Web accessible Databases
PHP

October 16, 2017

www.php.net

HTML Primer

- https://www.w3schools.com/html/default.asp
  - HOME
  - Introduction
  - Basic
  - Tables
  - Lists
- https://developer.mozilla.org/en-US/docs/Learn/HTML/Introduction_to_HTML
  - Getting started with HTML
  - What's in the head?
  - HTML text fundamentals
  - Creating hyperlinks

Coding Standards

- http://pear.php.net/manual/en/standards.php
  - we will use these
  - well organized
  - similar to our own C/C++
  - only downsides:

```
    indent four spaces
    if (test) {
```

- Other popular standards:

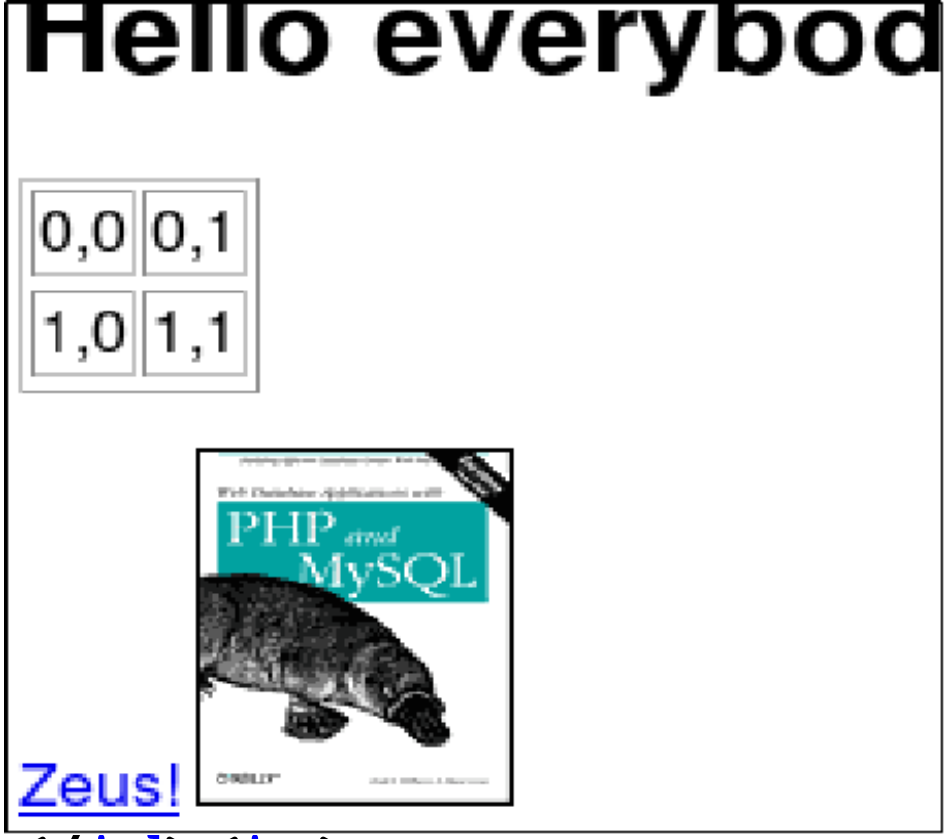
  http://www.php-fig.org/psr/psr-1/
  http://www.php-fig.org/psr/psr-2/

  http://symfony.com/doc/current/contributing/code/standards.html

  http://www.phptherightway.com/#code_style_guide

simple.html

```
<html>
   <head>
     <title>The Window Title
     </title>
   </head>
  <body>
     <h1>Hello everybody!</h1>
     <P/>
     <table border=1>
        <tr><td>0,0</td><td>0,1</td><tr>
        <tr><td>1,0</td><td>1,1</td><tr>
     </table>
     <p/>
     <a href="http://zeus.cs.pacificu.edu">Zeus!</a>
     <img src="image/php.gif">
  </body>

</html>
```



HelloWorld.php

```
<html>
   <head>
     <title>The Window Title
     </title>
   </head>
  <body>

     <?php
        // HelloWorld.php
        print "Hello World!";
        print "<H1>Hello World!</H1>";
     ?>

  </body>
</html>
```

Danger! Quotation marks do not copy and paste well!

Comment!

The web browser only sees the HTML, not the PHP.
View | Page Source

A file that contains ANY php MUST have a .php extension!

VariablesIfs.php

```
<body>
   <H1>
   <?php
      $counter = 1; // create variable
      if( 0 == $counter )
      {
         print "ZERO";
      }
      else
      {
         print $counter;
      }
   ?>
   </H1>
</body>
```

Loops.php

```
<body>

   <?php
      $counter = 1; // create variable
      while( $counter < 10)
      {
         print $counter . " " . $counter*2;
         print "<p/>";
         $counter += 1;
      }
   ?>

</body>
```

String concatenation is done with a dot .

LoopsTable.php

```
<table border=1 cellpadding=4>
    <?php
       $rows = 1; // create variable
       while( $rows < 10)
       {
          print "<tr>";
          $columns = 1; // create variable
          while( $columns < 10)
          {
             print "<td>";
             print $rows . " , " . $columns;
             print "</td>";
             $columns += 1;
          }
          print "</tr>";
          $rows += 1;
       }
    ?>
 </table>
```

Disjoint.php

```
<body>
    <?php
       print "<table border=1> <tr>";
       $columns = 1; // create variable
       while( $columns < 10)
       {
          print "<td>" . $columns . "</td>";
          $columns += 1;
       }
       print "</tr> </table>";
    ?>

    Hello out there
    <center> HI!</center>

    <?php
       print $columns; // retains value from above
    ?>
 </body>
```

```php
<?php          // sessionTest.php
  session_start();
  $_SESSION['PID']=2; // global associative array
                      // acts like a hash table
  header('Location: showPID.php');

 ?>
```

IMPORTANT:
There must be no blank lines or HTML before the **header**() function call!

```php
<?php          // showPID.php
  session_start();
  if( isset($_SESSION['PID']))
  {
    print $_SESSION['PID'];
  }

 ?>
```

Exercises

- Write a php file to display the first 100 odd integers in a table

- Write a php file to set a session variable (SESS_TEST) to 42 and redirect to another php page which prints all the integers 1 to SESS_TEST. Be sure to use isset() to determine if SESS_TEST is set.

- BONUS: Have the table in either of the above pages alternate colors for rows.

Stop Wednesday

Connect to MySQL

```
<?php          // connDB.php
  function db_connect ()
  {
     $dbh = new
       PDO("mysql:host=127.0.0.1;
       dbname=DBNAME","USER", "PASSWORD");

     $dbh->setAttribute (PDO::ATTR_ERRMODE,
        PDO::ERRMODE_EXCEPTION);

    return ($dbh);

  }
  function db_close (& $dbh)
  {
      $dbh = null;
  }
 ?>
```

Good Coding

- We want to separate the data access from the presentation as much as we can
  - query files
  - presentation files
  - all are .php files
- Query files: write data access functions.
  - many presentations files can access the same query
  - may have many functions per file
- skeleton.php is an example of a presentation file
  - lots of HTML and PHP function calls to get/present data

Good Coding: Database Security

- Security is a continuous process
- Separate Read Only and Read Write access via MySQL accounts
- Don't give a web page more access to MySQL than it needs
  - Reduces the change of SQL Injection
-

Good Coding: Web Security

- Cross Site Scripting (JavaScript Injection)
- https://www.owasp.org/index.php/XSS_(Cross_Site_Scripting)_Prevention_Cheat_Sheet
- http://www.php.net/manual/en/function.htmlentities.php
- http://www.php.net/manual/en/function.htmlentities.php#99896
- http://www.php.net/htmlspecialchars

Presentation file skeleton.php

```
<?php
  require_once 'connDB.php';
  session_start();

   $dbh = db_connect();

?>

<html>
  <head>
     <title></title>
  </head>
  <body>
     MIX OF PHP AND HTML
  </body>

</html>


<?php
     db_close($dbh);
?>
```

Rather than **print** every line of HTML, you can inline HTML outside of the **<?php ?>** tags and it is automatically printed

php functions

```
<?php  // print.php

  function printData ($data1, $data2)
  {

    $lString = $data1 . " " . $data2;

    print $lString;
    return $lString;

  }

?>
```

```
<?php   //testPrint.php
  require_once 'print.php';
  $result = printData("hello","World");

  print $result;

?>
```

This code could be in the <body> of the skeleton.php!

You might collect all the includes at the top.

php functions

```
<?php  // passByReference.php
  function printDataRef (&$data1, &$data2)
  {
    $lString = $data1 . " " . $data2;
    print $lString;
    return $lString;
  }
?>
```

```
<?php   //globalVariables.php
   $gValue = 1;
   function printDataGlobal($data)
   {
     global $gValue;  // this attaches the name
                      // to the global variable.
     print $gValue . ' ' . $data;
   }
?>
```

Query Syntax

```
// already opened the database
$sth = $dbh->prepare(
 "SELECT FName, LName FROM Client");

// run the query
$sth->execute();

printf("Results Count: %d\n", $sth->columnCount());
 while( $row = $sth->fetch() )
 {
   printf("FName: %s LName %s\n",
      $row[0], $row["LName"]);
 }
```

Error Handling

```
$sth = $dbh->prepare( …. );

try
{
    $sth ->execute( );
}
catch (PDOException $e)
{
    printf ("The statement failed.\n");
    printf ("getCode: ". $e->getCode () . "\n");
    printf ("getMessage: ". $e->getMessage () . "\n");
}
```

queryFunction.php

```
function getAllPeopleNames($dbh)

{
   $rows = array();

   $sth = $dbh->prepare(
    "SELECT FName, LName FROM Client");
    // run the query
   $sth->execute();

    printf("Results Count: %d\n", $sth->columnCount());
     while( $row = $sth->fetch() )
     {
        $rows[] = $row;
     }
    return $rows;
}
```

queryFunctionCall.php

```
<?php

include 'connDB.php';
include 'queryFunction.php';

$dbh = db_connect ();
$data = getAllPeopleNames($dbh);

foreach ( $data as $row )
{
  print $row['FName'] . ' ' . $row['LName']
     .' <br/> ';

}

db_close($dbh);

?>
```

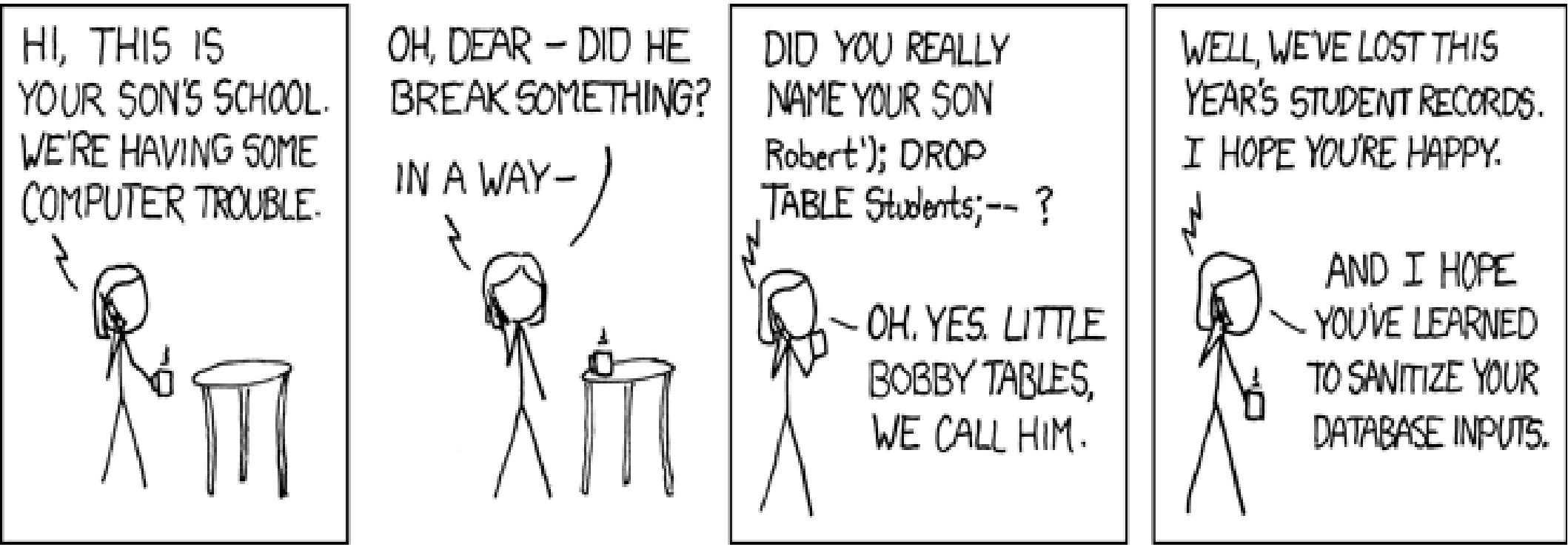
queryFunctionParams.php

```
function getAllPeopleNamesWhereLName
        ($dbh, $LName)

{
  $rows = array();

  $sth = $dbh->prepare(
   "SELECT FName, LName FROM Client
     WHERE LName like :name");


  $sth->bindValue(":name",$LName);
   // run the query
  $sth->execute();


  // same as getAllPeopleNames......
```

queryFunctionCallParams.php

```
<?php

include 'connDB.php';
include 'queryFunctionParams.php';

$dbh = db_connect();
$data = getAllPeopleNamesWhereLName($dbh, “Greene”);

foreach ( $data as $row )
{
  print $row['FName'] . ' ' . $row['LName'].'<br/>';

}

db_close($dbh);

?>
```

Why we use prepared statements?




```
SELECT username FROM users WHERE username ='$userId';

$userId = ' bob'); Drop Table Students; --”

mysql_real_escape_string(): $userId = ' bob\\\');
Drop Table Students; --”
```

SQL Injection

runQueryTable.php

Software	FName	LName	Email	Salary
Stellar Teller	Aline	Maddox	elementum.purus.accumsan@parturient.edu	153308
Word Precise	Quintessa	Frederick	et@Curae;Donectincidunt.ca	167687
Anodize	Carissa	Ford	Quisque@elitpellentesquea.ca	23308
ATM Buddy	Odette	Espinoza	non.sollicitudin@variusorciin.org	153903
Where Am I? GPS App	Ursula	Stewart	condimentum@a.edu	49855
Speller	Tyrone	Wong	lacus.Quisque@DonecnibhQuisque.edu	31763
SpellerLite	Wyatt	Figueroa	ullamcorper@montesnascetur.ca	73617
StoryTeller	Michael	Atkinson	laoreet.lectus@necorciDonec.com	63376
Stone Tablet	Oscar	Cox	Vivamus.rhoncus@Suspendissealiquet.org	61079
Vi	Quail	Crawford	convallis.in.cursus@orci.ca	77551
mauris id	Martin	Mccarthy	ipsum@nec.edu	178511
Vivamus nibh	Palmer	Albert	euismod.in@perconubia.edu	46726

Can you build this web page? This is software,manager, manager's email, and manager's salary.

Exercises

- Build a web page that displays the FName, LName, of each employee and the FName, LName of that employee's Manager.

- Build a web page the displays the total salary earned by all the employees who work on each software product (One row per software product).
  Display $0.00 if a product has no one working on it.

STOP FRIDAY

Input from the user

Textfield

Textarea

List RED GREE YELL BLUE

Checkbox green red blue

Buton FM AM

Dropdown Wyatt Figueroa

Submit

```
<form method="post" action="showWorksOn.php">
    Manager:
    <select NAME="EmpID">
      <option VALUE="9">Wyatt Figueroa</option>
      <option VALUE="7">Ursula Stewart</option>
      <option VALUE="6">Odette Espinoza</option>
    </select>
    <input TYPE="submit" NAME="Request" VALUE="Go" />
</form>
```

showWorksOn.php

```php
<?php

  require_once 'connDB.php';
  require_once 'queryWorksOnByEmpID.php';

  if( !isset ($_POST['EmpID']) )
  {
     die("ERROR: No EmpID");
  }

  $EmpID = $_POST['EmpID'];

  $dbh = db_connect();
  $data = getWorksOnByEmpID($dbh, $EmpID);

  // display data in table

?>
```

Other Input Types

```
<input TYPE="submit" NAME="Request" VALUE="Go" />
```

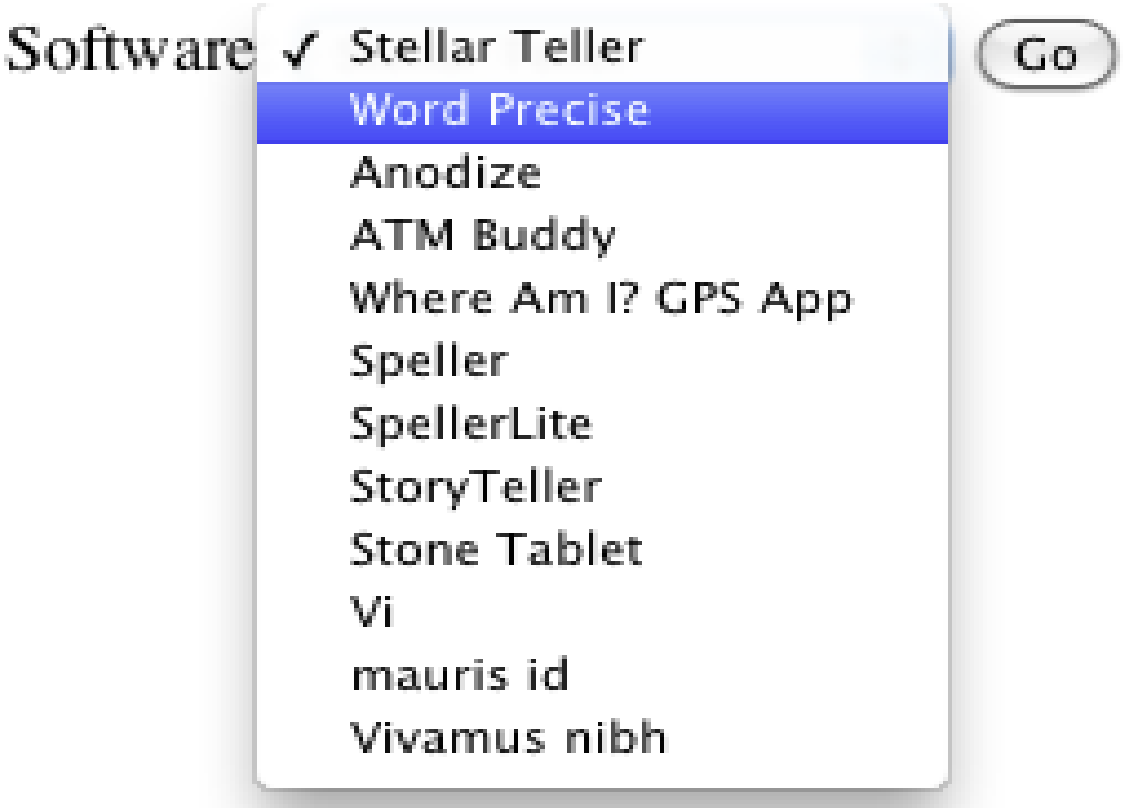
- **TYPE=”text”**
- **TYPE=”password”**
- **TYPE=”radio”**
- **TYPE=”checkbox”**
- **TYPE=”textarea”**

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

Exercises

- Build a page to show all client information

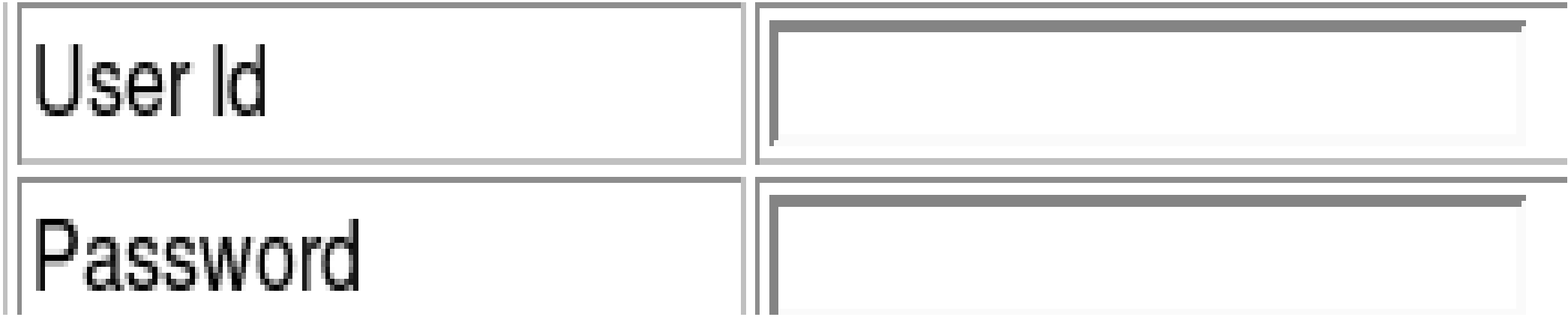
- Build a page to auto-populate a select box with software and then show all the software the chosen software directly depends on.



User Authentication

- Store usernames and passwords in the DB
  - Don't make a MySQL account for every user!
  - Securely store the passwords!

```
create table users (username varbinary(25),
                    passwd varbinary(XX),
                    salt varbinary(YY),
                    Primary Key (username));
```



Password Security

- Threats:
  - Intercept in flight
    - Solution: SSL/https
  - Brute force attack (external)
    - Solution: strong passwords, limited login failures
  - Brute force attack (internal)
    - someone stole your database and has the users table!
    - Solution: store hashed passwords
      - salted passwords
      - choose a good hash algorithm

Pseudo-code

```
$salt = generateRandomString();
$hashedPwd = somehash($passwd . $salt);

“Insert into table users values ($user,
$hashedPwd, $salt);”
```

Job of the salt:

Job of the hash:

Other Resources

http://www.php.net/manual/en/faq.passwords.php

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

http://www.ibm.com/developerworks/opensource/library/os-php-encrypt/

http://stackoverflow.com/questions/1581610/how-can-i-store-my-users-passwords-safely

http://php.net/manual/en/function.crypt.php

http://www.openwall.com/phpass/

userAuth.php

```
<?php
$_SESSION['VALID'] = 0;

if(  isset($_POST['txtUser']) &&
     isset($_POST['txtPassword']))
{
   $userID = $_POST['txtUser'];
   $passwd = $_POST['txtPassword'];

   $result = queryValidUser($dbh, $userID, $passwd);

   if( TRUE == $result )
   {
      $_SESSION['VALID'] = 1;
      header('Location: loggedIn.php');
   }
   else
   {
      header('Location: login.html');
```

queryValidateUser.php

```
<?php
function queryValidateUser($dbh, $user, $passwd)
{
   $retVal = FALSE;
   $salt = queryGetSalt($dbh, $user);

   $hashedPW = crypt($passwd.$salt,
      '$2y$07$8d88bb4a9916b302c1c68c$');

   $sth = $dbh->prepare("SELECT * FROM users WHERE
        username = :user and passwd = :pass");
    $sth->bindValue(":user",$user);
    $sth->bindValue(":pass",$hashedPW);
   $sth->execute();

    if( 1 == $sthWhereName -> columnCount()  )
    {
      $retVal = TRUE;
    }
   return $retVal;
}
```

login.html

```
<body>

  <form method="post" name="frmLogin" action=
  "authUser.php">

  Username:
  <input name="txtUserId" type="text" >

  Password:
  <input name="txtPassword" type="password">

  <input type="submit" name="btnLogin" value="Login">

  </form>

</body>
```

authHelper.php

```
<?php
   // include this code at the top of each
   // php file that requires the user to
   // have already been authenticated

   if( !isset($_SESSION['VALID']) ||
       $_SESSION['VALID'] != 1 )
   {
      header('Location: login.html');
   }

?>
```

Binary Data

```
CREATE TABLE pictures (
    `PicID` int(11) NOT NULL auto_increment,
    `image` mediumblob NOT NULL,
    `type` varchar(255) NOT NULL,
   PRIMARY KEY  (`PicID`)) ENGINE=InnoDB;
```

For binary data, we need to track the type of data we have stored.

Usually the MIME type.

image/gif

image/png

binaryDataInput.php

```
<body>
    <form method="post"
      action=binaryDataInput.php
      enctype="multipart/form-data">

      <input type="hidden" name="MAX_FILE_SIZE"
        value="1000000">

      <br>File to upload/store in database:<br>
      <input type="file" name="datafile"  size="40">

     <p>
      <input type="submit" name="submit"
        value="submit">
    </form>
</body>
```

binaryDataInput.php

```
<?php
if(isset($_POST['submit']) ) {

  $filename = $_FILES['datafile']['tmp_name'];
  $filesize = $_FILES['datafile']['size'];
  $filetype = $_FILES['datafile']['type'];

  $data = fread( fopen($filename, "r"),
                        filesize($filename));
  $sth = $dbh->prepare("INSERT INTO pictures
     VALUES (null, :data , :filetype)");
  $sth->bindValue(":data", $data);
  $sth->bindValue(":filetype",$filetype);

  $sth->execute();

  print "We just added PicID:". $dbh->lastInsertId();;
  print "{$filetype} {$_FILES['datafile']['name']}";

}
?>
```

http://www.phpbuilder.com/columns/florian19991014.php3?page=2

getData.php

```
<?php
  if( isset($_GET['id']) ) {

    $id = $_GET['id'];
    $sth = $dbh->prepare("select image, type from
        pictures where PicID=:picid";
    $sth->bindValue(":picid",$id);

    $sth->execute();

    $row = $sth->fetch(); // typo on handouts!
    $data = $row['image'];
    $type = $row['type'];

    Header( "Content-type: $type");
    print $data;

  }else{
     print "FILE NOT FOUND";
  }
?>
```

showImage.html

https://64.59.233.246/chadd/getData.php?id=1

```
<html>
   <body>
      Image: <img src="getData.php?id=24">
   </body>
</html>
```

```
<html>
   <body>
      Image: <img src="getData.php?id=
   <?php $_SESSION['PicID'] ?>">
   </body>
</html>
```

Practice Exercise

- Add an Editor field to the user table
  - only allow people marked as editors to insert data in the queries below
- Build a webpage to create a new user
- Build a webpage that allows a user to enter a new Student
  - provide a drop down box listing all majors
- Build a webpage that allows the user to search for Students that received a specific final grade
  - provide a drop down box listing grades (A,A-,B+,B,...)