

UMEÅ UNIVERSITET

January 8, 2015

Institutionen för Tillämpad fysik och elektronik

Laborationsrapport

Laboration 5

Script och webbrogrammering 7,5 hp

PHP och Databaser

Namn Christer Jakobsson

E-mail dv12c.jn@cs.umu.se

Adress <http://shinowa.tk>

Handledare

Ola Ågren, Kalle Prorok

1 Del 1

1.1 Problembeskrivning

Tanken med denna del av uppgiften är att skriva ett php script som hanterar något som skrivits in i ett formulär, jag ska således skapa en html sida med ett formulär som, när användaren trycker på *Submit* ska hanteras av ett php script som jag också ska skapa. Scriptet ska även hantera standardinformation som tex IP-nummer och tid.

1.2 Användarhandledning

Denna del går att komma åt via:

<http://shinowa.tk/lab5/formWithStandardInfo.html>. Eller genom att först använda länken `shinowa.tk` och sedan klicka under Del 1 *Fristående php: Add Form*.

När man har navigerat till sidan så får man fylla i alla fält:

First Name: Krävs, får bara innehålla stora och små bokstäver samt tecknet -.

Last Name: Krävs, får bara innehålla stora och små bokstäver samt tecknet -.

E-mail: Måste innehålla @ (snabel a)

Password: Krävs, måste vara samma som *Retype Password*

Retype Password: Krävs, måste vara samma som *Password*

Telephone: Krävs, måste vara ett nummer.

Sex: Krävs.

Favorite Color: Krävs.

Om användaren vill börja om med att skriva formuläret så kan denne trycka på *Reset* för att nollställa alla fält. När formuläret är ifyllt så trycker man på *Submit* och kommer då att komma till en sida som innehåller php kod för att hantera datat ifrån formuläret.

Först så valideras varje fält enligt reglerna i listan ovan och om alla fält är korrekt ifyllda så visas värdena på sidan tillsammans med standardinformation såsom ip-nummer, webbläsare samt datum och tid.

1.3 Exempelkörningar

1.3.1 Korrekt ifyllt formulär

Jag skriver in korrekta värden i varje fält och trycker på *Submit*.



Form

Personal Form

First Name: Christer

Last Name: Jakobsson

E-mail: jakobsson.christer@gmail.com

Telephone: 123456

Password:

Retype Password:

Favorite Color: Green

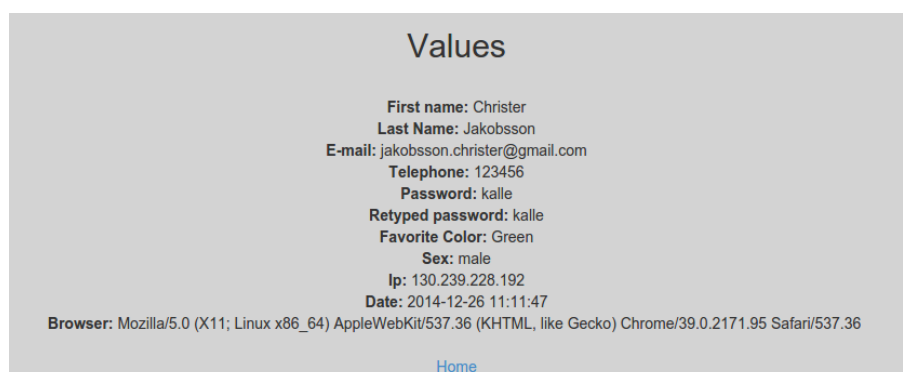
☒ Male ☐ Female

[Reset](#) [Submit](#)

[Home](#)

Figure 1: Formulär med korrekt ifyllda fält

Jag ser resultatet av det jag skrivit in i formuläret. Notera att lösenorden är identiska.



Values

First name: Christer

Last Name: Jakobsson

E-mail: jakobsson.christer@gmail.com

Telephone: 123456

Password: kalle

Retyped password: kalle

Favorite Color: Green

Sex: male

Ip: 130.239.228.192

Date: 2014-12-26 11:11:47

Browser: Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/39.0.2171.95 Safari/537.36

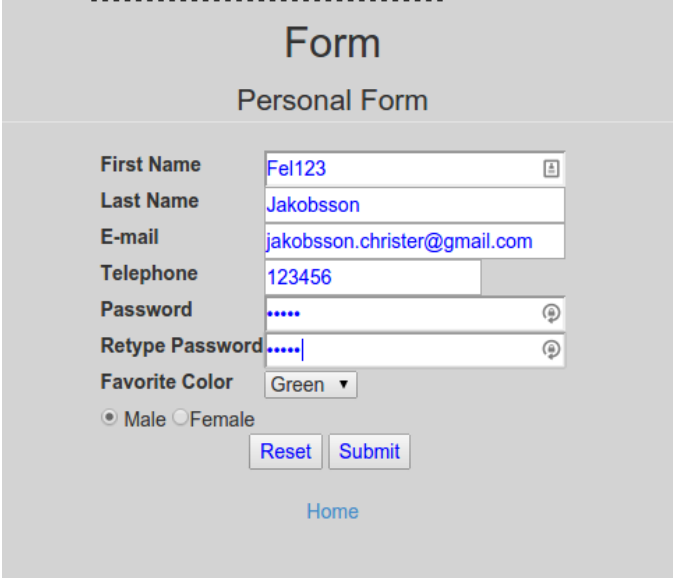
[Home](#)

Figure 2: Resultat efter *Submit*

Kommentar: Fungerar som tänkt, i en riktig version skulle man inte hantera ett lösenord så här men detta är bara till för att jag vill prova password typen i formuläret.

1.3.2 Felaktigt *First Name*

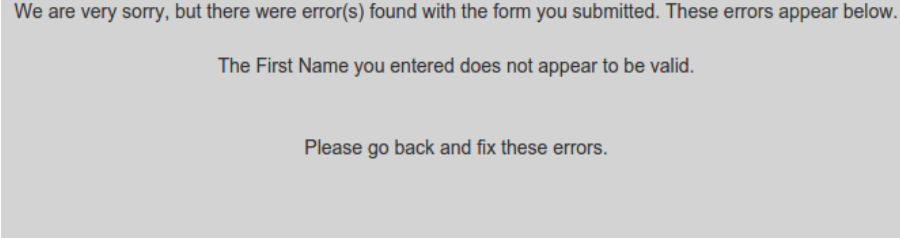
Jag skriver in ett namn med siffror, vilket inte ska godkännas och trycker på *Submit*.



The screenshot shows a web form titled "Form" with the subtitle "Personal Form". It contains several input fields: "First Name" (containing "Fel123"), "Last Name" (containing "Jakobsson"), "E-mail" (containing "jakobsson.christer@gmail.com"), "Telephone" (containing "123456"), "Password" (containing "...."), "Retype Password" (containing "...."), "Favorite Color" (a dropdown menu set to "Green"), and two radio buttons for "Male" and "Female". There are "Reset" and "Submit" buttons at the bottom, and a "Home" link below them.

Figure 3: Felaktigt First Name

Jag ser att detta inte godkändes och att felet var i *First Name*



The screenshot shows an error message box with the following text: "We are very sorry, but there were error(s) found with the form you submitted. These errors appear below." followed by "The First Name you entered does not appear to be valid." and "Please go back and fix these errors."

Figure 4: Resultat efter *Submit*

1.3.3 Två felaktiga fält

Jag skriver in ett förnamn och ett efternamn med siffror, vilket inte ska godkännas, resterande fält är korrekta.

First Name: test123

Last Name: test123

Trycker på *Submit*.

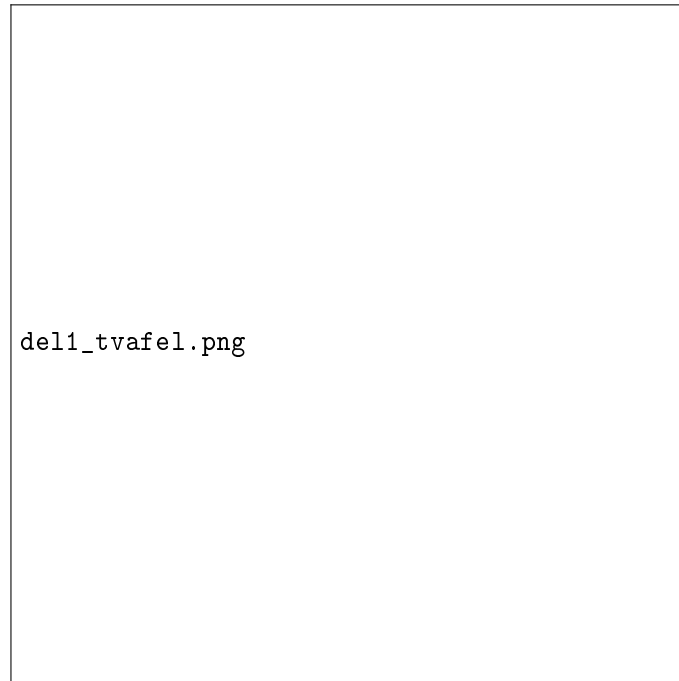


Figure 5: Felaktigt First Name

Kommentar: Här kan vi se att användaren får information om vilka fält som är fel, trots att det är mer än ett fält som är fel.

1.3.4 Felaktigt *E-mail*

Jag skriver in en felaktig E-mail, utan @.

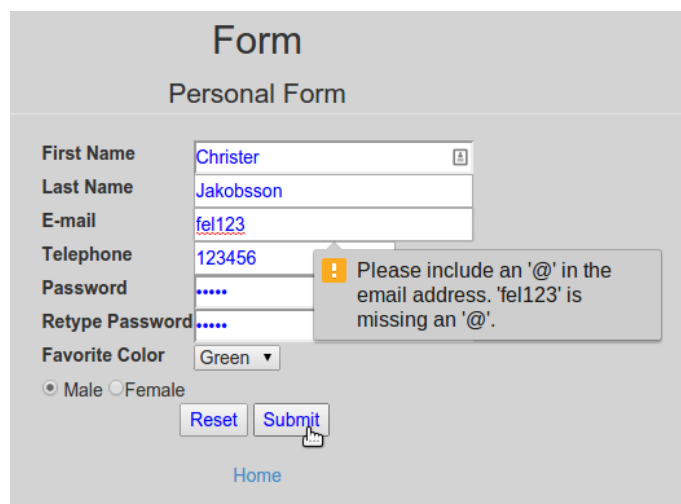


Figure 6: Vy vid klick av submit

1.3.5 Felaktigt telefonnummer

Jag skriver in ett nummer som innehåller bokstäver.

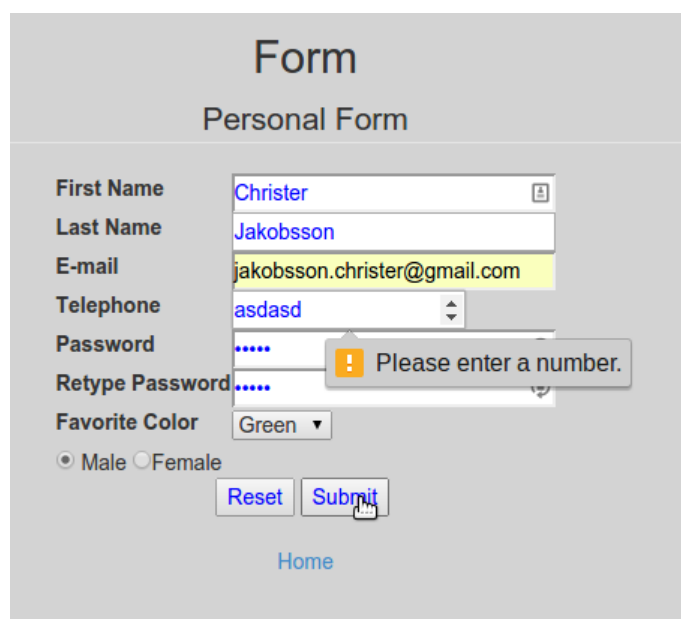


Figure 7: Vy vid klick av submit

1.3.6 Lösenord som inte matchar

Jag skriver in lösen1 i *Password* och lösen2 i *Retype Password* och klickar på *Submit*

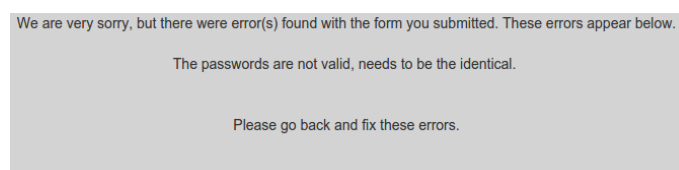


Figure 8: Vy efter Submit

2 Del 2

2.1 Problembeskrivning

Denna del av laborationen har gått ut på att göra ett eller flera skript som presenterar och ändrar något i en databas. Om man har tänkt att göra ett enkelt bokhanteringssystem i projektet så kan man göra skript som är en prototyp till det. Jag använder mig av html och php för att skapa några enkla sidor där användaren kan lägga till en bok, ta bort en bok, ändra en boks fält och en sida där man kan se alla böcker.

2.2 Användarhandledning

Denna del innehåller flera små delar där alla hanterar bokhanteringsdatabas, delen innehåller tre stycken formulär som kan användas för att manipulera databasen. Dessa formulär finns på: <http://shinowa.tk> under rubriken, del 2.

2.2.1 Lägga till bok

Detta formulär liknar hur det fungerade att lägga till en bok i det grafiska gränssnittet i laboration 4. Alla fält förutom *Comments* måste vara ifyllda. *Date Read* måste vara ett datum, och *Grade* måste vara en siffra emellan 1-5. *Genre* och *Genre2* är listboxes med fördefinierade val, *Genre2* ändrar dess värden beroende på vilket värde som är valt i *Genre*.

Användaren fyller i formuläret och klickar på *Add book* för att lägga in det i databasen, i detta steg så valideras fälten och om alla fält är korrekta så läggs det in i databasen. Om något inte är korrekt ifyllt så visas en popup med vilket fält som måste korrigeras för att kunna lägga in bokdata't.

Regler för fält:

Author: Krävs, får innehålla A-Z, a-z

Title: Krävs, får innehålla alla tecken

Date Read: Krävs, måste vara i ett datum format.

Genre: Krävs, Listbox

Genre2: Krävs, Listbox

Grade: Krävs, Listbox

Comments: Valfri, kan innehålla alla tecken

Detta formulär finns på: http://shinowa.tk/lab5/book_addForm.html

2.2.2 Ta bort bok

Detta formulär används för att ta bort en bok ifrån databasen, användaren får skriva in titeln på den bok som ska tas bort och sedan klicka på *Remove Title*. Sedan så visas en resultatsida där man kan se om titeln har tagits bort, om ingen titel tagits bort så finns ingen bok i databasen med den titeln.

För att veta vilka böcker som finns i databasen så kan man gå till: <http://shinowa.tk/bookData/bookData.php> som visar databasens innehåll.

Detta formulär finns på: http://shinowa.tk/lab5/book_deleteForm.html

2.2.3 Ändra bokdata

Detta formulär används för att ändra en befintlig boks data. Användaren får skriva in titeln på boken som ska ändras. Om den inte finns så skrivs ett felmeddelande ut och användaren får försöka igen, om titeln hittades så tas all data fram om titeln och läggs in i fält. Användaren får sedan välja vilka fält som ska ändras, här gäller samma validering för fälten som i formuläret för att lägga till böcker.

När användaren har gjort sina ändringar så läggs ändringarna in genom att trycka på *Submit Changes*


Formuläret finns på: http://shinowa.tk/lab5/book_changeTitleForm.html

2.3 Exempelkörningar

2.3.1 Lägga till bok

Test 1: Korrekt ifyllda fält

Jag fyller i alla fält med korrekta värden.



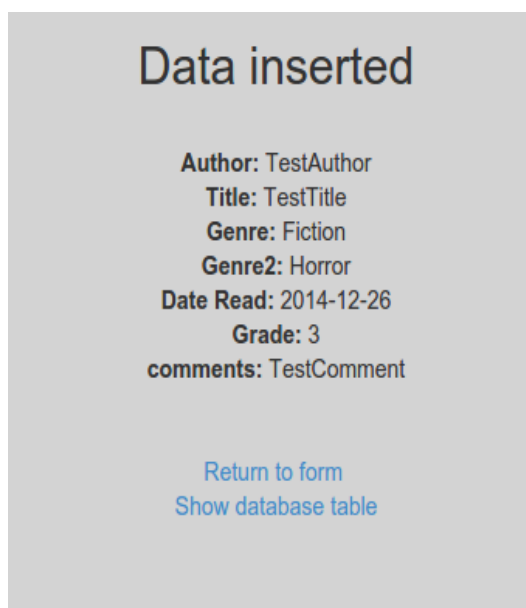
The screenshot shows a web form titled "Add Form" with a subtitle "Data". The form contains the following fields and values:

- Author: TestAuthor
- Title: TestTitle
- Date Read: 12/26/2014
- Genre: Fiction (dropdown menu)
- Genre2: Horror (dropdown menu)
- Grade: 3 (dropdown menu)
- Comments: TestComment (text area)

Below the form is a blue button labeled "Add book" and a blue link labeled "Show database table".

Figure 9: Formulär med korrekt ifyllda fält

Resultat:



The screenshot shows a web page titled "Data inserted". It displays the data that was added to the database:

- Author: TestAuthor
- Title: TestTitle
- Genre: Fiction
- Genre2: Horror
- Date Read: 2014-12-26
- Grade: 3
- comments: TestComment

Below the data is a blue link labeled "Return to form" and a blue link labeled "Show database table".

Figure 10: Resultat

Test 2: Felaktig Author

Jag skriver in i *Author* fältet: **fel1**, *Author* får ej innehålla siffror.

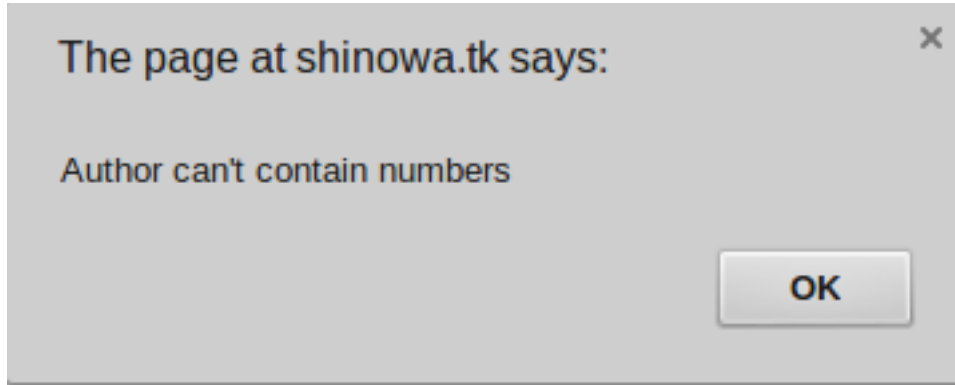


Figure 11: Resultat

Kommentar: När jag trycker på Add book så visas denna popup.

Title

Titel får innehålla alla tecken, inget att testa.

Felaktigt Datum

Är av typen date i formuläret så det är svårt att göra fel i detta fält. Man kan däremot mata in konstiga år tex: 123123, lite långt i framtiden kanske.

Genre

Går ej att göra fel på.

Genre2

Genre2's lista uppdateras beroende på vilken värde som *Genre* har, detta har jag testat manuellt och det verkar fungera felfritt.

Grade

Går ej att göra fel på.

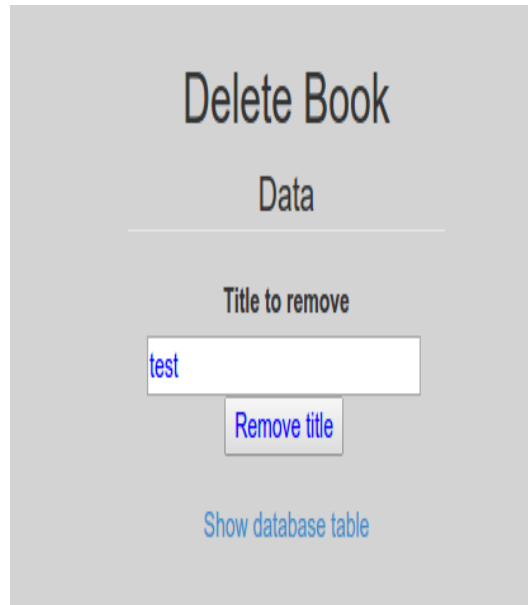
Comments

Får vara tom och innehålla alla sorters tecken, så inget att testa här.

2.3.2 Ta bort bok

Test 1: ta bort en titel som finns

Jag skriver in en titel som jag vet finns i databasen vid testets tidpunkt.



Delete Book

Data

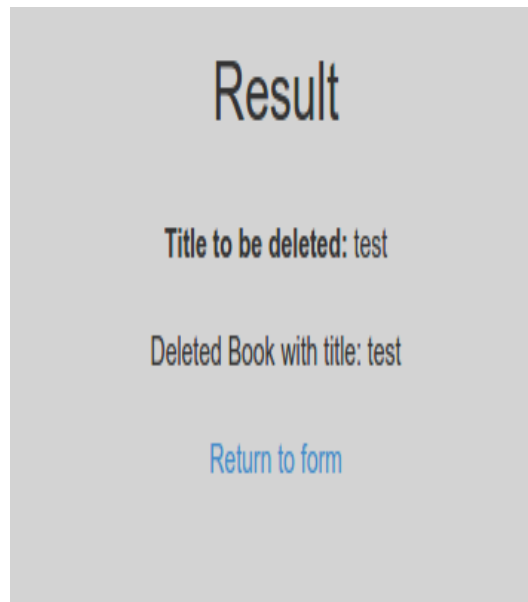
Title to remove

Remove title

[Show database table](#)

Figure 12: Titel att ta bort

Resultat:



Result

Title to be deleted: test

Deleted Book with title: test

[Return to form](#)

Figure 13: Resultat

Test 2: Ta bort titel som inte finns

Jag försöker ta bort titeln: finnsinte.

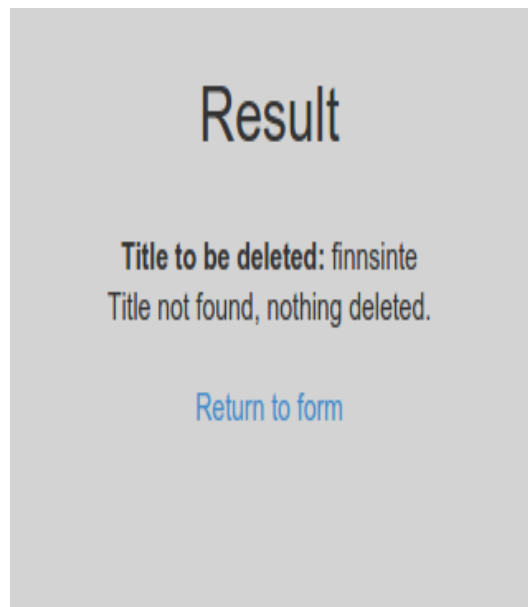


Figure 14: Resultat

2.3.3 Ändra bokdata

Innehåller samma delar som används för att lägga till en bok, så tester på denna del kommer att ge samma resultat som *Add book* formuläret för att de använder samma funktion för att valideras.

3 Diskussion

Jag har i mina lösningar valt att prova två stycken sätt att validera datat som skrivs in i formulären, i del 1 så använder jag php för detta och det fungerar bra, det negativa med detta är att användaren måste gå tillbaka och skriva om det som är fel. I del två så använder jag javascript och när användaren ska submitta så kontrolleras alla fält och en popup visas om något är fel utan att webbläsaren byter sida. Jag personligen tycker att del 2's lösning är bättre för det går snabbare att ändra formuläret så det blir korrekt och det behöver inte läggas ner tid på att visa en ny sida i webbläsaren och användaren behöver inte klicka *Back*.

Jag har inte hållit på med php förrut och det jag tycker var svårt med php är det att man bakar in html kod i php koden, det kändes svårt att få koden lättläslig när man blandar html och php, men detta kan vara en vanesak samt att det kan finnas bättre sätt att skriva vissa delar på än vad jag lyckats åstadkomma.

När det gäller formulären så hjälper de typer man kan sätta de olika input-fälten till väldigt mycket, email, date number är mycket smidiga eftersom att dom själva ser till så att användaren bara kan skriva in något i fältet som är korrekt.

Jag har använt PDO med parametriserade *Queries* för mysql delen för att försöka att göra det *Sql-Injection* säkert men jag har ingen tidigare erfarenhet av databaser eller *Sql-Injection* så det är nog inte supersäkert. Får gärna försöka att titta om det går att förstöra databasen då det troligen finns mera erfarenhet av sånt för dig som rättar.