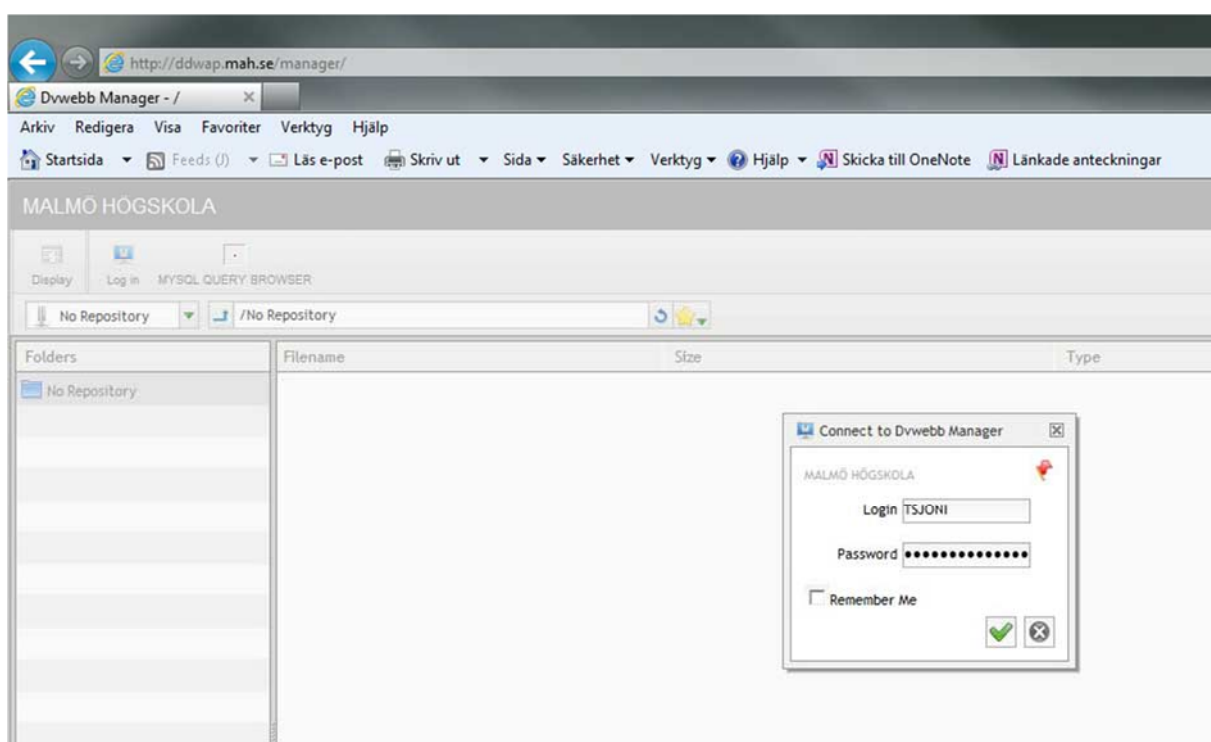
Dvwebb har bland annat stöd för visning av statiska HTML-sidor med JavaScript, CSS osv. och även för PHP och databaser.

På Dvwebb.mah.se skall ni kunna köra hela kursen Webbteknologier och databaser delkurs3, Webbprogrammering.

Steg1. Logga in.

Login: Datorid med versaler.

Password: Lösenord som det är skrivet när ni loggar in i ert distansutbildningsverktyg eller i webmailen (både gemener och versaler).



2. Fillista

När ni loggat in kommer ni till en fillista som specificerar vad ni har lagrat på dvwebb.

För första gången har ni inga filer.

(För att ladda upp filer klicka: upload. Se punkt6 för mer info)

För att titta på innehållet i en fil kan ni klicka på den och få en förhandstitt (längst ner till vänster) när det gäller både bilder och filmer.

Vill ni t.ex. editera en fil kan ni dubbelklicka på den direkt i webbläsaren och få upp en editeringsruta.

Filename	Size	Type	Modified	Dimensions
index.php	867 b	PHP script	2012/03/30 16:41	
MahLogga200.jpg	23,86 Kb	JPG picture	2012/04/03 14:08	200px X 239px
network.png	12,25 Kb	PNG picture	2012/04/03 11:24	526px X 112px
testar_databasen4.php	438 b	PHP script	2012/03/16 21:25	
testar_databasen5.php	1 Kb	PHP script	2012/03/19 12:06	
testar_databasen6.php	1,02 Kb	PHP script	2012/03/29 14:49	
webshare.mp4	6,92 Mb	MPEG File	2012/03/26 20:14	
Recycle Bin	-	Recycle Bin	2012/04/03 14:02	

File Info

Name	MahLogga200.jpg
Size	23,86 Kb
Last Modif.	2012/04/03 14:08
Type	JPG picture
File Meta Data	
Comment	

Source Editor - phpinfo.php

```
<?php\n\nphpinfo();\n\n?>
```

3. Skapa filer på webbservern via Ajaxplorer.

För att skapa en fil högerklicka och väl new file.

Välj filnamn och spara.

Naturligtvis går det även att skapa filerna lokalt

på er dator (klienten) och sen ladda upp dem via Upload)

Jag provar att skapa en fil på webbservern och dubbelklickar sedan på den för att få fram editorn i Ajaxplorer. Den innehåller radnummer och har stöd för syntax-highlighting.

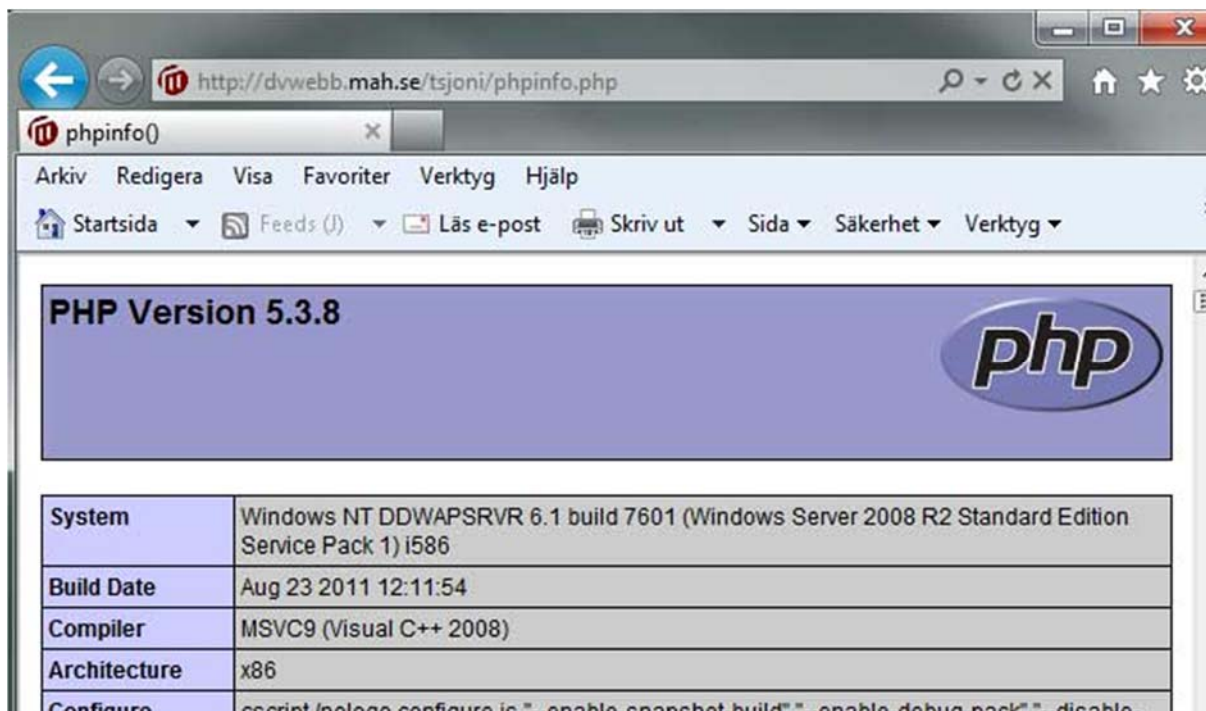
4. Komma åt filer på webbservern via webbläsare

När ni sedan ska nå innehållet i vår fil går ni in via URL:

<http://dvwebb.mah.se/datorid/filnamn.filändelse>

I mitt (John Niubo) fall så blir det

<http://dvwebb.mah.se/tsjoni/phpinfo.php>



Som ni ser så genererar funktionen `phpinfo()`; i min minimala PHP-fil massvis med data om PHP-versionen som är installerad på webservern.

Detta hade lika gärna kunnat vara en enkel statisk HTML-fil som ni laddat upp till webbservern och sedan nått på samma sätt med samma adresseringsmetod.

5. Katalogstrukturer på webservern

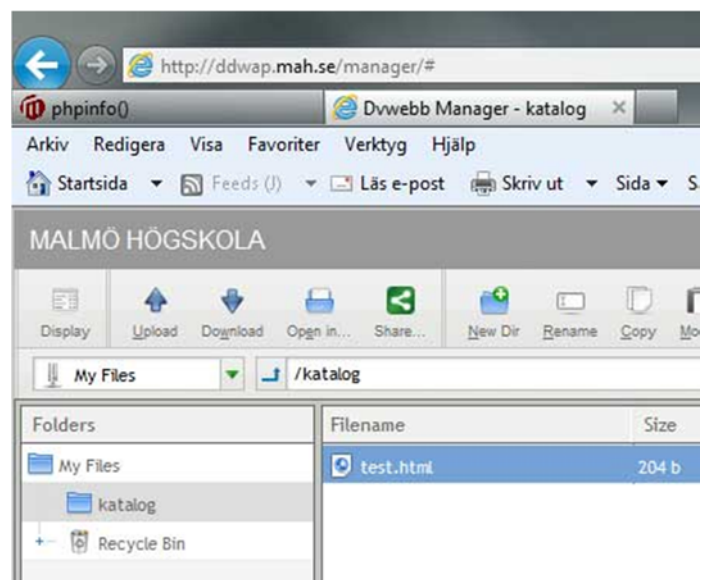
Vill man ha ordning och reda så kan man göra katalogstrukturer i sin del av dvwebb.mah.se

Adresseringen följer detta på följande sätt.

<http://dvwebb.mah.se/datorid/katalognamn/filnamn.html>

I exemplet till höger.

<http://dvwebb.mah.se/tsjoni/katalog/test.html>

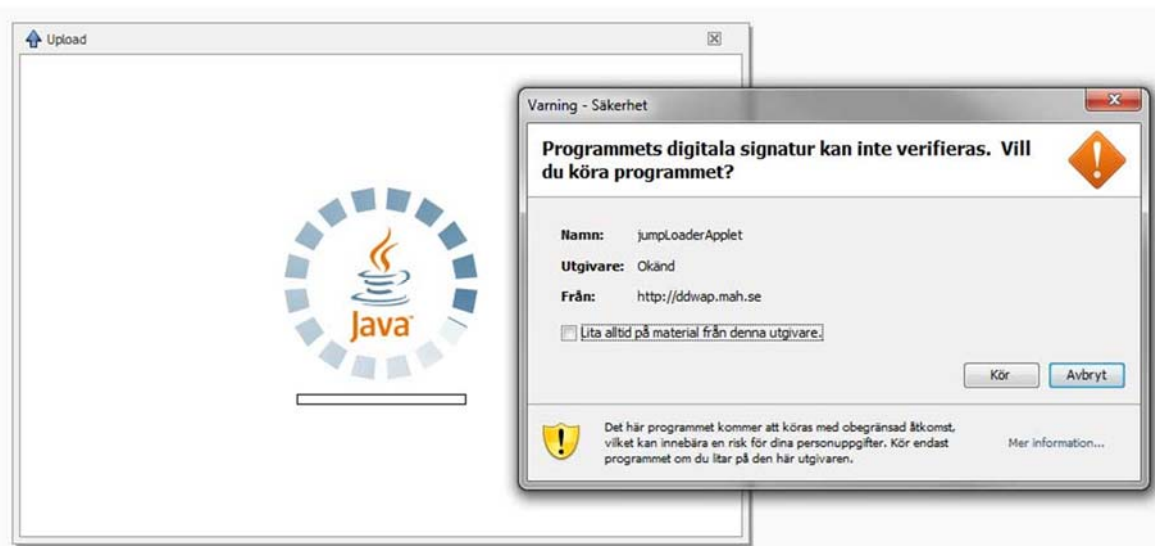


6. Ladda upp filer från klient (er dator) till webbserver.

Här krävs det att ni har "Java Virtual Machine" installerat på er klientdator. Har ni inte Java Virtual Machine kan ni hämta det på <http://www.java.com/sv/download/>

För att ladda upp filer

Klicka på knappen Upload i menyn.

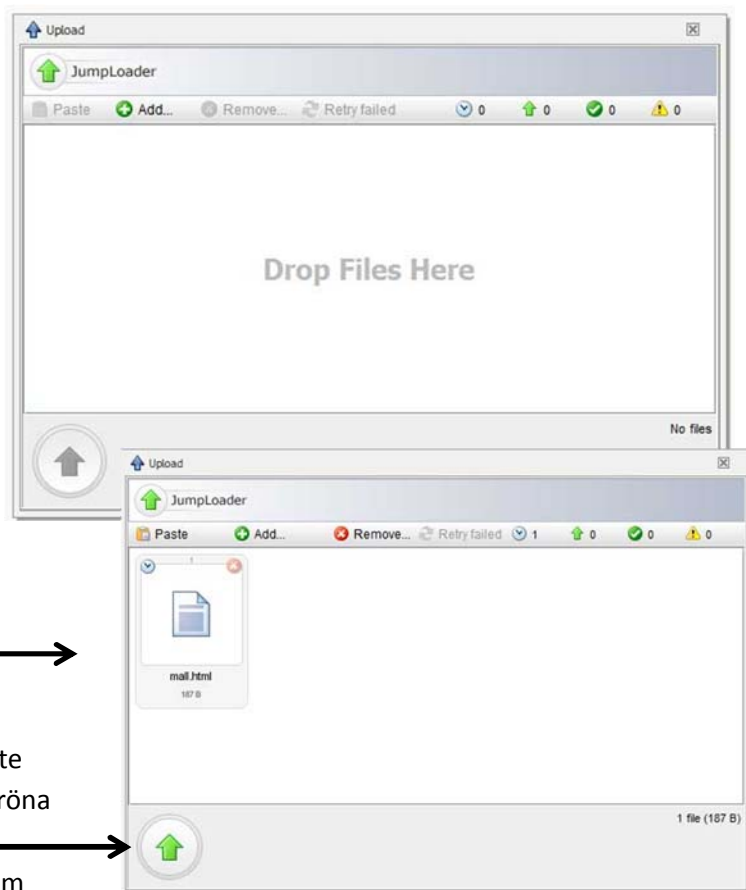


Acceptera programmets digitala signatur från den okända utgivaren på egen risk.

Du får fram en upload-ruta, klicka på Add.

Leta fram filen som ni ska ladda upp i ert filträd (på er dator).

Ni kan använda drag and drop-metoden i detta fönster.



När valt fil/filer ser det ut enligt följande. Men då har **inte** filen laddats upp ännu.

För att den ska laddas upp till webbservern måste ni klicka på ladda upp knappen med den stora gröna pilen längst ner.

Denna ska naturligtvis ha ett namn bredvid såsom Upload, men det finns inte ännu.

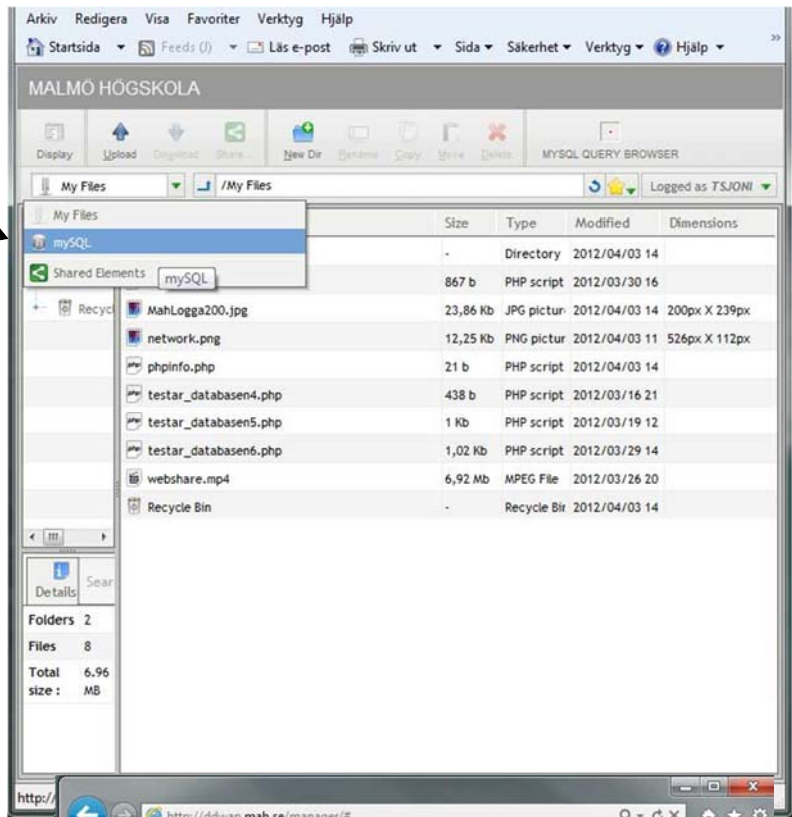
När filen är uppladdad syns den i er fillista (se punkt2) under My Files.

Först när filen är uppladdad kan den nås via en webbläsare genom att skriva korrekt sökväg eller URL.

Om filen hette test.html och laddades upp i My files hade sökvägen blivit <http://dvwebb.mah.se/datorid/test.html>

7. Databaser

Välj MySQL
för att gå in i
databasläge.



Ni har nu gått in i er egen databas
med namnet: DINDATORID (versaler)
För mig (John Niubo) heter min databas
TSJONI.

Vid första anblick ses
eventuella gemensamma tabeller
och sina egna tabeller.

Jag rekommenderar er i detta
läge att gå över till MySQL Query
Browser och skriva in SQL-satser
för att skapa tabeller och
lägga in data.

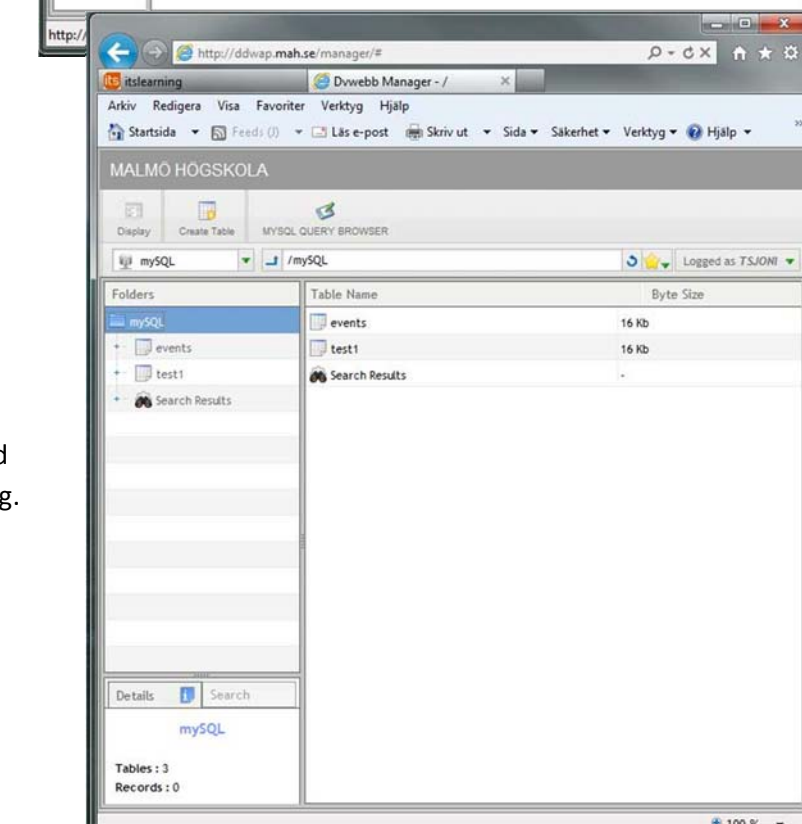
Språket SQL kommer att se ut på samma
sätt under en lång tid framöver till skillnad
från grafiska gränssnitt som ofta ändrar sig.
Det finns dock inget negativt med
att titta på det grafiska gränssnittet
och sen ta sig an MySQL Query Browser.

För att skapa en egen tabell i din
befintliga databas i grafiskt
gränssnitt.

Välj Create Table

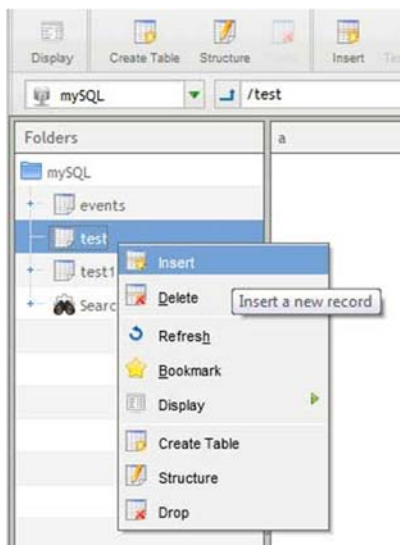
Ni frågas efter namn och antal kolumner i
tabellen.

Notera Can och Nex
för Cancel och Next.
Bugg i programvaran.



Därefter får ni välja namn på fälten (kolumnerna) och variabeltyp samt om dessa fält ska vara primärnycklar osv. Jag väljer att kalla kolumnerna a och b och sätter att den ena ska vara av variabeltypen varchar (tecken) och den andra av typen integer (heltal).

Nu finns min tabell i vänster fält bland mina andra tabeller.



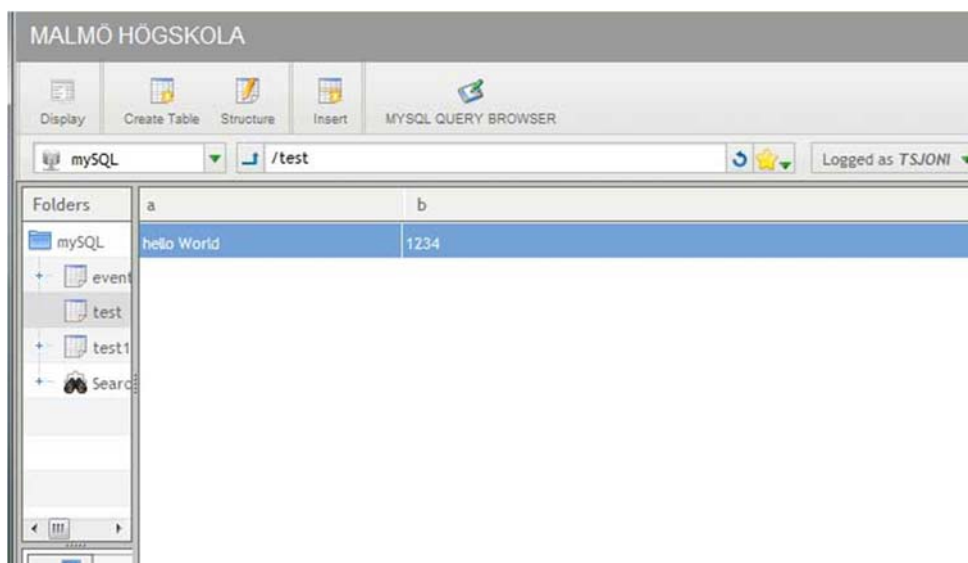
Välj därefter höger musknapp och Insert.

Fyll i några värden. Notera dina valda variabeltyper till höger.



Nu har vi fyllt vår tabell med data och den innehåller två värden, de ligger i en post eller en tupel.

För att se innehållet i tabeller kan man klicka på dem och helt enkelt se värdena i tabellerna.



8. Databaser via MYSQL QUERY BROWSER

Klicka på MYSQL QUERY BROWSER i menyn. Följande andra inloggning uppträder.

Skapar ni webbsystem i framtiden skall ni undvika att användaren ska behöva logga in flera gånger i samma system.

Detta är påpekat av mig som en av sakerna på listan att fixa.

MYSQL Connections:

Engine ☒ MySQL

User Name

Password

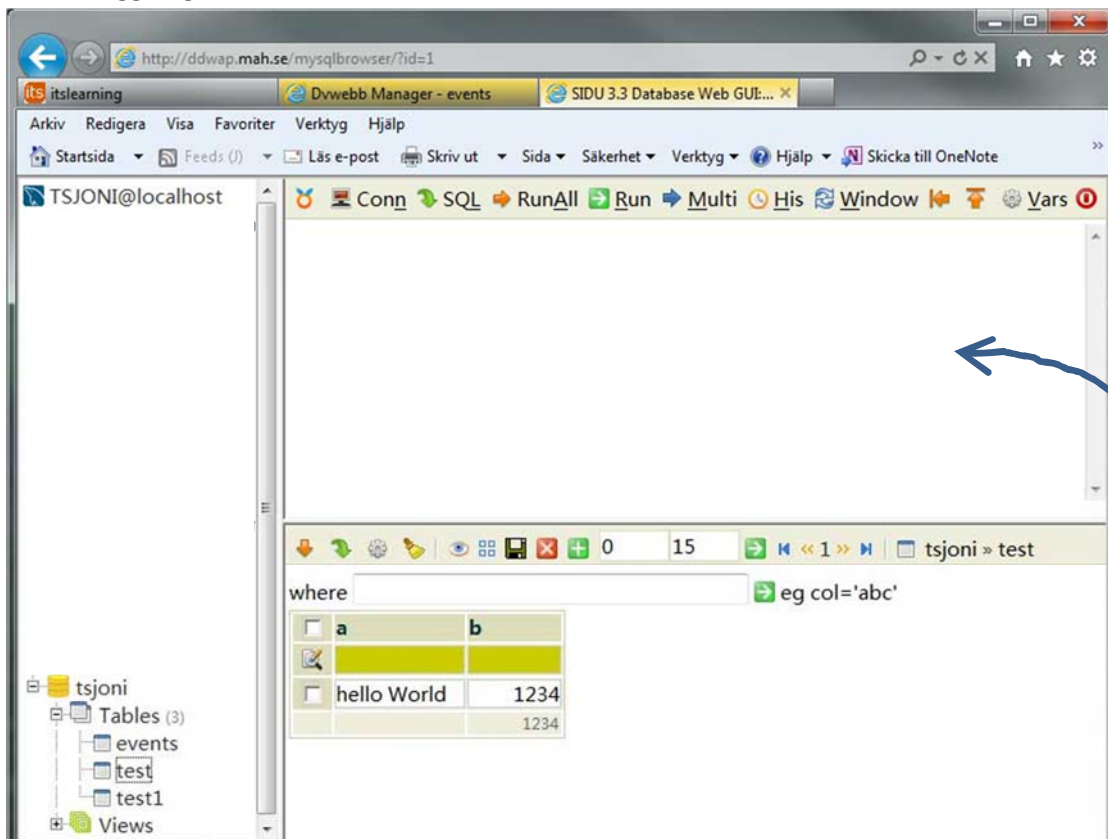
Här måste ni återigen logga in.

User Name: DATORID (Versaler)

Password: Det lösenord ni hade den första gången ni använde systemet. Detta är det samma som ni har i distansutbildnings-verktyget eller i högskolans nätverk. Men om ni ändrar det i högskolans nätverk ändras det inte i databassystemet.

Detta är ett exempel på ofullständig systemintegration som man skall undvika vid uppbyggnad av system. Det finns på min lista över förbättringar i dvwebb.

Efter inloggning kan databasen betraktas..



Samma tabell som jag tidigare skapade (test) finns även här åskådliggjord grafiskt.

Skillnaden här är att vi har en ruta där vi kan skriva SQL-kommando och köra dessa mot databasserver.

Vi kan således skriva ett kommando för att t.ex. läsa ut allt från tabellen test med SQL-kommandot.

```
SELECT * FROM test;
```



Efter att jag klickat Run så körs SQL-kommandot och jag får min utdata i nederstående ruta.

Jag får också information om hur lång tid databasservern tog på sig att köra kommandot.


Men varför är MYSQL QUERY BROWSER så bra?

Vi hade ju samma funktionalitet i det tidigare grafiska gränssnittet som i det senare MYSQL QUERY BROWSER.

Anledningen är att ni kan ställa samma frågor som ni lär er att ställa från PHP-scripten till databaser i denna ruta.

Om ni gör det innan ni ställer frågorna i PHP-koden så kan ni testa att er SQL-kod genererar svar och vilka svar den genererar **innan** ni skriver kod som skickar SQL-kommandot och hanterar utdata från databasen.

Det är mycket användbart och bör utnyttjas i kurserna Webbprogrammering och Databasteknik.

För att logga ut från databasservern klickar man på 

För att koppla upp mot min nygjorda databas via PHP skriver jag:

```
<?php
$user="DATORID";
$password="mitthemligapassword";
$host="localhost"; /* Här kan det endast stå localhost om och endast om webservern
ligger på samma ställe som databaserna vilket de gör nu, annars IP-nummer eller URL*/
$connection=mysql_connect($host, $user, $password);

/*Här bör vi om detta vore en riktig sida lägga DTD osv. för att starta HTML-output korrekt. Detta hoppas över i
detta exempel vilket som följd inte kommer att validera korrekt för HTML. */

//felhantering
if(!$connection){
    echo 'Kunde inte ansluta till databasen';
}
$mydatabase="DATORID"; //Databasen heter samma sak som din datorid med versaler.
$db=mysql_select_db($mydatabase,$connection);
if ($db){
    echo 'nu kom jag in i databasen1234: ';
}
//felhantering
if(!$db){
    echo 'Kunde inte ansluta till min databas';
}
//skapa SQL-fråga för att läsa ut alla inlägg från tabell, denna testas innan.
$query = "SELECT * FROM tabellnamn";

//ställ frågan till databashanteraren
$result = mysql_query($query, $connection);

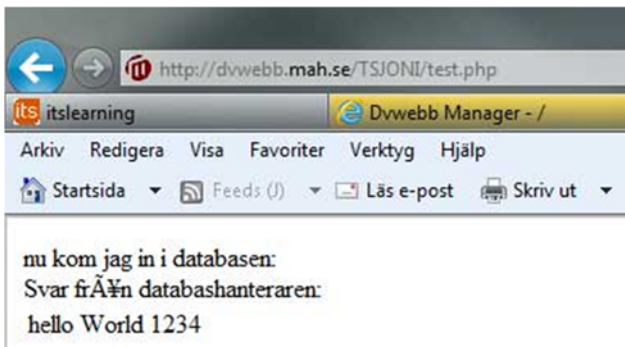
echo 'Svar från databashanteraren:<br>';

/*Hade vi skrivit ut echo ($result); så hade det blivit array utan innehåll. Resultaten lagras alltså i en variabel av
typen array och det måste vi tänka på och kunna hantera vid utskrift.*/

echo '<table>';
while( $row = mysql_fetch_array($result) ) {
    echo '<tr>';
    echo '<td>' . $row['a'] . '</td><td>' . $row['b'] . '</td>';
    echo '</tr>';
}
echo '</table>';

//Här hade vi avslutat body och HTML om vi skulle vilja ha en validerbar sida.
?>
```

Efter att jag sparat detta med mina inställningar får jag följande utdata från min webbläsare.



Som ni märker fungerar inte å utan ersätts med något konstigt.

Detta beror på att jag varit slarvig och inte angivit korrekt utdata-form för webbläsaren. Förutom att webbläsaren inte får reda på vad det är för data jag skickar till den från PHP-servern så får den heller ingen information om hur den ska tolka teckenuppsättningen.

Skickar vi med en korrekt DTD med korrekt teckenuppsättning av formen

```
<meta http-equiv=\"Content-Type\" content=\"text/html; charset=utf-8\">
```

så blir resultatet:



Prova nu att göra din egen tabell och läs ut data från denna.

Gå sedan vidare till uppgift 4 och 5 och lös dem med dina nyvunna kunskaper.

Denna instruktion skall uppdateras med information om hur man ansluter via FileZilla detta kommer att göras de närmaste dagarna.

mvh John Niubo

Mah