

# Discussion Forum website

## **Start date: 2/13/19**

In this project, you will be making a discussion forum website where users can discuss a wide variety of subjects. This type of discussion forum won't be restricted to one subject (such as gaming forums like IGN or Gamefaqs or Gamespot where they mainly discuss video games (they do discuss other non-video game subjects, but they are mainly a video game discussion site), or sites like TheApricity or ForumBiodiversity where they only discuss Anthropology related subjects (though sometimes cultural also)).

The main focus here is creating a database that stores the following user information:

- Full name of users
- Username of users
- Password of users
- Threads (topic question of a particular subject) last created by users
- Posts/Comments (reply to a Thread) last created by users
- Messages that users have sent to other users

The design of the forum website will consist of the following:

1. The signup/login page for users to create a forum account to access the forum home page.
2. The profile page of users where users can change any information such as username, password, picture, etc.
3. The forum home page itself consisting of the discussion topics.
4. The sub-forums as well as the message thread topic of the discussion at hand which also shows the most recent replies/views and who posted the most recent reply.
5. The actual message thread with the comments of users.

Create a folder called "Discussion Forum", and save it to the htdocs folder inside the XAMPP folder, and this is where all the necessary files for your Discussion Forum website will go.

The first step you want to do is design the layout of your website: create the login and signup pages, the actual forum home page that lists the subjects, the page containing the particular subject with the forum topics, the posts on that topic, and the user who lasted posted on that forum topic.

First is to design the home page of forum website. Download the necessary CSS/Bootstrap and JavaScript/jQuery files needed and store them in the folders 'css' and 'js' respectively.

Create a file called "forumhome.php", and this will be the forum home page. You just want to create the layout, then go back to add more functionality.

### **forumhome.php**

```
<html>
  <head>
    <title> Welcome to forum </title>
    <meta charset = "utf-8"/>

    <link href="https://fonts.googleapis.com/css?family=Lobster" rel="stylesheet">
    <link rel = "stylesheet" href = "css/bootstrap.css"/>
    <link rel = "stylesheet" href = "css/bootstrap.min.css"/>
    <link rel = "stylesheet" href = "css/main.css"/>
  </head>

  <body>
    <!-- Navbar -->
    <nav class = "navbar navbar-inverse">
      <div class = "container-fluid">
        <div class = "navbar-header">
          <a href = "#" class = "navbar-brand"> ForumWebsite </a>
        </div>

        <ul class = "nav navbar-nav">
          <li><a href = "#"> Home </a></li>
          <li><a href = "#"> Boards </a></li>
          <li><a href = "#"> Community </a></li>
        </ul>

        <ul class = "nav navbar-nav navbar-right">
          <li><a href = "#"><span class = "glyphicon glyphicon-user"></span> Logout
</a></li>
          <li><a href = "#"> Settings </a></li>
          <li><a href = "#"> Go to Profile </a></li>
        </ul>
      </div>
    </nav>
  </body>
</html>
```

```

</div>
</nav>

<!-- Search for forums -->
<div id = "searchwindow">
  <p> Search for your favorite topic on a subject </p>

  <form>
    <input type = "text" name = "subjectsearch" placeholder = "Enter Subject"/>
    <br/>
    <input type = "text" name = "topicsearch" placeholder = "Enter Topic of subject"/>
    <br/>
    <input type = "submit" name = "submit-search" value = "Search"/>
  </form>
</div>

<!-- Forum window -->
<div class = "container" id = "forumwindow">
  <table class = "table">
    <thead>
      <tr>
        <th> Subject </th>
        <th> Topics </th>
        <th> Posts </th>
        <th> Latest Post </th>
      </tr>
    </thead>

    <tbody>
      <tr>
        <td> subject 1 </td>
        <td> topic of subject 1 </td>
        <td> this is a thread question </td>
        <td> last person who posted </td>
      </tr>
      <tr>
        <td> subject 2 </td>
        <td> topic of subject 2 </td>
        <td> this is a thread question </td>
        <td> last person who posted </td>
      </tr>
    </tbody>
  </table>
</div>

<script src = "js/jquery-3.3.1.js"/></script>

```

```
<script src = "js/bootstrap.js"/></script>
<script src = "js/jquery-ui.min.js"/></script>
<script src = "js/main.js"/></script>
</body>
</html>
```

### **main.css**

```
body
{
    background-image:
url("https://png.pngtree.com/thumb_back/fh260/back_pic/04/00/23/4757fc9e77387db.jpg");
    background-repeat: no-repeat;
    background-size: cover;
}

#searchwindow
{
    padding-left: 25%;
    padding-top: 10%;
}

#searchwindow input[type=text]
{
    width: 60%;
    height: 30px;

    background-color: #F5F5DC;
    font-family: Times New Roman;
}

#searchwindow input[type=submit]
{
    width: 60%;
    height: 30px;

    background-color: #F5FFFA;
    font-family: Times New Roman;
}

#searchwindow p
{
    padding-right: 30%;
    font-family: "Lobster", cursive;
    font-size: 200%;
}
```

```
#forumwindow
{
    padding-top: 80px;
    padding-left: 40px;
}

#searchwindow p, #forumwindow .table th, #forumwindow .table td
{
    color: #D3D3D3;
}

#forumwindow .table th, #forumwindow .table td
{
    font-family: Times New Roman;
}
```

(again, you just want to create the layout, then you can go back to add more)

---

## **2/16/19**

Now you want to create the signup and login pages where users can login to the forum home page. You also want to create the profile page for users to edit their profile such as their name, photo, email, etc.

Again, you just want to create the layout. You can always go back to add more design and dynamic functionality.

Create a file called “signupPage.php”, and this is where users will be able to create their account for the forum.

```
signupPage.php
<html>
  <head>
    <title> Register an account now! </title>
    <meta charset = "utf-8"/>

    <link rel = "stylesheet" href = "css/signup.css"/>
  </head>

  <body>
    <div id = "headings">
      <h1> Welcome </h1>
```

```
<h4> Please sign up now to join our discussion forum </h4>
</div>

<div id = "signupwindow">
  <h3> Member Signup </h3>

  <form method = "POST">
    <input type = "text" placeholder = "Enter First Name" name = "fname"/>
    <br/>
    <input type = "text" placeholder = "Enter Last Name" name = "lname"/>
    <br/>
    <input type = "text" placeholder = "Enter Desired Username" name = "uname"/>
    <br/>
    <input type = "password" placeholder = "Enter Desired Password" name =
"pwd"/>
    <br/>
    <input type = "submit" value = "Register" name = "submitButton"/>
  </form>

  <a href = '#'> <h5> Already have an account, log in now </h5> </a>
</div>
</body>
</html>
```

#### signup.css

```
body
{
  background-color: #191970;
}

#headings
{
  text-align: center;

  margin-top: 10%;
}

#headings h1, #headings h4
{
  color: #DCDCDC;
  font-family: cursive;
}

#signupwindow
{
  border: 1px solid black;
```

```
margin-top: 5%;
margin-right: 20%;
margin-left: 20%;

background-image:
url("https://png.pngtree.com/thumb_back/fh260/back_pic/04/00/23/4757fc9e77387db.jpg");
background-repeat: no-repeat;
background-size: cover;

box-shadow: 5px black;
}

#signupwindow h3
{
    color: #FAF0E6;
    text-align: center;
}

#signupwindow form
{
    padding-top: 10px;
    text-align: center;
}

#signupwindow form input[type=text], #signupwindow form input[type=password]
{
    width: 45%;
    height: 7%;

    font-size: 20px;
    font-family: "Times New Roman";
    background-color: #FAF0E6;

    border-radius: 5px;
}

#signupwindow form input[type=submit]
{
    width: 20%;
    height: 5%;

    font-family: cursive, "Times New Roman";
    font-size: 20px;
    color: #F0FFFF;
```

```

border: none;
border-radius: 5px;

cursor: pointer;

background-color: #4169E1;
}

#signupwindow h5
{
text-align: center;
color: #F0FFF0;
}

```

Now create file called “loginPage.php”, and this is where users will login with their username and password credentials.

### **loginPage.php**

```

<html>
  <head>
    <title> Login to account </title>
    <meta charset = "utf-8"/>

    <link rel = "stylesheet" href = "css/login.css"/>
    <link href="https://fonts.googleapis.com/css?family=Pacifico" rel="stylesheet">
  </head>

  <body>
    <div id = "headings">
      <h1> Welcome to the Discussion forums </h1>
      <h4> Please login to enter </h4>
    </div>

    <div id = "loginwindow">
      <h4> Member Login </h4>

      <form method = "POST">
        <input type = "text" placeholder = "Enter Username" name = "loginuname"/>
        <br/>
        <input type = "password" placeholder = "Enter Password" name = "loginpwd"/>
        <br/>
        <input type = "submit" value = "Login" name = "loginButton"/>
      </form>

      <a href = "#"><h4> Forgot password? Click here </h4></a>
    </div>

```



```
</body>
</html>
```

### **login.css**

```
body
{
  background-color: #191970;
}

#headings
{
  margin-top: 20%;
  margin-right: 23%;
  margin-left: 23%;

  text-align: center;

  color: #EEE8AA;
}

#headings h1
{
  font-family: 'Pacifico', cursive;
}

#loginwindow
{
  background-image:
url("https://png.pngtree.com/thumb_back/fh260/back_pic/04/00/23/4757fc9e77387db.jpg");
  background-repeat: no-repeat;
  background-size: cover;

  border: 1px solid black;

  margin-top: 5%;
  margin-right: 20%;
  margin-left: 20%;

  text-align: center;

  box-shadow: 5px black;
}

#loginwindow h4
{
  color: #EEE8AA;
```

```
}

#loginwindow input[type=text], #loginwindow input[type=password]
{
    width: 50%;
    height: 5%;

    font-size: 20px;
    font-family: "Times New Roman";
    background-color: #FAF0E6;

    border-radius: 5px;
}

#loginwindow input[type=submit]
{
    background-color: #4169E1;

    width: 30%;
    height: 5%;

    font-family: cursive, "Times New Roman";
    font-size: 20px;
    color: #F0FFFF;

    border: none;
    border-radius: 5px;

    cursor: pointer;
}

#loginwindow h4
{
    text-align: center;
}
```

Now you want to design the user profile for users when they successfully signed up an account for your forum website. The profile page should look the same for all users, but should be unique for each user. In other words, one user cannot have the same information as another user such as name and picture.

Create a file called “userprofile.php”, and this will be the user profile. Again, you just want to create the layout. You can finish everything later on:

#### **userprofile.php**

```
<html>
  <head>
    <title> Welcome to your profile </title>
    <meta charset = "utf-8"/>

    <link rel = "stylesheet" href = "css/bootstrap.css"/>
    <link rel = "stylesheet" href = "css/bootstrap.min.css"/>
    <link rel = "stylesheet" href = "css/userprofile.css"/>
  </head>

  <body>
    <div id = "topheading">
      <h1> Discussion Forums </h1>
    </div>

    <div id = "tab">
      <nav class = "navbar navbar-default">
        <div class = "container-fluid">
          <div class = "navbar-header">
            <a href = "#" class = "navbar-brand"> Your Profile </a>
          </div>

          <ul class = "nav navbar-nav navbar-right">
            <li><a href = "#"> Settings </a></li>
            <li><a href = "#"> Logout </a></li>
          </ul>
        </div>
      </nav>
    </div>

    <div id = "profilewindow">
      <ul class = "nav nav-tabs">
        <li><a data-toggle = "tab" href = "#home"> Home </a></li>
        <li><a data-toggle = "tab" href = "#menu1"> Edit Information </a></li>
        <li><a data-toggle = "tab" href = "#menu2"> Messages </a></li>
        <li><a data-toggle = "tab" href = "#menu3"> Recent Posts </a></li>
        <li><a data-toggle = "tab" href = "#menu4"> Friends/Acquaintances </a></li>
      </ul>
```

```

<div class = "tab-content">
  <div id = "home" class = "tab-pane fade in active">
    <p> Profile picture goes here </p>
    <p> User email goes here </p>
    <p> Full name goes here </p>
    <p> Gender goes here </p>
    <p> Where user resides goes here </p>
    <p> A short story about user goes here </p>
  </div>

  <div id = "menu1" class = "tab-pane fade">
    <!-- These are the profile settings -->
    <p> Change profile pic </p>
    <p> Change name </p>
    <p> Change gender </p>
    <p> Change email </p>
    <p> Change Location </p>
    <p> Change About me </p>
  </div>
</div>

<script src = "js/jquery-3.3.1.js"/></script>
<script src = "js/bootstrap.js"/></script>
<script src = "js/jquery-ui.min.js"/></script>
</body>
</html>

```

---

*Handling the signup and login process, Creating the forum database*

---

**2/24/19**

(Now you want to take a stop from designing right now and start implementing the php code for signing up new users, bringing new users to the profile page upon signup).

You also want to create the database that contains user information from the signup page.

First, create a folder called “includes”, and this file will contain all the Php files that do all the database and server access code in the background.

Then, inside localhost/phymyadmin, create the database named “forumaccounts” and the table called “accounts” which will store user account information for signing into the forum and other information such as saved messages, threads, posts, etc.

Enter the following SQL code to create the database and table (have sure you have the forumaccount database selected):

```
create table accounts(  
  user_id int(11) not null PRIMARY KEY AUTO_INCREMENT,  
  user_firstname varchar(100) not null,  
  user_lastname varchar(100) not null,  
  user_email varchar(100) not null,  
  user_username varchar(100) not null,  
  user_password varchar(100) not null  
);
```

Now you should see an ‘accounts’ table inside the forumaccounts database with the following columns:

user_id	user_firstname	user_lastname	user_email	user_username	user_password
---------	----------------	---------------	------------	---------------	---------------

Now create the php file called “database.php”, and this gives access to the forumaccounts database. Inside the database.php, input the following:

```
database.php  
<?php  
  $Servername = "localhost"; //name of server the database is located (which is localhost)  
  $Username = "root"; //username of the server (for localhost, it is 'root')  
  $Password = ""; //password of the server (for localhost, it is blank by default)  
  $databaseName = "forumaccounts"; //name of the database to be accessed  
  
  //creates connection to the forumaccounts $databaseName  
  $conn = mysqli_connect($Servername, $Username, $Password, $databaseName);  
?>
```

Now you want to add the database connection to both the loginPage.php and signupPage.php:

```
loginPage.php  
<?php  
  include_once 'includes/database.php';  
?>  
  
<html>  
  <head>  
    <title> Login to account </title>  
    <meta charset = "utf-8"/>
```

```
signupPage.php  
<?php  
  include_once  
'includes/database.php';  
?>  
  
<html>  
  <head>  
    <title> Register an account  
now! </title>
```

<pre> &lt;link rel = "stylesheet" href = "css/login.css"/&gt; &lt;link href="https://fonts.googleapis.com/css?family=Pacifico" rel="stylesheet"&gt; &lt;/head&gt;  &lt;body&gt; &lt;div id = "headings"&gt; &lt;h1&gt; Welcome to the Discussion forums &lt;/h1&gt; &lt;h4&gt; Please login to enter &lt;/h4&gt; &lt;/div&gt;  &lt;div id = "loginwindow"&gt; &lt;h4&gt; Member Login &lt;/h4&gt;  &lt;form method = "POST"&gt; &lt;input type = "text" placeholder = "Enter Username" name = "loginuname"/&gt; &lt;br/&gt; &lt;input type = "password" placeholder = "Enter Password" name = "loginpwd"/&gt; &lt;br/&gt; &lt;input type = "submit" value = "Login" name = "loginButton"/&gt; &lt;/form&gt;  &lt;a href = "#"&gt;&lt;h4&gt; Forgot password? Click here &lt;/h4&gt;&lt;/a&gt; &lt;/div&gt; &lt;/body&gt; &lt;/html&gt; </pre>	<pre> &lt;meta charset = "utf-8"/&gt;  &lt;link rel = "stylesheet" href = "css/signup.css"/&gt; &lt;/head&gt;  &lt;body&gt; &lt;div id = "headings"&gt; &lt;h1&gt; Welcome &lt;/h1&gt; &lt;h4&gt; Please sign up now to join our discussion forum &lt;/h4&gt; &lt;/div&gt;  &lt;div id = "signupwindow"&gt; &lt;h3&gt; Member Signup &lt;/h3&gt;  &lt;form method = "POST"&gt; &lt;input type = "text" placeholder = "Enter First Name" name = "fname"/&gt; &lt;br/&gt; &lt;input type = "text" placeholder = "Enter Last Name" name = "lname"/&gt; &lt;br/&gt; &lt;input type = "text" placeholder = "Enter Desired Username" name = "uname"/&gt; &lt;br/&gt; &lt;input type = "password" placeholder = "Enter Desired Password" name = "pwd"/&gt; &lt;br/&gt; &lt;input type = "submit" value = "Register" name = "submitButton"/&gt; &lt;/form&gt;  &lt;a href = '#'&gt; &lt;h5&gt; Already have an account, log in now &lt;/h5&gt; &lt;/a&gt; &lt;/div&gt; &lt;/body&gt; &lt;/html&gt; </pre>
---	---

Now create the php file called “signup.php”, and this will implement the code that will add the information the user inputs in the signup page into the accounts table of the forumaccounts database. Save the signup.php inside the includes folder.

### **signup.php**

```
<?php
/*Since you are already inside the includes folder, since the form action
in the signupPage.php is set to includes/signup.php, you don't have to reference
the includes folder here*/
include_once 'database.php';

//the information the user inputs in the signupPage.php
$firstname = $_POST['fname'];
$lastname = $_POST['lname'];
$email = $_POST['email'];
$username = $_POST['uname'];
$password = $_POST['pwd'];

//adds user information into the accounts table of the forumaccounts database.
//the values are left unspecified, the variables are not shown.
$sql = "insert into accounts(user_firstname, user_lastname, user_email, user_username,
user_password)
      values(?, ?, ?, ?, ?);";

//initializes the database connection
$stmt = mysqli_stmt_init($conn);

//prepares the connection
if(!mysqli_stmt_prepare($stmt, $sql))//if connection fails for whatever reason
{
    echo "SQL error!";
}
else //if function successfully runs
{
    //hashes the password (to prevent hackers from seeing password)
    $hashedPwd = password_hash($password, PASSWORD_DEFAULT);

    //since you have 5 different placeholders inside $sql, you will use 5 's'.
    mysqli_stmt_bind_param($stmt, 'sssss', $firstname, $lastname, $email, $username,
$hashedPwd);

    mysqli_stmt_execute($stmt); //exetures the database connection
```

```

    }

    //takes the user to the front page (loginPage.php) when signup is complete
    header("Location: ../loginPage.php?signup=success");
?>

```

Also, don't forget to add the email text input and set the form action to "includes/signup.php"

```

signupPage.php
<?php
    include_once 'includes/database.php';
?>

<html>
    <head>
        <title> Register an account now! </title>
        <meta charset = "utf-8"/>

        <link rel = "stylesheet" href = "css/signup.css"/>
    </head>

    <body>
        <div id = "headings">
            <h1> Welcome </h1>
            <h4> Please sign up now to join our discussion forum </h4>
        </div>

        <div id = "signupwindow">
            <h3> Member Signup </h3>

            <form method = "POST" action = "includes/signup.php">
                <input type = "text" placeholder = "Enter First Name" name = "fname"/>
                <br/>
                <input type = "text" placeholder = "Enter Last Name" name = "lname"/>
                <br/>
                <input type = "text" placeholder = "Enter your Email" name = "email"/>
                <br/>
                <input type = "text" placeholder = "Enter Desired Username" name = "uname"/>
                <br/>
                <input type = "password" placeholder = "Enter Desired Password" name =
"pwd"/>
                <br/>
                <input type = "submit" value = "Register" name = "submitButton"/>
            </form>

            <a href = '#'> <h5> Already have an account, log in now </h5> </a>
        </div>

```



```
</body>
</html>
```

Now, to test if this works, type into the signup fields:

Firstname: ftest

Lastname: ltest

Email: [test@example.com](mailto:test@example.com)

Username: testusername

Password: testpassword

Now you should see the following in the accounts table:

user_id	user_first name	user_last name	user_email	user_user name	user_password
1	ftest	ltest	test@examp le.com	testuserna me	\$2y\$10\$3zQA.S163L3H/5FTqtwkx.N.2WbD /ZY0uZeJHTj23Q5...

After the successful signup, you should be directed to the loginPage.php with the following URL:

```
http://localhost/Discussion%20Forum/loginPage.php?signup=successful
```

Next is to handle the login process. Create a php file called “login.php”, and this implements the code that will handle the process when users input their username and password to log into the forum page. Save the login.php in the includes folder.

#### **login.php**

```
<?php
if(isset($_POST['loginButton']))
{
    /*Since you are already inside the includes folder, since the form action
    in the loginPage.php is set to includes/login.php, you don't have to reference
    the includes folder here*/
    require 'database.php';

    //obtains the username and password inputs from the login window
    $loginusername = $_POST['loginuname'];
    $loginpassword = $_POST['loginpwd'];

    if(empty($loginusername) && empty($loginpassword)) //if both the username and
    password fields are empty
    {
```

```

    header("Location: ../loginPage.php?error=emptyfields");
    exit();
}
else if(empty($loginusername)) //else if only the username field is empty
{
    header("Location: ../loginPage.php?error=usernameempty");
    exit();
}
else if(empty($loginpassword)) //else if only the password field is empty
{
    header("Location: ../loginPage.php?error=passwordempty");
    exit();
}
else //else if both fields are filled in
{
    //obtains the username and password data from the accounts table of the forumaccounts
table
    $sql = "select * from accounts where user_username=? or user_password=?";

    $stmt = mysqli_stmt_init($conn); //initializes the connection to the forumaccounts
database

    if(!mysqli_stmt_prepare($stmt, $sql)) //if connection fails
    {
        header("Location: ../loginPage.php?error=sqlerror");
        exit();
    }
    else //if connection is fine
    {
        mysqli_stmt_bind_param($stmt, "ss", $loginusername, $loginusername);
        mysqli_stmt_execute($stmt); //executes the database connection

        $result = mysqli_stmt_get_result($stmt); //obtains the results from the database

        //checks if you did get any result from the database
        if($row = mysqli_fetch_assoc($result))
        {
            //takes the password the user used to log in and the password from the database and
see if they match
            $pwdCheck = password_verify($loginpassword, $row['user_password']);

            if($pwdCheck == false) //if the passwords don't match
            {
                header("Location: ../loginPage.php?error=wrongpassword");
                exit();
            }
        }
    }
}

```

```

else if($pwdCheck == true) //else if the passwords do match
{
    session_start(); // you need to start a session to start the login system

    //stores the information in session variables
    $_SESSION['userid'] = $row['user_id'];
    $_SESSION['username'] = $row['user_username'];

    //sends user to the forum homepage upon successful login
    header("Location: ../forumhome.php?login=successful");
    exit();
}
}
else // if there is no user in the database that matches the inputted password and
username
{
    header("Location: ../loginPage.php?error=nouser");
    exit();
}
}
}
else //if the login button is not set, send the user back to the login page
{
    header("Location: ../loginPage.php");
    exit();
}
?>

```

You also want to make sure you have a session started in the loginPage.php and have the form action set to “includes/login.php”

### **loginPage.php**

```

<?php
    include_once 'includes/database.php';

    //you need to start a session in order for the login system to work
    session_start();

?>

<html>
    <head>
        <title> Login to account </title>
        <meta charset = "utf-8"/>

        <link rel = "stylesheet" href = "css/login.css"/>

```

```
<link href="https://fonts.googleapis.com/css?family=Pacifico" rel="stylesheet">
</head>

<body>
  <div id = "headings">
    <h1> Welcome to the Discussion forums </h1>
    <h4> Please login to enter </h4>
  </div>

  <div id = "loginwindow">
    <h4> Member Login </h4>

    <form method = "POST" action = "includes/login.php">
      <input type = "text" placeholder = "Enter Username" name = "loginuname"/>
      <br/>
      <input type = "password" placeholder = "Enter Password" name = "loginpwd"/>
      <br/>
      <input type = "submit" value = "Login" name = "loginButton"/>
    </form>

    <a href = "#"><h4> Forgot password? Click here </h4></a>
  </div>
</body>
</html>
```

Type in any of the usernames and passwords from the accounts table, and it should successfully send you to the forum home page with the following URL:

```
http://localhost/Discussion%20Forum/forumhome.php?login=successful
```

**2/26/19, 2/28/19**

Now you want to resume designing the forum home page and user profile, adding functionality and user-friendly layout before moving on to adding the php code that will allow users to post threads on forums, save messages and posts on to the database, allow users to change information such as name and pics, etc.

First create the layout, then the following day, you can continue building off the layout. Rebuild the forumhome.php and userprofile.php, as well as their respective css files:

<b><u>forumhome.php</u></b> <html> <head> <title> Welcome to forum </title> <meta charset = "utf-8"/>  <link href="https://fonts.googleapis.com /css?family=Lobster" rel="stylesheet"> <link rel = "stylesheet" href = "css/bootstrap.css"/> <link rel = "stylesheet" href = "css/bootstrap.min.css"/> <link rel = "stylesheet" href = "css/main.css"/> </head>  <body> <!-- Navbar --> <nav class = "navbar navbar- inverse"> <div class = "container- fluid"> <div class = "navbar- header"> <a href = "#" class = "navbar-brand"> ForumWebsite </a> </div>  <ul class = "nav navbar-nav">	<b><u>main.css</u></b> body { background-image: url("https://png.pngtree.com/thumb_back/fh260/back_pic /04/00/23/4757fc9e77387db.jpg"); background-repeat: no-repeat; background-size: cover; }  #searchwindow { padding-left: 25%; padding-top: 10%; }  #searchwindow input[type=text] { width: 60%; height: 30px;  background-color: #F5F5DC; font-family: Times New Roman; }  #searchwindow input[type=submit] { width: 60%; height: 30px;  background-color: #F5FFFA; font-family: Times New Roman; }
---	---

<pre>                 &lt;li&gt;&lt;a href = "forumhome.php"&gt; Home &lt;/a&gt;&lt;/li&gt;                 &lt;li&gt;&lt;a href = "#"&gt; Boards &lt;/a&gt;&lt;/li&gt;                 &lt;li&gt;&lt;a href = "#"&gt; Community &lt;/a&gt;&lt;/li&gt;             &lt;/ul&gt;              &lt;ul class = "nav navbar-nav navbar-right"&gt;                 &lt;li&gt;&lt;a href = "#"&gt;&lt;span class = "glyphicon glyphicon-user"&gt;&lt;/span&gt; Logout &lt;/a&gt;&lt;/li&gt;                 &lt;li&gt;&lt;a href = "userprofile.php"&gt; Go to Profile &lt;/a&gt;&lt;/li&gt;             &lt;/ul&gt;         &lt;/div&gt;     &lt;/nav&gt;      &lt;!-- Search for forums --&gt;     &lt;div id = "searchwindow"&gt;         &lt;p&gt; Search for your favorite topic on a subject &lt;/p&gt;          &lt;form&gt;             &lt;input type = "text" name = "subjectsearch" placeholder = "Enter Subject"/&gt;             &lt;br/&gt;             &lt;input type = "text" name = "topicsearch" placeholder = "Enter Topic of subject"/&gt;             &lt;br/&gt;             &lt;input type = "submit" name = "submit-search" value = "Search"/&gt;         &lt;/form&gt;     &lt;/div&gt;      &lt;!-- Forum window --&gt;     &lt;div class = "container" id = "forumwindow"&gt;         &lt;table class = "table"&gt; </pre>	<pre> #searchwindow p {     padding-right: 30%;     font-family: "Lobster", cursive;     font-size: 200%; }  #forumwindow {     padding-top: 80px;     padding-left: 40px; }  #searchwindow p, #forumwindow .table th, #forumwindow .table td {     color: #D3D3D3; }  #forumwindow .table th, #forumwindow .table td {     font-family: Times New Roman; }  #forumwindow .table thead {     background-color: purple; }  #forumwindow .table tbody {     background-color: #2F4F4F; } </pre>
--	---

Subject	Posts	Latest Post
topic of subject 1	this is a thread question	last person who posted
topic of subject 1	this is a thread question	last person who posted

Subject2	Posts	Latest Post
topic of subject 2		

<pre>                 &lt;td&gt; this is a thread question &lt;/td&gt;                 &lt;td&gt; last person who posted &lt;/td&gt;             &lt;/tr&gt;             &lt;tr&gt;                 &lt;td&gt; topic of subject 2 &lt;/td&gt;                 &lt;td&gt; this is a thread question &lt;/td&gt;                 &lt;td&gt; last person who posted &lt;/td&gt;             &lt;/tr&gt;         &lt;/tbody&gt;     &lt;/table&gt; &lt;/div&gt;      &lt;script src = "js/jquery- 3.3.1.js"/&gt;&lt;/script&gt;     &lt;script src = "js/bootstrap.js"/&gt;&lt;/script&gt;     &lt;script src = "js/jquery- ui.min.js"/&gt;&lt;/script&gt;     &lt;script src = "js/main.js"/&gt;&lt;/script&gt; &lt;/body&gt; &lt;/html&gt; </pre>	
---	--

<p><b><u>userprofile.php</u></b></p> <pre> &lt;html&gt;     &lt;head&gt;         &lt;title&gt; Welcome to your profile &lt;/title&gt;         &lt;meta charset = "utf-8"/&gt;          &lt;link href="https://fonts.googleapis.c om/css?family=Molle:400i" rel="stylesheet"&gt;         &lt;link rel = "stylesheet" href = "css/bootstrap.css"/&gt;         &lt;link rel = "stylesheet" href = "css/bootstrap.min.css"/&gt;         &lt;link rel = "stylesheet" href = "css/userprofile.css"/&gt; </pre>	<p><b><u>userprofile.css</u></b></p> <pre> body {     background-color: #000080; }  #topheading {     background-image: url("https://encrypted- tbn0.gstatic.com/images?q=tbn:ANd9GcTQYeeMhSfmq0rZ DQoKPe17Xhn7ytm7fVMWS2imxySICfhMgQ7i");     background-repeat: no-repeat;     background-size: cover; }  #topheading h1 {     text-align: center; </pre>
--	---



<pre> &lt;/head&gt;  &lt;body&gt;   &lt;div id = "topheading"&gt;     &lt;h1&gt; Discussion Forums &lt;/h1&gt;   &lt;/div&gt;    &lt;div id = "tab"&gt;     &lt;nav class = "navbar navbar-inverse"&gt;       &lt;div class = "container-fluid"&gt;         &lt;div class = "navbar-header"&gt;           &lt;a href = "#" class = "navbar-brand"&gt; Your Profile &lt;/a&gt;         &lt;/div&gt;          &lt;ul class = "nav navbar-nav navbar-right"&gt;           &lt;li&gt;&lt;a href = "forumhome.php"&gt; Go to forum &lt;/a&gt;&lt;/li&gt;           &lt;li&gt;&lt;a href = "#"&gt; Settings &lt;/a&gt;&lt;/li&gt;           &lt;li&gt;&lt;a href = "#"&gt; Logout &lt;/a&gt;&lt;/li&gt;         &lt;/ul&gt;       &lt;/div&gt;     &lt;/nav&gt;   &lt;/div&gt;    &lt;div id = "profilewindow"&gt;     &lt;ul class = "nav nav- tabs"&gt;       &lt;li&gt;&lt;a data-toggle = "tab" href = "#profile"&gt; Profile Page &lt;/a&gt;&lt;/li&gt;       &lt;li&gt;&lt;a data-toggle = "tab" href = "#messages"&gt; Messages &lt;/a&gt;&lt;/li&gt; </pre>	<pre> color: white; font-family: 'Molle', cursive; }  #profilewindow {   background-color: #C0C0C0;   height: 50%; }  #profilewindow .tab-content #profile {   box-shadow: 0 4px 8px 0 rgba(0, 0, 0, 0.2);   max-width: 300px;   margin: auto;    background-color: #F5F5DC; }  #profilewindow .tab-content #profile #profile-pic {   margin-left: 35%; } </pre>
---	--

```

        <li><a data-toggle
= "tab" href = "#posts"> Recent
Posts </a></li>
        <li><a data-toggle
= "tab" href = "#friends">
Friends/Acquaintances
</a></li>
    </ul>

    <div class = "tab-
content">
        <div id = "profile"
class = "tab-pane fade in
active">
            <!--
                <p> Profile
picture goes here </p>
                <p> User email
goes here </p>
                <p> Full name
goes here </p>
                <p> Gender
goes here </p>
                <p> Where user
resides goes here </p>
                <p> A short
story about user goes here </p>
            -->
            <img src =
"https://blog.ramboll.com/fehm
arnbelt/wp-
content/themes/ramboll2/image
s/profile-img.jpg" alt = "Profile
Pic" style = "width:30%" id =
"profile-pic"/>
            <h3> Username:
</h3>
            <p class =
"userstats"> Status: </p>
            <p class =
"userstats"> Join Date: </p>
            <p class =
"userstats"> Number of Posts:
</p>
            <p class =
"userstats"> Total Badges: </p>

```

<pre>         &lt;/div&gt;       &lt;/div&gt;     &lt;/div&gt;      &lt;script src = "js/jquery- 3.3.1.js"/&gt;&lt;/script&gt;     &lt;script src = "js/bootstrap.js"/&gt;&lt;/script&gt;     &lt;script src = "js/jquery- ui.min.js"/&gt;&lt;/script&gt;   &lt;/body&gt; &lt;/html&gt; </pre>	
--	--

### **3/8/19, 3/11/19**

You want to edit one part in the forumhome.php:

<pre> <b><u>forumhome.php</u></b> &lt;html&gt;   &lt;head&gt;     &lt;title&gt; Welcome to forum &lt;/title&gt;     &lt;meta charset = "utf-8"/&gt;      &lt;link href="https://fonts.googleapis.com/css?family=Lobster" rel="stylesheet"&gt;     &lt;link rel = "stylesheet" href = "css/bootstrap.css"/&gt;     &lt;link rel = "stylesheet" href = "css/bootstrap.min.css"/&gt;     &lt;link rel = "stylesheet" href = "css/main.css"/&gt;   &lt;/head&gt;    &lt;body&gt;     &lt;!-- Navbar --&gt;     &lt;nav class = "navbar navbar-inverse"&gt;       &lt;div class = "container-fluid"&gt;         &lt;div class = "navbar-header"&gt;           &lt;a href = "#" class = "navbar-brand"&gt; ForumWebsite &lt;/a&gt;         &lt;/div&gt;          &lt;ul class = "nav navbar-nav"&gt;           &lt;li&gt;&lt;a href = "forumhome.php"&gt; Home &lt;/a&gt;&lt;/li&gt;           &lt;li&gt;&lt;a href = "#"&gt; Boards &lt;/a&gt;&lt;/li&gt;           &lt;li&gt;&lt;a href = "#"&gt; Community &lt;/a&gt;&lt;/li&gt;         &lt;/ul&gt;          &lt;ul class = "nav navbar-nav navbar-right"&gt; </pre>
--

```

        <li><a href = "#"><span class = "glyphicon glyphicon-user"></span> Logout
</a></li>
        <li><a href = "userprofile.php"> Go to Profile </a></li>
    </ul>
</div>
</nav>

<!-- Search for forums -->
<div id = "searchwindow">
    <p> Search for your favorite topic on a subject </p>

    <form>
        <input type = "text" name = "subjectsearch" placeholder = "Enter Subject"/>
        <br/>
        <input type = "text" name = "topicsearch" placeholder = "Enter Topic of subject"/>
        <br/>
        <input type = "submit" name = "submit-search" value = "Search"/>
    </form>
</div>

<!-- Forum window -->
<div class = "container" id = "forumwindow">
    <table class = "table">
        <thead>
            <tr>
                <th> Subject </th>
                <th> Latest Post </th>
            </tr>
        </thead>

        <tbody>
            <tr>
                <td> topic of subject 1 </td>
                <td> thread title of topic </br> by (username of user who created the
subject) </td>
            </tr>
            <tr>
                <td> topic of subject 1 </td>
                <td> thread title of topic </br> by (username of user who created the
subject) </td>
            </tr>
        </tbody>
    </table>

    <table class = "table table2">
        <thead>

```

```
<tr>
  <th> Subject2 </th>
  <th> Latest Post </th>
</tr>
</thead>

<tbody>
  <tr>
    <td> topic of subject 2 </td>
    <td> thread title of topic </br> by (username of user who created the
subject) </td>
  </tr>
  <tr>
    <td> topic of subject 2 </td>
    <td> thread title of topic </br> by (username of user who created the
subject) </td>
  </tr>
</tbody>
</table>
</div>

<script src = "js/jquery-3.3.1.js"/></script>
<script src = "js/bootstrap.js"/></script>
<script src = "js/jquery-ui.min.js"/></script>
<script src = "js/main.js"/></script>
</body>
</html>
```

Now, you want to do the following:

Add the subjects of the forum. To keep things simple, use these subjects with the following topics:

<b><u>Subject 1</u></b> Culture	<b><u>Topics</u></b> Philosophy Religion Art
------------------------------------	---

<b><u>Subject 2</u></b> Social Science	<b><u>Topics</u></b> Anthropology Archeology Sociology Psychology Linguistics
---	--

<b><u>Subject 3</u></b> Natural Science	<b><u>Topics</u></b> Biology Chemistry Physics Earth Science Astrology
--	---

<b><u>Subject 4</u></b> Things happening in the real world	<b><u>Topics</u></b> International news
---	--

<b><u>Subject 5</u></b> Tech	<b><u>Topics</u></b> Video games New phones New computers Programs Etc...
---------------------------------	--

<b><u>forumhome.php</u></b> <html> <head> <title> Welcome to forum </title> <meta charset = "utf-8"/>  <link href="https://fonts.googleapis.com/css?family=Lobster" rel="stylesheet"> <link rel = "stylesheet" href = "css/bootstrap.css"/>
---

```

<link rel = "stylesheet" href = "css/bootstrap.min.css"/>
<link rel = "stylesheet" href = "css/main.css"/>
</head>

<body>
<!-- Navbar -->
<nav class = "navbar navbar-inverse">
  <div class = "container-fluid">
    <div class = "navbar-header">
      <a href = "#" class = "navbar-brand"> ForumWebsite </a>
    </div>

    <ul class = "nav navbar-nav">
      <li><a href = "forumhome.php"> Home </a></li>
      <li><a href = "#"> Boards </a></li>
      <li><a href = "#"> Community </a></li>
    </ul>

    <ul class = "nav navbar-nav navbar-right">
      <li><a href = "#"><span class = "glyphicon glyphicon-
user"></span> Logout </a></li>
      <li><a href = "userprofile.php"> Go to Profile </a></li>
    </ul>
  </div>
</nav>

<!-- Search for forums -->
<div id = "searchwindow">
  <p> Search for your favorite topic on a subject </p>

  <form>
    <input type = "text" name = "subjectsearch" placeholder = "Enter
Subject"/>
    <br/>
    <input type = "text" name = "topicsearch" placeholder = "Enter Topic
of subject"/>
    <br/>
    <input type = "submit" name = "submit-search" value = "Search"/>
  </form>
</div>

<!-- Forum window -->
<div class = "container" id = "forumwindow">
  <table class = "table table1">
    <thead>
      <tr>

```

```

        <th> Culture </th>
        <th> Latest Posts </th>
    </tr>
</thead>

<tbody>
    <tr>
        <td> <a href = "#">Philosophy</a> </td>
        <td> thread title of topic </br> by <em>(username of user who
created the subject)</em> </td>
    </tr>
    <tr>
        <td> <a href = "#">Religion</a> </td>
        <td> thread title of topic </br> by <em>(username of user who
created the subject)</em> </td>
    </tr>
    <tr>
        <td> <a href = "#">Art</a> </td>
        <td> thread title of topic </br> by <em>(username of user who
created the subject)</em> </td>
    </tr>
</tbody>
</table>

```

```

<table class = "table table2">
    <thead>
        <tr>
            <th> Social Science </th>
            <th> Latest Posts </th>
        </tr>
    </thead>

    <tbody>
        <tr>
            <td> <a href = "#">Anthropology</a> </td>
            <td> thread title of topic </br> by <em>(username of user who
created the subject)</em> </td>
        </tr>
        <tr>
            <td> <a href = "#">Archeology</a> </td>
            <td> thread title of topic </br> by <em>(username of user who
created the subject)</em> </td>
        </tr>
        <tr>
            <td> <a href = "#">Sociology</a></td>

```



<a href="#">Psychology</a> thread title of topic by (username of user who created the subject)
<a href="#">Linguistics</a> thread title of topic by (username of user who created the subject)

Natural Science and Mathematics Latest Posts	
<a href="#">Biology</a> thread title of topic by (username of user who created the subject)	
<a href="#">Chemistry</a> thread title of topic by (username of user who created the subject)	
<a href="#">Physics</a> thread title of topic by (username of user who created the subject)	
<a href="#">Earth Science</a> thread title of topic by (username of user who created the subject)	

```

        <td> <a href = "#">Astrology</a> </td>
        <td> thread title of topic </br> by <em>(username of user who
created the subject)</em> </td>
    </tr>
    <tr>
        <td> <a href = "#">Mathematics</a> </td>
        <td> thread title of topic </br> by <em>(username of user who
created the subject)</em> </td>
    </tr>
</tbody>
</table>

```

```

<table class = "table table4">
    <thead>
        <tr>
            <th> Things happening outside the Internet </th>
            <th> Latest Posts </th>
        </tr>
    </thead>

    <tbody>
        <tr>
            <td> <a href = "#">International News</a> </td>
            <td> thread title of topic </br> by <em>(username of user who
created the subject)</em> </td>
        </tr>
    </tbody>
</table>

```

```

<table class = "table table5">
    <thead>
        <tr>
            <th> What's going on now in Tech </th>
            <th> Latest Posts </th>
        </tr>
    </thead>

    <tbody>
        <tr>
            <td> <a href = "#">Computer Software/Programs</a> </td>
            <td> thread title of topic </br> by <em>(username of user who
created the subject)</em> </td>
        </tr>
        <tr>
            <td> <a href = "#">Computer Hardware</a> </td>

```

```

        <td> thread title of topic </br> by <em>(username of user who
created the subject)</em> </td>
    </tr>
    <tr>
        <td> <a href = "#">Video Games</a> </td>
        <td> thread title of topic </br> by <em>(username of user who
created the subject)</em> </td>
    </tr>
    <tr>
        <td> <a href = "#">Smartphones and Tablets</a> </td>
        <td> thread title of topic </br> by <em>(username of user who
created the subject)</em> </td>
    </tr>
</tbody>
</table>
</div>

<script src = "js/jquery-3.3.1.js"/></script>
<script src = "js/bootstrap.js"/></script>
<script src = "js/jquery-ui.min.js"/></script>
<script src = "js/main.js"/></script>
</body>
</html>

```

You want to create a separate page for each of the subject topics and an option that allows users to create a new discussion thread of the topic of that particular subject. You want to store all those thread.php files in a folder called “Threads”. Create two files in the Threads file: ThreadsCssFiles (which will contain the files: bootstrap.css and bootstrap.min.css) and ThreadsJSFiles (which will contain the files: bootstrap.js and jquery-3.3.1.js) because for some reason, the topics thread files, which are located in another file (the Threads file of course) cannot access the bootstrap and jQuery files in the css file, so just create another file in the Threads file where the topic threads files are located with the specified bootstrap and jQuery files to link them up easily.

Create the following files in the Threads folder:

```

Philosophy.php, Religion.php, Art.php, Anthropology.php, Archeology.php,
Sociology.php, Psychology.php, Linguistics.php, Biology.php, Chemistry.php,
Physics.php, EarthScience.php, Astronomy.php, Mathematics.php

```

It is time-consuming to create all these files, so for now, all of the above should contain the following code:

```

<html>
  <head>
    <title> Threads </title>

```

```

<meta charset = "utf-8"/>

<link rel = "stylesheet" href = "ThreadsCssFiles/bootstrap.css"/>
<link rel = "stylesheet" href = "ThreadsCssFiles/bootstrap.min.css"/>
<link rel = "stylesheet" href = "ThreadsCssFiles/threads.css"/>
</head>

<body>
  <div class = "container">
    <br/>
    <button> Create new Thread</button>
    <br/><br/>

    <table class = "table" id = "sub-forums">
      <thead>
        <tr>
          <th> Sub-Forums </th>
          <th> Threads/Posts </th>
          <th> Latest Posts</th>
        </tr>
      </thead>

      <tbody>
        <tr>
          <td> Sub-Forum 1 </td>
          <td> Threads: <br/> Posts: </td>
          <td> Last person who posted a thread in this sub-forum </td>
        </tr>
      </tbody>
    </table>

    <br/><br/><br/>

    <table class = "table threads-section" id = "threads-section">
      <thead>
        <tr>
          <th> Title/Threads </th>
          <th> Replies/Views </th>
          <th> Latest Posts </th>
        </tr>
      </thead>

      <tbody>
        <tr>
          <td> Thread title </td>
          <td> Replies: <br/> Views: </td>

```

```

                <td> Last person who posted in this thread </td>
            </tr>
        </tbody>
    </table>
</div>

<script src = "ThreadsJSFiles/jquery-3.3.1.js"/></script>
<script src = "ThreadsJSFiles/bootstrap.js"/></script>
</body>
</html>

```

In some of the thread pages, you want to include a sub-forum, which is basically a section of an online forum dedicated to a specific topic, typically one relating to that of the main forum. For example, in the Natural Science and Mathematics subject section, in the Mathematics topic section, the following sub-forums would be something like: Algebra, Geometry, Calculus, Analysis, etc, all these sub-forums, dedicated to talking about these particular topics.

### 3/13/19

What you want to do now is create the following files in the Threads folder:

**Philosophy.php, Religion.php, Art.php, Anthropology.php, Archeology.php, Sociology.php, Psychology.php, Linguistics.php, Biology.php, Chemistry.php, Physics.php, EarthScience.php, Astronomy.php, Mathematics.php, News.php, CompSoft.php, CompHard.php, VidGm.php, Phones.php**

You want to add the same code below, but change it up a little:

```

<html>
    <head>
        <title> (Topic name) Threads </title>
        <meta charset = "utf-8"/>

        <link rel = "stylesheet" href = "ThreadsCssFiles/bootstrap.css"/>
        <link rel = "stylesheet" href = "ThreadsCssFiles/bootstrap.min.css"/>
        <link rel = "stylesheet" href = "ThreadsCssFiles/threads.css"/>
    </head>

    <body id = (Topic name)>
        <!-- Navbar -->
        <nav class = "navbar navbar-inverse">
            <div class = "container-fluid">
                <div class = "navbar-header">
                    <a href = "#" class = "navbar-brand"> ForumWebsite </a>
                </div>

```

```

        <ul class = "nav navbar-nav">
            <li><a href = "../forumhome.php"> Home </a></li>
            <li><a href = "#"> Boards </a></li>
            <li><a href = "#"> Community </a></li>
        </ul>

        <ul class = "nav navbar-nav navbar-right">
            <li><a href = "#"><span class = "glyphicon glyphicon-user"></span> Logout
</a></li>
            <li><a href = "../userprofile.php"> Go to Profile </a></li>
        </ul>
    </div>
</nav>

<div class = "container">
    <br/>
    <button> Create new Thread</button>
    <br/><br/>

    <table class = "table" id = "sub-forums">
        <thead>
            <tr>
                <th> Sub-Forums </th>
                <th> Threads/Posts </th>
                <th> Latest Posts</th>
            </tr>
        </thead>

        <tbody>
            <tr>
                <td> Sub-Forum 1 </td>
                <td> Threads: <br/> Posts: </td>
                <td> Last person who posted a thread in this sub-forum </td>
            </tr>
        </tbody>
    </table>

    <br/><br/><br/>

    <table class = "table threads-section" id = "threads-section">
        <thead>
            <tr>
                <th> Title/Threads </th>
                <th> Replies/Views </th>
                <th> Latest Posts </th>

```

```

        </tr>
    </thead>

    <tbody>
        <tr>
            <td> Thread title </td>
            <td> Replies: <br> Views: </td>
            <td> Last person who posted in this thread </td>
        </tr>
    </tbody>
</table>
</div>

<script src = "ThreadsJSFiles/jquery-3.3.1.js"/></script>
<script src = "ThreadsJSFiles/bootstrap.js"/></script>
</body>
</html>

```

(replace the “Topic name” with the name of the topic of the particular subject)

The threads.css should be the same for all files in the Threads folder:

```

threads.css
body
{
    background-color: lightblue;
}

.container button
{
    width: 50%;
    height: 30px;
    border: none;
    border-radius: 4px;
    cursor: pointer;
    background-color: #FFDEAD;
    font-family: serif;
}

#sub-forums thead, #threads-section thead
{
    background-color: purple;
}

```

```

#sub-forums tbody, #threads-section tbody
{
    background-color: #2F4F4F;
}

#sub-forums thead th, #sub-forums thead td, #sub-forums tbody th, #sub-forums tbody td,
#threads-section thead th, #threads-section thead td, #threads-section tbody th, #threads-
section tbody td
{
    color: #D3D3D3;
    font-family: Times New Roman;
}

#(Topic name)
{
    background-image: url("(Background URL of Topic Image)");
    background-repeat: no-repeat;
    background-size: cover;
}

```

---

**3/16/19**

You want to implement the php code that allows users to create a new thread topic for that particular subject and add that discussion topic to the most recently added topics in the topics page of that particular subject

Create a php file called “add\_topic\_window.php” and save it to the Discussion Forum folder.

**add\_topic\_window.php**

```

tml>
<head>
    <title> Welcome to forum </title>
    <meta charset = "utf-8"/>

    <link href="https://fonts.googleapis.com/css?family=Lobster" rel="stylesheet">
    <link rel = "stylesheet" href = "css/bootstrap.css"/>
    <link rel = "stylesheet" href = "css/bootstrap.min.css"/>
    <link rel = "stylesheet" href = "css/main.css"/>
</head>

<body>
    <!-- Navbar -->
    <nav class = "navbar navbar-inverse">
        <div class = "container-fluid">

```



```

<div class = "navbar-header">
  <a href = "#" class = "navbar-brand"> ForumWebsite </a>
</div>

<ul class = "nav navbar-nav">
  <li><a href = "forumhome.php"> Home </a></li>
  <li><a href = "#"> Boards </a></li>
  <li><a href = "#"> Community </a></li>
</ul>

<ul class = "nav navbar-nav navbar-right">
  <li><a href = "#"><span class = "glyphicon glyphicon-user"></span> Logout
</a></li>
  <li><a href = "userprofile.php"> Go to Profile </a></li>
</ul>
</div>
</nav>

<!-- Search for forums -->
<div id = "searchwindow">
  <p> Search for your favorite topic on a subject </p>

  <form>
    <input type = "text" name = "subjectsearch" placeholder = "Enter Subject"/>
    <br/>
    <input type = "text" name = "topicsearch" placeholder = "Enter Topic of subject"/>
    <br/>
    <input type = "submit" name = "submit-search" value = "Search"/>
  </form>
</div>

<!-- Forum window -->
<div class = "container" id = "forumwindow">
  <table class = "table table1">
    <thead>
      <tr>
        <th> Culture </th>
        <th> Latest Posts </th>
      </tr>
    </thead>

    <tbody>
      <tr>
        <td> <a href = "Threads/Philosophy.php">Philosophy</a> </td>
        <td> thread title of topic </br> by <em>(username of user who created the
subject)</em> </td>

```

```

        </tr>
        <tr>
            <td> <a href = "Threads/Religion.php">Religion</a> </td>
            <td> thread title of topic </br> by <em>(username of user who created the
subject)</em> </td>
        </tr>
        <tr>
            <td> <a href = "Threads/Art.php">Art</a> </td>
            <td> thread title of topic </br> by <em>(username of user who created the
subject)</em> </td>
        </tr>
    </tbody>
</table>

```

```

<table class = "table table2">
    <thead>
        <tr>
            <th> Social Science </th>
            <th> Latest Posts </th>
        </tr>
    </thead>

    <tbody>
        <tr>
            <td> <a href = "Threads/Anthropology.php">Anthropology</a> </td>
            <td> thread title of topic </br> by <em>(username of user who created the
subject)</em> </td>
        </tr>
        <tr>
            <td> <a href = "Threads/Archeology.php">Archeology</a> </td>
            <td> thread title of topic </br> by <em>(username of user who created the
subject)</em> </td>
        </tr>
        <tr>
            <td> <a href = "Threads/Sociology.php">Sociology</a></td>
            <td> thread title of topic </br> by <em>(username of user who created the
subject)</em> </td>
        </tr>
        <tr>
            <td> <a href = "Threads/Psychology.php">Psychology</a></td>
            <td> thread title of topic </br> by <em>(username of user who created the
subject)</em> </td>
        </tr>
        <tr>
            <td> <a href = "Threads/Linguistics.php">Linguistics</a></td>

```

```

        <td> thread title of topic </br> by <em>(username of user who created the
subject)</em> </td>
    </tr>
</tbody>
</table>

```

```

<table class = "table table3">
    <thead>
        <tr>
            <th> Natural Science and Mathematics </th>
            <th> Latest Posts </th>
        </tr>
    </thead>

    <tbody>
        <tr>
            <td> <a href = "Threads/Biology.php">Biology</a> </td>
            <td> thread title of topic </br> by <em>(username of user who created the
subject)</em> </td>
        </tr>
        <tr>
            <td> <a href = "Threads/Chemistry.php">Chemistry</a> </td>
            <td> thread title of topic </br> by <em>(username of user who created the
subject)</em> </td>
        </tr>
        <tr>
            <td> <a href = "Threads/Physics.php">Physics</a> </td>
            <td> thread title of topic </br> by <em>(username of user who created the
subject)</em> </td>
        </tr>
        <tr>
            <td> <a href = "Threads/EarthScience.php">Earth Science</a> </td>
            <td> thread title of topic </br> by <em>(username of user who created the
subject)</em> </td>
        </tr>
        <tr>
            <td> <a href = "Threads/Astronomy.php">Astronomy</a> </td>
            <td> thread title of topic </br> by <em>(username of user who created the
subject)</em> </td>
        </tr>
        <tr>
            <td> <a href = "Threads/Mathematics.php">Mathematics</a> </td>
            <td> thread title of topic </br> by <em>(username of user who created the
subject)</em> </td>
        </tr>
    </tbody>

```

</table>

<table class = "table table4">

<thead>

<tr>

<th> Things happening outside the Internet </th>

<th> Latest Posts </th>

</tr>

</thead>

<tbody>

<tr>

<td> <a href = "Threads/News.php">International News</a> </td>

<td> thread title of topic </br> by <em>(username of user who created the subject)</em> </td>

</tr>

</tbody>

</table>

<table class = "table table5">

<thead>

<tr>

<th> What's going on now in Tech </th>

<th> Latest Posts </th>

</tr>

</thead>

<tbody>

<tr>

<td> <a href = "Threads/CompSoft.php">Computer Software/Programs</a>

</td>

<td> thread title of topic </br> by <em>(username of user who created the subject)</em> </td>

</tr>

<tr>

<td> <a href = "Threads/CompHard.php">Computer Hardware</a> </td>

<td> thread title of topic </br> by <em>(username of user who created the subject)</em> </td>

</tr>

<tr>

<td> <a href = "Threads/VidGm.php">Video Games</a> </td>

<td> thread title of topic </br> by <em>(username of user who created the subject)</em> </td>

</tr>

<tr>

<td> <a href = "Threads/Phones.php">Smartphones and Tablets</a> </td>

```

                <td> thread title of topic </br> by <em>(username of user who created the
subject)</em> </td>
            </tr>
        </tbody>
    </table>
</div>

<script src = "js/jquery-3.3.1.js"/></script>
<script src = "js/bootstrap.js"/></script>
<script src = "js/jquery-ui.min.js"/></script>
<script src = "js/main.js"/></script>
</body>
</html>

```

### **addWindow.css**

```

#addnewtopic
{
    border: 1px solid black;

    /*pushes the elements close to the center*/
    padding-left: 20%;

    /*creates a scroll bar to keep everything in the border when changing size of browser*/
    overflow: scroll;

    /*formats the left and right borders*/
    margin: 0 10%;
}

#addnewtopic h3
{
    margin-left: 25%;
}

#addnewtopic label
{
    font-family: serif;
}

#addnewtopic input[type=text]
{
    width: 50%;
    height: 30px;
    border: 1px solid #ccc;
    border-radius: 4px;
}

```

```

#addnewtopic select
{
  width: 50%;
  height: 30px;
  border-radius: 4px;
}

#addnewtopic input[type=button]
{
  width: 50%;
  height: 30px;
  border: none;
  border-radius: 4px;
  cursor: pointer;
  background-color: #FAEBD7;
  font-family: serif;
}

```

You also want to add the following to all the thread files in your Threads folder:

```

<html>
  <head>
    <title> (topic name) Threads </title>
    <meta charset = "utf-8"/>

    <link rel = "stylesheet" href = "ThreadsCssFiles/bootstrap.css"/>
    <link rel = "stylesheet" href = "ThreadsCssFiles/bootstrap.min.css"/>
    <link rel = "stylesheet" href = "ThreadsCssFiles/threads.css"/>
  </head>

  <body id = "(topic name)">
    <!-- Navbar -->
    <nav class = "navbar navbar-inverse">
      <div class = "container-fluid">
        <div class = "navbar-header">
          <a href = "#" class = "navbar-brand"> ForumWebsite </a>
        </div>

        <ul class = "nav navbar-nav">
          <li><a href = "../forumhome.php"> Home </a></li>
          <li><a href = "#"> Boards </a></li>
          <li><a href = "#"> Community </a></li>
        </ul>

        <ul class = "nav navbar-nav navbar-right">

```

```

        <li><a href = "#"><span class = "glyphicon glyphicon-user"></span> Logout
</a></li>
        <li><a href = "../userprofile.php"> Go to Profile </a></li>
    </ul>
</div>
</nav>

<div class = "container">
    <br/>
    <a href = "../add_topic_window.php"><button> Create new Thread </button></a>
    <br/><br/>

    <table class = "table" id = "sub-forums">
        <thead>
            <tr>
                <th> Sub-Forums </th>
                <th> Threads/Posts </th>
                <th> Latest Posts</th>
            </tr>
        </thead>

        <tbody>
            <tr>
                <td> Sub-Forum 1 </td>
                <td> Threads: <br/> Posts: </td>
                <td> Last person who posted a thread in this sub-forum </td>
            </tr>
        </tbody>
    </table>

    <br/><br/><br/>

    <table class = "table threads-section" id = "threads-section">
        <thead>
            <tr>
                <th> Title/Threads </th>
                <th> Replies/Views </th>
                <th> Latest Posts </th>
            </tr>
        </thead>

        <tbody>
            <tr>
                <td> Thread title </td>
                <td> Replies: <br/> Views: </td>
                <td> Last person who posted in this thread </td>
            </tr>
        </tbody>
    </table>

```

```
        </tr>
      </tbody>
    </table>
  </div>

  <script src = "ThreadsJSFiles/jquery-3.3.1.js"/></script>
  <script src = "ThreadsJSFiles/bootstrap.js"/></script>
</body>
</html>
```

---

### 3/18/19, 3/25/19

Before moving on, what you want to do now is re-decorate and fix stuff in the `userprofile.php` and `userprofile.css` user profile page and make it more user friendly.

#### **userprofile.php**

```
<html>
  <head>
    <title> Welcome to your profile </title>
    <meta charset = "utf-8"/>

    <link rel = "stylesheet" href = "css/bootstrap.css"/>
    <link rel = "stylesheet" href = "css/bootstrap.min.css"/>
    <link rel = "stylesheet" href = "css/userprofile.css"/>
  </head>

  <body>
    <div id = "tab">
      <nav class = "navbar navbar-inverse">
        <div class = "container-fluid">
          <div class = "navbar-header">
            <a href = "#" class = "navbar-brand"> ForumWebsite </a>
          </div>

          <ul class = "nav navbar-nav">
            <li><a href = "forumhome.php"> Home </a></li>
            <li><a href = "#"> Boards </a></li>
            <li><a href = "#"> Community </a></li>
          </ul>

          <ul class = "nav navbar-nav navbar-right">
            <li><a href = "forumhome.php"> Go to forum </a></li>
            <li><a href = "#"> Settings </a></li>
            <li><a href = "#"> Logout </a></li>
          </ul>
```



```

        </div>
    </nav>
</div>

<div id = "profilewindow">
    <ul class = "nav nav-tabs">
        <li><a data-toggle = "tab" href = "#profile"> Profile Page </a></li>
        <li><a data-toggle = "tab" href = "#messages"> Private Messages </a></li>
        <li><a data-toggle = "tab" href = "#sendmessage"> Send PM </a></li>
        <li><a data-toggle = "tab" href = "#posts"> Recent Threads and Posts </a></li>
    </ul>

    <div class = "tab-content">
        <!-- Profile Page tab -->
        <div id = "profile" class = "tab-pane fade">
            <form id = "userstatsinfo">
                <img src = "https://blog.ramboll.com/fehmarbalt/wp-
content/themes/ramboll2/images/profile-img.jpg" alt = "Profile Pic" style = "width:30%" id =
"profile-pic"/>
                <label class = "userstats"> Username: <input type = "text"/> </label>
                <label class = "userstats"> Email: <input type = "text"/> </label>
                <label class = "userstats"> Status: <input type = "text"/> </label>
                <label class = "userstats"> Join Date: <input type = "text"/> </label>

                <input type = "submit" value = "Save info"/>
            </form>
        </div>

        <!-- Message Inbox tab -->
        <div id = "messages" class = "tab-pane fade">
            <div id = "messageinbox">
                <h4> Your message inbox: </h4>
                <br/>
                <table>
                    <tr>
                        <th> From </th>
                        <th> Subject </th>
                        <th> Topic </th>
                        <th> Date sent </th>
                        <th> Status </th>
                        <th> Message </th>
                    </tr>
                    <tr>
                        <td> from </td>
                        <td> subject </td>
                        <td> topic </td>

```

```

        <td> date sent </td>
        <td> status </td>
        <td> message </td>
    </tr>
</table>
</div>

<br/><br/>

<div id = "sentmessages">
    <h4> Messages you have sent: </h4>
    <br/>
    <table>
        <tr>
            <th> To </th>
            <th> Subject </th>
            <th> Topic </th>
            <th> Date sent </th>
            <th> Status </th>
            <th> Message </th>
        </tr>
        <tr>
            <td> to </td>
            <td> subject </td>
            <td> topic </td>
            <td> date sent </td>
            <td> status </td>
            <td> message </td>
        </tr>
    </table>
</div>
</div>

<!-- Private Message tab -->
<div id = "sendmessage" class = "tab-pane fade in active">
    <h3> Send Private Message </h3>

    <form method = "POST" id = "sendpm">
        <label> Enter username of forum member: <input type = "text"/> </label>
        <br/>
        <label> Enter subject: <input type = "text"/> </label>
        <br/>
        <label> Enter topic: <input type = "text"/> </label>
        <br/><br/>
        <label> Enter Message: </label>
        <br/>

```

```

        <textarea rows = "10" cols = "50"></textarea>
        <br/><br/>
        <input type = "submit" value = "Send PM"/>
    </form>
</div>

<!-- Recent Threads and Posts tab -->
<div id = "posts" class = "tab-pane fade">
    <div id = "listofposts">
        <h4> Number of posts you made: </h4>
        <br/>
        <table>
            <tr>
                <th> Thread title </th>
                <th> Subject </th>
                <th> Lasted posted </th>
            </tr>
            <tr>
                <td> title of thread last posted </td>
                <td> subject of the thread </td>
                <td> date of thread last posted </td>
            </tr>
        </table>
    </div>
</div>
</div>
</div>
</div>

<script src = "js/jquery-3.3.1.js"/></script>
<script src = "js/bootstrap.js"/></script>
<script src = "js/jquery-ui.min.js"/></script>
</body>
</html>

```

### userprofile.css

```

body
{
    background-color: #F8F8FF;
}

#profilewindow
{
    background-color: #C0C0C0;
    height: 70%;
}

```

```

/*handles the Profile Page tab*/
#profilewindow .tab-content #profile
{
    box-shadow: 0 4px 8px 0 rgba(0, 0, 0, 0.2);
    max-width: 300px;
    margin: 20px 0 0 35%;

    background-color: #F5F5DC;
}
#profilewindow .tab-content #profile #userstatsinfo .userstats
{
    font-family: serif;
}
#profilewindow .tab-content #profile #userstatsinfo .userstats input[type=text]
{
    font-family: serif;
    width: 50%;
    height: 30px;
    border: 1px solid #ccc;
    border-radius: 4px;
}
#profilewindow .tab-content #profile #userstatsinfo input[type=submit]
{
    font-family: serif;
    margin-left: 40%;
    background-color: lightgreen;
    border: none;
    border-radius: 3px;
}
#profilewindow .tab-content #profile #profile-pic
{
    margin: 10px 0 20px 35%;
}

/*handles the Message Inbox tab*/
#profilewindow .tab-content #messages #messageinbox
{
    box-shadow: 0 4px 8px 0 rgba(0, 0, 0, 0.2);
    max-width: 80%;
    margin: 20px 0 0 10%;

    font-family: serif;
    text-decoration: underline;
    text-align: center;

    background-color: #F5F5DC;
}

```

```

}
#profilewindow .tab-content #messages #messageinbox table
{
    font-family: arial, sans-serif;
    border-collapse: collapse;
    width: 100%;
}
#profilewindow .tab-content #messages #messageinbox td,
#profilewindow .tab-content #messages #messageinbox th
{
    border: 1px solid #dddddd;
    text-align: left;
    padding: 8px;
}
#profilewindow .tab-content #messages #sentmessages
{
    box-shadow: 0 4px 8px 0 rgba(0, 0, 0, 0.2);
    max-width: 80%;
    margin: 20px 0 0 10%;

    font-family: serif;
    text-decoration: underline;
    text-align: center;

    background-color: #F5F5DC;
}
#profilewindow .tab-content #messages #sentmessages table
{
    font-family: arial, sans-serif;
    border-collapse: collapse;
    width: 100%;
}
#profilewindow .tab-content #messages #sentmessages td,
#profilewindow .tab-content #messages #sentmessages th
{
    border: 1px solid #dddddd;
    text-align: left;
    padding: 8px;
}

/*handles the Private Page tab*/
#profilewindow .tab-content #sendmessage
{
    box-shadow: 0 4px 8px 0 rgba(0, 0, 0, 0.2);
    max-width: 90%;
    margin-left: 5%;
}

```

```

    background-color: #F5F5DC;
}
#profilewindow .tab-content #sendmessage h3
{
    text-align: center;
}
#profilewindow .tab-content #sendmessage #sendpm
{
    margin-left: 10px;
}
#profilewindow .tab-content #sendmessage #sendpm input[type=text]
{
    font-family: serif;
    width: 50%;
    height: 30px;
    border: 1px solid #ccc;
    border-radius: 4px;
}
#profilewindow .tab-content #sendmessage #sendpm textarea
{
    border: 1px solid #ccc;
    border-radius: 4px;
}
#profilewindow .tab-content #sendmessage #sendpm input[type=submit]
{
    font-family: serif;
    margin-left: 40%;
    background-color: lightgreen;
    border: none;
    border-radius: 3px;
}

/*handles the Recent Threads and Posts tab*/
#profilewindow .tab-content #posts #listofposts
{
    box-shadow: 0 4px 8px 0 rgba(0, 0, 0, 0.2);
    max-width: 80%;
    margin: 20px 0 0 10%;

    font-family: serif;
    text-decoration: underline;
    text-align: center;

    background-color: #F5F5DC;
}

```

```
#profilewindow .tab-content #posts #listofposts table
{
    font-family: arial, sans-serif;
    border-collapse: collapse;
    width: 100%;
}
#profilewindow .tab-content #posts #listofposts td,
#profilewindow .tab-content #posts #listofposts th
{
    border: 1px solid #dddddd;
    text-align: left;
    padding: 8px;
}
```

---

*Handling the background processes and database management*

---

**3/27/19**

Now you want to change a lot of stuff. You want to change loginPage.php and sigupPage.php, and delete the login.php and signup.php in the includes folder, then change all the background processes happen in one page. This way, you don't have to simultaneously switch between files, and you can have all processes happen in one file.

**loginPage.php**

```
<?php
    require 'includes/database.php';

    $loginusername = $loginpassword = "";

    //error message variable
    $printErr = "";

    if($_SERVER["REQUEST_METHOD"] == "POST")
    {
        if(isset($_POST["loginButton"]))
        {
            //if both fields are empty
            if(empty($_POST["loginuname"]) && empty($_POST["loginpwd"]))
            {
```

```

    $printErr = "Do not leave anything blank and then press the login button!";
}
//if the username field is empty
else if(empty($_POST['loginuname']))
{
    $printErr = "You left the username field blank!";
}
//if the password field is empty
else if(empty($_POST['loginpwd']))
{
    $printErr = "You left the password field blank!";
}
//else if everything is filled out
else
{
    //obtains the username and login input from the loginwindow form
    $loginusername = $_POST['loginuname'];
    $loginpassword = $_POST['loginpwd'];

    //obtains the username and password data from the accounts table of the
forumaccounts table
    $sql = "select * from accounts where user_username=? or user_password=?";

    $stmt = mysqli_stmt_init($conn); //initializes the connection to the forumaccounts
database

    if(!mysqli_stmt_prepare($stmt, $sql)) //if connection fails
    {
        header("Location: loginPage.php?error=sqlerror");
        exit();
    }
    else //if connection is fine
    {
        mysqli_stmt_bind_param($stmt, "ss", $loginusername, $loginusername);
        mysqli_stmt_execute($stmt); //executes the database connection

        $result = mysqli_stmt_get_result($stmt); //obtains the results from the database

        //checks if you did get any result from the database
        if($row = mysqli_fetch_assoc($result))
        {
            //takes the password the user used to log in and the password from the
database and see if they match
            $pwdCheck = password_verify($loginpassword, $row['user_password']);

            if($pwdCheck == false) //if the passwords don't match

```



```

        {
            $printErr = "Password is wrong!";
        }
        else if($pwdCheck == true) //else if the passwords do match
        {
            session_start(); // you need to start a session to start the login system

            //stores the information in session variables
            $_SESSION['userid'] = $row['user_id'];
            $_SESSION['username'] = $row['user_username'];

            //sends user to the forum homepage upon successful login
            header("Location: forumhome.php?login=successful");
            exit();
        }
    }
    else // if there is no user in the database that matches the inputted password and
username
    {
        $printErr = "There is no such account!";    }
    }
}
}
?>

<html>
<head>
    <title> Login to account </title>
    <meta charset = "utf-8"/>

    <link rel = "stylesheet" href = "css/login.css"/>
    <link href="https://fonts.googleapis.com/css?family=Pacifico" rel="stylesheet">
</head>

<body>
    <div id = "headings">
        <h1> Welcome to the Discussion forums </h1>
        <h4> Please login to enter </h4>
    </div>

    <div id = "loginwindow">
        <h4> Member Login </h4>

        <form method = "POST" action = "<?php echo
htmlspecialchars($_SERVER['PHP_SELF']);?>">

```

```

        <input type = "text" placeholder = "Enter Username" name = "loginuname"/>
        <br/>
        <input type = "password" placeholder = "Enter Password" name = "loginpwd"/>
        <br/>
        <input type = "submit" value = "Login" name = "loginButton"/>
    </form>

    <p id = "errorMessage"> <?php echo $printErr; ?> </p>

    <a href = "#"><h4> Forgot password? Click here </h4></a>
    <a href = "signupPage.php"><h4> Already have an account? Click here to sign in
</hr></a>
    </div>
</body>
</html>

```

### signupPage.php

```

<?php
    require 'includes/database.php';

    $firstname = $lastname = $email = $username = $password = '';

    //error message variable
    $printErr = '';

    if($_SERVER["REQUEST_METHOD"] == "POST")
    {
        if(isset($_POST["submitButton"]))
        {
            if(empty($_POST["fname"]) && empty($_POST["lname"]) &&
empty($_POST["email"]) && empty($_POST["uname"]) && empty($_POST["pwd"]))
            {
                $printErr = "Do not leave anything blank!";
            }
            else if(empty($_POST["fname"]))
            {
                $printErr = "You left the first name field blank!";
            }
            else if(empty($_POST["lname"]))
            {
                $printErr = "You left the last name field blank!";
            }
            else if(empty($_POST["email"]))
            {
                $printErr = "You left the email field blank!";
            }
        }
    }

```

```

else if(empty($_POST['uname']))
{
    $printErr = "You left the username field blank!";
}
else if(empty($_POST['pwd']))
{
    $printErr = "You left the password field blank!";
}
else
{
    //the information the user inputs in the signupPage.php
    $firstname = $_POST['fname'];
    $lastname = $_POST['lname'];
    $email = $_POST['email'];
    $username = $_POST['uname'];
    $password = $_POST['pwd'];

    //adds user information into the accounts table of the forumaccounts database.
    //the values are left unspecified, the variables are not shown.
    $sql = "insert into accounts(user_firstname, user_lastname, user_email,
user_username, user_password)
        values(?, ?, ?, ?, ?)";

    //initializes database connection
    $stmt = mysqli_stmt_init($conn);

    //prepares the connection
    if(!mysqli_stmt_prepare($stmt, $sql))//if connection fails for whatever reason
    {
        $printErr = "SQL error!";
    }
    else //if function successfully runs
    {
        //hashes the password (to prevent hackers from seeing password)
        $hashedPwd = password_hash($password, PASSWORD_DEFAULT);

        //since you have 5 different placeholders inside $sql, you will use 5 's'.
        mysqli_stmt_bind_param($stmt, 'sssss', $firstname, $lastname, $email,
$username, $hashedPwd);

        mysqli_stmt_execute($stmt); //executes the database connection

        //takes the user to the front page (loginPage.php) when signup is complete
        header("Location: loginPage.php?signup=success");
    }
}
}

```

```

    }
  }
?>

<html>
  <head>
    <title> Register an account now! </title>
    <meta charset = "utf-8"/>

    <link rel = "stylesheet" href = "css/signup.css"/>
  </head>

  <body>
    <div id = "headings">
      <h1> Welcome </h1>
      <h4> Please sign up now to join our discussion forum </h4>
    </div>

    <div id = "signupwindow">
      <h3> Member Signup </h3>

      <form method = "POST" action = "<?php echo
htmlspecialchars($_SERVER["PHP_SELF"]);?>">
        <input type = "text" placeholder = "Enter First Name" name = "fname"/>
        <br/>
        <input type = "text" placeholder = "Enter Last Name" name = "lname"/>
        <br/>
        <input type = "text" placeholder = "Enter your Email" name = "email"/>
        <br/>
        <input type = "text" placeholder = "Enter Desired Username" name = "uname"/>
        <br/>
        <input type = "password" placeholder = "Enter Desired Password" name =
"pwd"/>
        <br/>
        <input type = "submit" value = "Register" name = "submitButton"/>
      </form>

      <a href = 'loginPage.php'> <h5> Already have an account, log in now </h5> </a>

      <p id = "errorMessage"> <?php echo $printErr; ?> </p>
    </div>
  </body>
</html>

```

**3/29/19, 4/1/19**

For the black nav bar on top for the forumhome.php, the userprofile.php and the threads, you want to change the rightmost nav links with the following code:

```
...
    <ul class = "nav navbar-nav navbar-right">
        <li><a href = "userprofile.php"> Settings </a></li>
        <li><a href = "#"><span class = "glyphicon glyphicon-user"></span> Logout
</a></li>
        <li class = "dropdown">
            <a class = "dropdown-toggle" data-toggle = "dropdown" href = "#"> Go
to Profile <span class = "caret"></span> </a>
            <ul class = "dropdown-menu">
                <li><a href = "#"> Profile Pic </a></li>
                <li><a href = "#"> Full Name </a></li>
                <li><a href = "#"> Status </a></li>
                <li><a href = "#"> Join date </a></li>
            </ul>
        </li>
    </ul>
...
```

For the threads files in the Threads folder, you want to add the following code to access the userprofile.php:

```
<ul class = "nav navbar-nav navbar-right">
    <li><a href = "../userprofile.php"> Settings </a></li>
    <li><a href = "#"><span class = "glyphicon glyphicon-user"></span> Logout
</a></li>
    <li class = "dropdown">
        <a class = "dropdown-toggle" data-toggle = "dropdown" href = "#"> Go to
Profile <span class = "caret"></span> </a>
        <ul class = "dropdown-menu">
            <li><a href = "#"> Profile Pic </a></li>
            <li><a href = "#"> Full Name </a></li>
            <li><a href = "#"> Status </a></li>
            <li><a href = "#"> Join date </a></li>
        </ul>
    </li>
</ul>
```

You also want to fix a few stuff both in he signup.css and signupPage.php, primarily making a proper success message when the user has successfully registered an account

### **signupPage.php**

```
<?php
require 'includes/database.php';

$firstname = $lastname = $email = $username = $password = "";

//error message variable
$printErr = "";

//success message
$printSucc = "";

if($_SERVER["REQUEST_METHOD"] == "POST")
{
    if(isset($_POST["submitButton"]))
    {
        if(empty($_POST["fname"]) && empty($_POST["lname"]) &&
empty($_POST["email"]) && empty($_POST["uname"]) && empty($_POST["pwd"]))
        {
            $printErr = "Do not leave anything blank!";
        }
        else if(empty($_POST["fname"]))
        {
            $printErr = "You left the first name field blank!";
        }
        else if(empty($_POST["lname"]))
        {
            $printErr = "You left the last name field blank!";
        }
        else if(empty($_POST["email"]))
        {
            $printErr = "You left the email field blank!";
        }
        else if(empty($_POST["uname"]))
        {
            $printErr = "You left the username field blank!";
        }
        else if(empty($_POST["pwd"]))
        {
            $printErr = "You left the password field blank!";
        }
        else
        {
            //the information the user inputs in the signupPage.php
```

```

$firstname = $_POST['fname'];
$lastname = $_POST['lname'];
$email = $_POST['email'];
$username = $_POST['uname'];
$password = $_POST['pwd'];

//adds user information into the accounts table of the forumaccounts database.
//the values are left unspecified, the variables are not shown.
$sql = "insert into accounts(user_firstname, user_lastname, user_email, user_username,
user_password)
      values(?, ?, ?, ?, ?)";

//initializes database connection
$stmt = mysqli_stmt_init($conn);

//prepares the connection
if(!mysqli_stmt_prepare($stmt, $sql))//if connection fails for whatever reason
{
    $printErr = "SQL error!";
}
else //if function successfully runs
{
    //hashes the password (to prevent hackers from seeing password)
    $hashedPwd = password_hash($password, PASSWORD_DEFAULT);

    //since you have 5 different placeholders inside $sql, you will use 5 's'.
    mysqli_stmt_bind_param($stmt, 'sssss', $firstname, $lastname, $email, $username,
$hashedPwd);

    mysqli_stmt_execute($stmt); //executes the database connection

    //upon successful signup, output a message
    $printSucc = "You have successfully signed up! Sign in now to enter forum
discussion!";
}
}
}
}
?>

<html>
<head>
<title> Register an account now! </title>
<meta charset = "utf-8"/>

<link rel = "stylesheet" href = "css/signup.css"/>

```

```

</head>

<body>
  <div id = "headings">
    <h1> Welcome </h1>
    <h4> Please sign up now to join our discussion forum </h4>
  </div>

  <div id = "signupwindow">
    <h3> Member Signup </h3>

    <form method = "POST" action = "<?php echo
htmlspecialchars($_SERVER["PHP_SELF"]);?>">
      <input type = "text" placeholder = "Enter First Name" name = "fname"/>
      <br/>
      <input type = "text" placeholder = "Enter Last Name" name = "lname"/>
      <br/>
      <input type = "text" placeholder = "Enter your Email" name = "email"/>
      <br/>
      <input type = "text" placeholder = "Enter Desired Username" name = "uname"/>
      <br/>
      <input type = "password" placeholder = "Enter Desired Password" name =
"pwd"/>
      <br/>
      <input type = "submit" value = "Register" name = "submitButton"/>
    </form>

    <a href = 'loginPage.php'> <h5> Already have an account, log in now </h5> </a>

    <p id = "errorMessage"> <?php echo $printErr; ?> </p>
    <p id = "successmessage"> <?php echo $printSucc; ?></p>
  </div>
</body>
</html>

```

### signup.css

```

body
{
  background-color: #191970;
}

#headings
{
  text-align: center;
}

```



```
margin-top: 10%;
}

#headings h1, #headings h4
{
  color: #DCDCDC;
  font-family: cursive;
}

#signupwindow
{
  border: 1px solid black;

  margin-top: 5%;
  margin-right: 20%;
  margin-left: 20%;

  background-image:
url("https://png.pngtree.com/thumb_back/fh260/back_pic/04/00/23/4757fc9e77387db.jpg");
  background-repeat: no-repeat;
  background-size: cover;

  box-shadow: 5px black;
}

#signupwindow h3
{
  color: #FAF0E6;
  text-align: center;
}

#signupwindow form
{
  padding-top: 10px;
  text-align: center;
}

#signupwindow form input[type=text], #signupwindow form input[type=password]
{
  width: 45%;
  height: 7%;

  font-size: 20px;
  font-family: "Times New Roman";
  background-color: #FAF0E6;
```

```
border-radius: 5px;
}

#signupwindow form input[type=submit]
{
    width: 20%;
    height: 5%;

    font-family: cursive, "Times New Roman";
    font-size: 20px;
    color: #F0FFFF;

    border: none;
    border-radius: 5px;

    cursor: pointer;

    background-color: #4169E1;
}

#signupwindow h5
{
    text-align: center;
    color: #F0FFF0;
}

#signupwindow #errormessage
{
    text-align: center;
    color: red;
}

#signupwindow #successmessage
{
    text-align: center;
    color: lightgreen;
}
```

Then you want to fix the whole login process, such as staying login while switching between tabs, checking if user is logged in, in order to actually browse the entire site, making sure the profile is unique to the user, configuring the mailbox system and making sure that mail data stays there when user logs off.

First is to make sure that when the user is logged in, he/she stays logged on to all the pages of the forum, making sure the profile is unique to all users, save all information until the user is logged off. In the forumhome.php and userprofile.php and add\_topic\_window.php, input the following:

```
<?php
    session_start();
?>
...
    <ul class = "nav navbar-nav navbar-right">
        <li><a href = "userprofile.php"> Settings </a></li>
        <li><a href = "#"><span class = "glyphicon glyphicon-user"></span> Logout
    </a></li>
        <li class = "dropdown">
            <a class = "dropdown-toggle" data-toggle = "dropdown" href = "#">
                Hello: <?php echo $_SESSION['username']; ?> <span class =
"caret"></span> </a>
                <ul class = "dropdown-menu">
                    <li><a href = "#"> Profile Pic </a></li>
                    <li><a href = "#"> Full Name </a></li>
                    <li><a href = "#"> Status </a></li>
                    <li><a href = "#"> Join date </a></li>
                </ul>
            </li>
        </ul>
    ...
```

Remember the \$\_SESSION["username"] session variable in the loginPage.php. This is crucial in order to successfully keep the user logged on all pages until the users logs off.

In the threads, input the following on each thread php file

```
<?php
    session_start();
?>
...
<!-- Navbar -->
    <nav class = "navbar navbar-inverse">
        <div class = "container-fluid">
            <div class = "navbar-header">
                <a href = "#" class = "navbar-brand"> Discussion Forum </a>
            </div>
```

```

        <ul class = "nav navbar-nav">
            <li><a href = "../forumhome.php"> Home </a></li>
            <li><a href = "#"> Boards </a></li>
            <li><a href = "#"> Community </a></li>
        </ul>

        <ul class = "nav navbar-nav navbar-right">
            <li><a href = "../userprofile.php"> Settings </a></li>
            <li><a href = "#"><span class = "glyphicon glyphicon-user"></span> Logout
</a></li>
            <li class = "dropdown">
                <a class = "dropdown-toggle" data-toggle = "dropdown" href = "#">
                    Hello: <?php echo $_SESSION['username']; ?> <span class =
"caret"></span> </a>
                <ul class = "dropdown-menu">
                    <li><a href = "#"> Profile Pic </a></li>
                    <li><a href = "#"> Full Name </a></li>
                    <li><a href = "#"> Status </a></li>
                    <li><a href = "#"> Join date </a></li>
                </ul>
            </li>
        </ul>
    </div>
</nav>

```

Next is to create the logout.php file, which will delete all session variables, thus logging out the user from the forum. Save the logout.php file in the includes file.

```

logout.php
<?php
    session_start();

    //remove all session variables
    session_unset();

    //destroy all session_variables
    session_destroy();

    //sends user back to the login page upon successful logout
    header("Location: ../loginPage.php?logout=successful");
?>

```

Now in the forumhome.php, userprofile.php, and add\_topic\_window.php, add the following to the nav right:

```

...
        <ul class = "nav navbar-nav navbar-right">
            <li><a href = "userprofile.php"> Settings </a></li>

```

```

        <li><a href = "includes/logout.php"><span class = "glyphicon glyphicon-
user"></span> Logout </a></li>
        <li class = "dropdown">
            <a class = "dropdown-toggle" data-toggle = "dropdown" href = "#">
                Hello: <?php echo $_SESSION['username']; ?> <span class =
"caret"></span> </a>
            <ul class = "dropdown-menu">
                <li><a href = "#"> Profile Pic </a></li>
                <li><a href = "#"> Full Name </a></li>
                <li><a href = "#"> Status </a></li>
                <li><a href = "#"> Join date </a></li>
            </ul>
        </li>
    </ul>
    ...

```

Now, in the threads php files, add the following:

```

...
    <ul class = "nav navbar-nav navbar-right">
        <li><a href = "../userprofile.php"> Settings </a></li>
        <li><a href = "../includes/logout.php"><span class = "glyphicon glyphicon-
user"></span> Logout </a></li>
        <li class = "dropdown">
            <a class = "dropdown-toggle" data-toggle = "dropdown" href = "#">
                Hello: <?php echo $_SESSION['username']; ?> <span class =
"caret"></span> </a>
            <ul class = "dropdown-menu">
                <li><a href = "#"> Profile Pic </a></li>
                <li><a href = "#"> Full Name </a></li>
                <li><a href = "#"> Status </a></li>
                <li><a href = "#"> Join date </a></li>
            </ul>
        </li>
    </ul>
    ...

```

Now what you want to do is add a few more session variables in the loginPage.php, needed to set up the user profile, so in the loginPage.php, add the following:

#### loginPage.php

```
...
    //obtains the username and password data from the accounts table of the
forumaccounts table
    $sql = "select * from accounts where user_username=? or user_password=? and
user_firstname or user_lastname;";
...
    else if($pwdCheck == true) //else if the passwords do match
    {
        session_start(); // you need to start a session to start the login system

        //stores the information in session variables
        $_SESSION['userid'] = $row['user_id']; //id of user
        $_SESSION['username'] = $row['user_username']; //username of user
        $_SESSION['firstname'] = $row['user_firstname']; //firstname of user
        $_SESSION['lastname'] = $row['user_lastname']; //lastname of user

        //sends user to the forum homepage upon successful login
        header("Location: forumhome.php?login=successful");
        exit();
    }
...
```

Now, in the userprofile.php, forumhome.php, and add\_topic\_window.php, add the following code:

```
...
    <ul class = "nav navbar-nav navbar-right">
        <li><a href = "userprofile.php"> Settings </a></li>
        <li><a href = "includes/logout.php"><span class = "glyphicon glyphicon-
user"></span> Logout </a></li>
        <li class = "dropdown">
            <a class = "dropdown-toggle" data-toggle = "dropdown" href = "#">
                Hello: <?php echo $_SESSION['username']; ?> <span class =
"caret"></span> </a>
            <ul class = "dropdown-menu">
                <li><a href = "#"> Profile Pic </a></li>
                <li><a href = "#"> <?php echo $_SESSION['firstname'] . " " .
$_SESSION['lastname'] ?> </a></li>
                <li><a href = "#"> Status </a></li>
                <li><a href = "#"> Join date </a></li>
            </ul>
        </li>
    </ul>
...
```

...and in the threads files, add the following:

```
...
    <ul class = "nav navbar-nav navbar-right">
        <li><a href = "../userprofile.php"> Settings </a></li>
        <li><a href = "../includes/logout.php"><span class = "glyphicon glyphicon-
user"></span> Logout </a></li>
        <li class = "dropdown">
            <a class = "dropdown-toggle" data-toggle = "dropdown" href = "#">
                Hello: <?php echo $_SESSION['username']; ?> <span class =
"caret"></span> </a>
            <ul class = "dropdown-menu">
                <li><a href = "#"> Profile Pic </a></li>
                <li><a href = "#"> <?php echo $_SESSION['firstname'] . " " .
$_SESSION['lastname'] ?> </a></li>
                <li><a href = "#"> Status </a></li>
                <li><a href = "#"> Join date </a></li>
            </ul>
        </li>
    </ul>
...
```

#### **4/4/19 – 4/8/19**

Next, you want to implement the code for the profile picture, the email, password and status parts, where users can change their profile picture, their username, login password, email and status information.

First, you'll want to handle the profile pics bit. Create a file called "uploadPics.php" and save it in the includes folder, then add the following:

##### **uploadPics.php**

```
<?php
//obtains the photo to be updated
$photo = "";

//obtain the type of file (".jpg", ".png", etc...)
$imageFileType = "";

//error messages
$printErr = "";
$printSucc = "";

if(isset($_POST["submit-pic"]))
{
    //the folder to store the uploaded pictures
    $folder = "ProfilePics/";

    //obtains the photo to be updated and store them in the ProfilePics folder
    $photo = $folder . basename($_FILES["getFile"]["name"]);

    //obtain the type of file (".jpg", ".png", etc...)
    $imageFileType = strtolower(pathinfo($photo, PATHINFO_EXTENSION));

    //checks the size of the image
    $check = getimagesize($_FILES["getFile"]["tmp_name"]);

    //check if the file is a real image or not
    if($check !== false)
    {
        //if the image file is not jpg, jpeg, png and gif files
        if($imageFileType != "jpg" && $imageFileType != "png" && $imageFileType !=
"jpeg" && $imageFileType != "gif")
        {
            $printErr = "Sorry, only JPG, JPEG, PNG, and GIF files are allowed!";
        }
        //else if the image file is a jpg, jpeg, png, and/or gif
        else
        {
            //if file is too large
```



```

if($_FILES["getFile"]["size"] > 500000)
{
    $printErr = "Sorry, your file is too big to upload!";
}
//else if file is not too large
else
{
    //upload image
    if(move_uploaded_file($_FILES["getFile"]["tmp_name"], $photo))
    {
        include 'includes/database.php';

        $x = intval($_SESSION['userid']); //intval gets the integer value (the id)
        $insertsql = "update accounts set profile_picture='$photo' where user_id=$x";

        if(mysqli_query($conn, $insertsql))
        {
            $printSucc = "\"$photo\" successfully uploaded!";
        }
        else
        {
            $printErr = "Error updating data: " . mysqli_error($conn);
        }
    }
    //if image could not be uploaded
    else
    {
        $printErr = "Sorry, there was a problem uploading your image file";
    }
}
}
}
//else if the image is not a real image
else
{
    $printErr = "Image is not a file!";
}
}
?>

```

You want to create a folder called “ProfilePics”, and this folder will be where all the uploaded pictures will be stored.

Before you do anything else, what you now want to do is create a new column in the accounts table of the forumaccounts database called “profile\_picture”, and this column will store the location of the photo that was uploaded (you cannot store a whole picture in a database, so the best way is to store the location of the picture instead). Go to localhost/phpMyAdmin, go to the account table of the forumaccount database and add the following SQL code:

```
alter table accounts
add profile_picture varchar(255)
```

The accounts table should now look like this:

user_id	user_firstname	user_lastname	user_email	user_username	user_password	profile_picture
1	ftest	ltest	test@example.com	testusername	\$2y\$10\$3zQA.S163L3H/5FTqtwkx.N.2WbD/ZY0uZeJHTj23Q5...	NULL

In the loginPage.php, you want to add the following session variable:

#### loginPage.php

```
...
    session_start(); // you need to start a session to start the login system

    /*obtain the id, username, email, firstname, lastname and password as well as
profile picture
    and status and stores them in session variables*/
    $_SESSION['userid'] = $row['user_id']; //id of user (intval gets the integer value)
    $_SESSION['useremail'] = $row['user_email']; //email of user
    $_SESSION['username'] = $row['user_username']; //username of user
    $_SESSION['pwd'] = $row['user_password']; //password of user
    $_SESSION['firstname'] = $row['user_firstname']; //firstname of user
    $_SESSION['lastname'] = $row['user_lastname']; //lastname of user
    $_SESSION['pic'] = $row['profile_picture']; //path of the uploaded profile
picture
    $_SESSION['status'] = $row['user_status']; //status of user

    //sends user to the forum homepage upon successful login
    header("Location: forumhome.php?login=successful");
    exit();
...
```

A side note, this took almost 4 days to figure out why it was not working, until I found the error, which was I has two forms inside another form in the userprofile.php like this:

**userprofile.php**

```
...
<form id = "userinfo">
    <?php
        include 'includes/uploadPics.php';
    ?>

    <form action = '<?php echo htmlspecialchars($_SERVER["PHP_SELF"]);?>' method =
'POST' enctype = 'multipart/form-data'>
        <h3> Select image to upload: </h3>

        <input type = 'file' name = 'getFile'/>
        <input type = 'submit' value = 'Upload Photo' name = 'submit-pic'/>
    </form>

    <p> <?php echo $printErr; ?></p>
    <p> <?php echo $printSucc; ?></p>

    <form method = "POST">
        <label class = "userstats"> Username: <input type = "text"/> </label>
        <label class = "userstats"> Password: <input type = "text"/> </label>
        <label class = "userstats"> Email: <input type = "text"/> </label>
        <label class = "userstats"> Current Status: <input type = "text"/> </label>
        <input type = "submit" value = "Set info"/>
    </form>
</form>
...
```

This will not work. Delete the “form id = ‘userinfo’” element and leave the two forms.

So in the userprofile.php, input the following:

**userprofile.php**

```
<?php
    session_start(); //needed for starting session

    include 'includes/uploadPics.php'; //needed for uploading pictures
?>

<html>
    <head>
        <title> Welcome to your profile </title>
        <meta charset = "utf-8"/>
```

```

<link rel = "stylesheet" href = "css/bootstrap.css"/>
<link rel = "stylesheet" href = "css/bootstrap.min.css"/>
<link rel = "stylesheet" href = "css/userprofile.css"/>
</head>

<body>
<div id = "tab">
<nav class = "navbar navbar-inverse">
<div class = "container-fluid">
<div class = "navbar-header">
<a href = "#" class = "navbar-brand"> Discussion Forum </a>
</div>

<ul class = "nav navbar-nav">
<li><a href = "forumhome.php"> Home </a></li>
<li><a href = "#"> Boards </a></li>
<li><a href = "#"> Community </a></li>
</ul>

<ul class = "nav navbar-nav navbar-right">
<li><a href = "userprofile.php"> Settings </a></li>
<li><a href = "includes/logout.php"><span class = "glyphicon glyphicon-
user"></span> Logout </a></li>
<li class = "dropdown">
<a class = "dropdown-toggle" data-toggle = "dropdown" href = "#">
Hello: <?php echo $_SESSION['username']; ?> <span class =
"caret"></span> </a>
<ul class = "dropdown-menu">
<li><a href = "#"> <?php echo "<img src = '' . $_SESSION['pic'] .
'' height='70' width='70' />"; ?> </a></li>
<li><a href = "#"> <?php echo $_SESSION['firstname'] . " " .
$_SESSION['lastname'] ?> </a></li>
<li><a href = "#"> Email </a></li>
<li><a href = "#"> Join date </a></li>
</ul>
</li>
</ul>
</div>
</nav>
</div>

<div id = "profilewindow">
<ul class = "nav nav-tabs">
<li><a data-toggle = "tab" href = "#profile"> Profile Page </a></li>
<li><a data-toggle = "tab" href = "#messages"> Private Messages </a></li>
<li><a data-toggle = "tab" href = "#sendmessage"> Send PM </a></li>

```

```

        <li><a data-toggle = "tab" href = "#posts"> Recent Threads and Posts </a></li>
    </ul>

    <div class = "tab-content">
        <!-- Profile Page tab -->
        <div id = "profile" class = "tab-pane fade in active">
            <form action = '<?php echo
htmlspecialchars($_SERVER["PHP_SELF"]);?>' method = 'POST' enctype =
'multipart/form-data'>
                <h3> Select image to upload: </h3>

                <input type = 'file' name = 'getFile' />
                <input type = 'submit' value = 'Upload Photo' name = 'submit-pic' />
            </form>

            <p> <?php echo $printErr; ?></p>
            <p> <?php echo $printSucc; ?></p>

            <form method = "POST">
                <label class = "userstats"> Username: <input type = "text" /> </label>
                <label class = "userstats"> Password: <input type = "text" /> </label>
                <label class = "userstats"> Email: <input type = "text" /> </label>
                <label class = "userstats"> Current Status: <input type = "text" />
            </label>

                <input type = "submit" value = "Set info" />
            </form>
        </div>

        <!-- Message Inbox tab -->
        <div id = "messages" class = "tab-pane fade">
            <div id = "messageinbox">
                <h4> Your message inbox: </h4>
                <br />
                <table>
                    <tr>
                        <th> From </th>
                        <th> Subject </th>
                        <th> Topic </th>
                        <th> Date sent </th>
                        <th> Status </th>
                        <th> Message </th>
                    </tr>
                    <tr>
                        <td> from </td>
                        <td> subject </td>
                        <td> topic </td>

```

```

        <td> date sent </td>
        <td> status </td>
        <td> message </td>
    </tr>
</table>
</div>

<br/><br/>

<div id = "sentmessages">
    <h4> Messages you have sent: </h4>
    <br/>
    <table>
        <tr>
            <th> To </th>
            <th> Subject </th>
            <th> Topic </th>
            <th> Date sent </th>
            <th> Status </th>
            <th> Message </th>
        </tr>
        <tr>
            <td> to </td>
            <td> subject </td>
            <td> topic </td>
            <td> date sent </td>
            <td> status </td>
            <td> message </td>
        </tr>
    </table>
</div>
</div>

<!-- Private Message tab -->
<div id = "sendmessage" class = "tab-pane fade">
    <h3> Send Private Message </h3>

    <form method = "POST" id = "sendpm">
        <label> Enter username of forum member: <input type = "text"/> </label>
        <br/>
        <label> Enter subject: <input type = "text"/> </label>
        <br/>
        <label> Enter topic: <input type = "text"/> </label>
        <br/><br/>
        <label> Enter Message: </label>
        <br/>

```

```

        <textarea rows = "10" cols = "50"></textarea>
        <br/><br/>
        <input type = "submit" value = "Send PM"/>
    </form>
</div>

<!-- Recent Threads and Posts tab -->
<div id = "posts" class = "tab-pane fade">
    <div id = "listofposts">
        <h4> Number of posts you made: </h4>
        <br/>
        <table>
            <tr>
                <th> Thread title </th>
                <th> Subject </th>
                <th> Lasted posted </th>
            </tr>
            <tr>
                <td> title of thread last posted </td>
                <td> subject of the thread </td>
                <td> date of thread last posted </td>
            </tr>
        </table>
    </div>
</div>
</div>
</div>
</div>

<script src = "js/jquery-3.3.1.js"/></script>
<script src = "js/bootstrap.js"/></script>
<script src = "js/jquery-ui.min.js"/></script>
</body>
</html>

```

**4/9/19**

You now want to add the following to all pages that have the top black nav bar, especially the bolded, that will display the profile picture:

```
...
<ul class = "nav navbar-nav navbar-right">
    <li><a href = "userprofile.php"> Settings </a></li>
    <li><a href = "includes/logout.php"><span class = "glyphicon glyphicon-
user"></span> Logout </a></li>
    <li class = "dropdown">
        <a class = "dropdown-toggle" data-toggle = "dropdown" href = "#">
            Hello: <?php echo $_SESSION['username']; ?> <span class =
"caret"></span> </a>
        <ul class = "dropdown-menu">
            <li><a href = "#"> <?php echo "<img src = '" . $_SESSION['pic'] . '"
height='70' width='70'/>"; ?> </a></li>
            <li><a href = "#"> <?php echo $_SESSION['firstname'] . " " .
$_SESSION['lastname'] ?> </a></li>
            <li><a href = "#"> Email </a></li>
            <li><a href = "#"> Join date </a></li>
        </ul>
    </li>
</ul>
...
```



For the threads, because the ProfilePic folder cannot be seen in the threads php files because it's in a separate folder, you'll want to add the following to the threads instead:

```
...
<ul class = "nav navbar-nav navbar-right">
    <li><a href = "../userprofile.php"> Settings </a></li>
    <li><a href = "../includes/logout.php"><span class = "glyphicon glyphicon-
user"></span> Logout </a></li>
    <li class = "dropdown">
        <a class = "dropdown-toggle" data-toggle = "dropdown" href = "#">
            Hello: <?php echo $_SESSION['username']; ?> <span class =
"caret"></span> </a>
        <ul class = "dropdown-menu">
            <li><a href = "#"> <?php echo "<img src = '../' . $_SESSION['pic'] .
'" height='70' width='70' />"; ?> </a></li>
            <li><a href = "#"> <?php echo $_SESSION['firstname'] . " " .
$_SESSION['lastname'] ?> </a></li>
            <li><a href = "#"> Status </a></li>
            <li><a href = "#"> Join date </a></li>
        </ul>
    </li>
</ul>
...
```

Keep in mind that when the user uploads a new picture, it won't immediately show the picture, so the user must log off, then log back in again so the new picture will be displayed. Fix this problem in the future, but now you want to move on with the username, password, email and status bits.

Moving on, now you want to handle the bit where users can update their information such as their username, password, email and status. First create a new column in the accounts table called 'user\_status', and this will contain the status of users. Input the following SQL code:

```
alter table accounts
add user_status varchar(100)
```

You should now have the following table:

use r_id	user_fir stname	user_la stname	user_ema il	user_us ername	user_password	profile_ picture	user_ status
1	ftest	ltest	test@exa mple.com	testusern ame	\$2y\$10\$3zQA.S163L3H/5FTqtwkx. N.2WbD/ZY0uZeJHTj23Q5...	NULL	NULL

The number of characters should be less than or equal to 100. The status just tells how the user is doing. No need for an entire paragraph to tell others about the users current status.

To add new user info to the database, then display it in the profile, create a new file called “uploadstatus.php” and add it to the includes folder:

**uploadstatus.php**

```
<?php
include 'includes/database.php'; //needed for accessing the forumaccounts database and the
accounts table

//obtains the user inputs
$newUsername = "";
$newPassword = "";
$newEmail = "";
$setStatus = "";

$Sid = ""; //the intval obtains the inter value of the current user id in session

//contains the SQL code for adding new info to the accounts table
$newinfo = "";

//displays error and success messages
$message = "";
$successMessage = "";

if(isset($_POST['new-info']))
{
    //if the username field is not empty, but the others are, change only the username
    if(empty($_POST['changePassword']) && empty($_POST['changeEmail']) &&
empty($_POST['status']))
    {
        $newUsername = $_POST['changeUsername'];
        $Sid = intval($_SESSION['userid']);
        $newinfo = "update accounts set user_username='$newUsername' where user_id=$Sid;";

        if(mysqli_query($conn, $newinfo))
        {
            $successMessage = "Username has been changed!";
        }
        else
        {
            $message = "Error adding new username: " . $newinfo . "<br>";
            mysqli_error($conn);
        }
    }
    //if the password field is not empty, but the others are, change only the password
    else if(empty($_POST['changeUsername']) && empty($_POST['changeEmail']) &&
empty($_POST['status']))
    {
```

```

$newPassword = $_POST['changePassword'];
$hashedPwd = password_hash($newPassword, PASSWORD_DEFAULT);

$id = intval($_SESSION['userid']);
$newinfo = "update accounts set user_password='$hashedPwd' where user_id=$id;";

if(mysqli_query($conn, $newinfo))
{
    $succMessage = "Password has been changed!";
}
else
{
    $errMessage = "Error adding new password: " . $newinfo . "<br>";
    mysqli_error($conn);
}
}
//if the email field is not empty, but the others are, change only the email
else if(empty($_POST['changeUsername']) && empty($_POST['changePassword']) &&
empty($_POST['status']))
{
    $newEmail = $_POST['changeEmail'];

    $id = intval($_SESSION['userid']);
    $newinfo = "update accounts set user_email='$newEmail' where user_id=$id;";

    if(mysqli_query($conn, $newinfo))
    {
        $succMessage = "Email has been changed!";
    }
    else
    {
        $errMessage = "Error adding new email: " . $newinfo . "<br>";
        mysqli_error($conn);
    }
}
//if the status is not empty, but the others are, change only the status
else if(empty($_POST['changeUsername']) && empty($_POST['changePassword']) &&
empty($_POST['changeEmail']))
{
    $setStatus = $_POST['status'];

    $id = intval($_SESSION['userid']);
    $newinfo = "update accounts set user_status='$setStatus' where user_id=$id;";

    if(mysqli_query($conn, $newinfo))
    {
        $succMessage = "Status has been changed!";
    }
}

```

```

    }
    else
    {
        $errMessage = "Error adding new status: " . $newinfo . "<br>" . mysqli_error($conn);
    }
}
//if only the username field is blank, but the rest is not, change the password, email and
status
else if(empty($_POST['changeUsername']))
{
    $newPassword = $_POST['changePassword'];
    $hashedPwd = password_hash($newPassword, PASSWORD_DEFAULT);
    $newEmail = $_POST['changeEmail'];
    $setStatus = $_POST['status'];

    $id = intval($_SESSION['userid']);
    $newinfo = "update accounts set user_password='$hashedPwd', user_email='$newEmail',
user_status='$setStatus' where user_id=$id;";

    if(mysqli_query($conn, $newinfo))
    {
        $succMessage = "The password, email and status has been changed!";
    }
    else
    {
        $errMessage = "Error adding new password, email and status: " . $newinfo . "<br>" .
mysqli_error($conn);
    }
}
//if only the password field is blank, but the rest is not, change the username, email and
status
else if(empty($_POST['changePassword']))
{
    $newUsername = $_POST['changeUsername'];
    $newEmail = $_POST['changeEmail'];
    $setStatus = $_POST['status'];

    $id = intval($_SESSION['userid']);
    $newinfo = "update accounts set user_username='$newUsername',
user_email='$newEmail', user_status='$setStatus' where user_id=$id;";

    if(mysqli_query($conn, $newinfo))
    {
        $succMessage = "The username, email and status has been changed!";
    }
    else

```

```

        {
            $errMessage = "Error adding new username, email and status: " . $newinfo . "<br>" .
mysqli_error($conn);
        }
    }
    //if only the email field is blank, but the rest is not, change the username, password and
status
    else if(empty($_POST['changeEmail']))
    {
        $newUsername = $_POST['changeUsername'];
        $newPassword = $_POST['changePassword'];
        $hashedPwd = password_hash($newPassword, PASSWORD_DEFAULT);
        $setStatus = $_POST['status'];

        $id = intval($_SESSION['userid']);
        $newinfo = "update accounts set user_username='$newUsername',
user_password='$hashedPwd', user_status='$setStatus' where user_id=$id;";

        if(mysqli_query($conn, $newinfo))
        {
            $succMessage = "The username, password and status has been changed!";
        }
        else
        {
            $errMessage = "Error adding new username, password and status: " . $newinfo . "<br>"
. mysqli_error($conn);
        }
    }
    //if only the status field is blank, but the rest is not, change the username, password and
email
    else if(empty($_POST['status']))
    {
        $newUsername = $_POST['changeUsername'];
        $newPassword = $_POST['changePassword'];
        $hashedPwd = password_hash($newPassword, PASSWORD_DEFAULT);
        $newEmail = $_POST['changeEmail'];

        $id = intval($_SESSION['userid']);
        $newinfo = "update accounts set user_username='$newUsername',
user_password='$hashedPwd', user_email='$newEmail' where user_id=$id;";

        if(mysqli_query($conn, $newinfo))
        {
            $succMessage = "The username, password and email has been changed!";
        }
        else

```

```

    {
        $errMessage = "Error adding new username, password and email: " . $newinfo . "<br>"
        . mysqli_error($conn);
    }
}
}
?>

```

Now in the `userprofile.php`, add the following:

### **userprofile.php**

```

<?php
    session_start(); //needed for starting session

    include 'includes/uploadPics.php'; //needed for uploading pictures
    include 'includes/updatestatus.php'; //needed for updating information
?>

<html>
    <head>
        <title> Welcome to your profile </title>
        <meta charset = "utf-8"/>

        <link rel = "stylesheet" href = "css/bootstrap.css"/>
        <link rel = "stylesheet" href = "css/bootstrap.min.css"/>
        <link rel = "stylesheet" href = "css/userprofile.css"/>
    </head>

    <body>
        <div id = "tab">
            <nav class = "navbar navbar-inverse">
                <div class = "container-fluid">
                    <div class = "navbar-header">
                        <a href = "#" class = "navbar-brand"> Discussion Forum </a>
                    </div>

                    <ul class = "nav navbar-nav">
                        <li><a href = "forumhome.php"> Home </a></li>
                        <li><a href = "#"> Boards </a></li>
                        <li><a href = "#"> Community </a></li>
                    </ul>

                    <ul class = "nav navbar-nav navbar-right">
                        <li><a href = "userprofile.php"> Settings </a></li>
                        <li><a href = "includes/logout.php"><span class = "glyphicon glyphicon-
user"></span> Logout </a></li>

```

```

        <li class = "dropdown">
            <a class = "dropdown-toggle" data-toggle = "dropdown" href = "#">
                Hello: <?php echo $_SESSION['username']; ?> <span class =
"caret"></span> </a>
                <ul class = "dropdown-menu">
                    <li><a href = "#"> <?php echo "<img src = '" . $_SESSION['pic'] .
'" height='70' width='70'/>"; ?> </a></li>
                    <li><a href = "#"> Full Name: <?php echo
$_SESSION['firstname'] . " " . $_SESSION['lastname'] ?> </a></li>
                    <li><a href = "#"> Email: <?php echo $_SESSION['useremail']; ?>
</a></li>
                    <li><a href = "#"> Current Status: <?php echo
$_SESSION['status']; ?> </a></li>
                    <li><a href = "#"> Join date </a></li>
                </ul>
            </li>
        </ul>
    </div>
</nav>
</div>

<div id = "profilewindow">
    <ul class = "nav nav-tabs">
        <li><a data-toggle = "tab" href = "#profile"> Profile Page </a></li>
        <li><a data-toggle = "tab" href = "#messages"> Private Messages </a></li>
        <li><a data-toggle = "tab" href = "#sendmessage"> Send PM </a></li>
        <li><a data-toggle = "tab" href = "#posts"> Recent Threads and Posts </a></li>
    </ul>

    <div class = "tab-content">
        <!-- Profile Page tab -->
        <div id = "profile" class = "tab-pane fade in active">
            <form action = '<?php echo htmlspecialchars($_SERVER["PHP_SELF"]);?>'
method = 'POST' enctype = 'multipart/form-data'>
                <h3> Select image to upload: </h3>

                <input type = 'file' name = 'getFile'/>
                <input type = 'submit' value = 'Upload Photo' name = 'submit-pic'/>
            </form>

            <p> <?php echo $printErr; ?></p>
            <p> <?php echo $printSucc; ?></p>

            <form action = '<?php echo
htmlspecialchars($_SERVER["PHP_SELF"]);?>' method = "POST">

```

```

        <label class = "userstats"> Username: <input type = "text" name =
"changeUsername"/> </label>
        <label class = "userstats"> Password: <input type = "text" name =
"changePassword"/> </label>
        <label class = "userstats"> Email: <input type = "text" name =
"changeEmail"/> </label>
        <label class = "userstats"> Current Status: <input type = "text" name
= "status"/> </label>
        <input type = "submit" name = "new-info" value = "Set info"/>
    </form>

    <p> <?php echo $errMessage; ?></p>
    <p> <?php echo $succMessage; ?></p>
</div>
...

```

Two things to:

- After the user has added their new information, it will not be displayed right away, so the user has to re-login (with their new username and password of course if those were changed) to see the new information changed.
- There is more you have to add to the updatestatus.php (such as what if the username and status are not blank and so on)

You can do all of these in the future, but you have to move on to do other stuff. Now you want to add the join date, when the user first signed up on the site. So enter the following SQL code in the forumaccount database:

```

alter table accounts
add reg_date TIMESTAMP

```

Now you should see the new accounts table:

use r_id	user_fir stname	user_la stname	user_em ail	user_us ername	user_password	profile _pictur e	user_ statu s	reg_ date
1	ftest	ltest	test@exa mple.com	testuser name	\$2y\$10\$3zQA.S163L3H/5FTqtwk x.N.2WbD/ZY0uZeJHTj23Q5...	NULL	NULL	201 9- 04- 09 18



Now in the loginPage.php, add the following session variable:

#### loginPage.php

```
...
        /*obtain the id, username, email, firstname, lastname and password as well as
profile picture
        and status and registration date, and stores them in session variables*/
$_SESSION['userid'] = $row['user_id']; //id of user (intval gets the integer value)
$_SESSION['useremail'] = $row['user_email']; //email of user
$_SESSION['username'] = $row['user_username']; //username of user
$_SESSION['pwd'] = $row['user_password']; //password of user
$_SESSION['firstname'] = $row['user_firstname']; //firstname of user
$_SESSION['lastname'] = $row['user_lastname']; //lastname of user
$_SESSION['pic'] = $row['profile_picture']; //path of the uploaded profile
picture
$_SESSION['status'] = $row['user_status']; //status of user
$_SESSION['regdate'] = $row['reg_date']; //registration date of user
...
```

Now you want to add the following in all pages that have the black nav bar:

```
...
<div id = "tab">
    <nav class = "navbar navbar-inverse">
        <div class = "container-fluid">
            <div class = "navbar-header">
                <a href = "#" class = "navbar-brand"> Discussion Forum </a>
            </div>

            <ul class = "nav navbar-nav">
                <li><a href = "forumhome.php"> Home </a></li>
                <li><a href = "#"> Boards </a></li>
                <li><a href = "#"> Community </a></li>
            </ul>

            <ul class = "nav navbar-nav navbar-right">
                <li><a href = "userprofile.php"> Settings </a></li>
                <li><a href = "includes/logout.php"><span class = "glyphicon glyphicon-
user"></span> Logout </a></li>
                <li class = "dropdown">
                    <a class = "dropdown-toggle" data-toggle = "dropdown" href = "#">
                        Hello: <?php echo $_SESSION['username']; ?> <span class =
"caret"></span> </a>
                    <ul class = "dropdown-menu">
                        <li><a href = "#"> <?php echo "<img src = '' . $_SESSION['pic'] .
'' height='70' width='70' />"; ?> </a></li>
```

```

        <li><a href = "#"> Full Name: <?php echo
$_SESSION['firstname'] . " " . $_SESSION['lastname'] ?> </a></li>
        <li><a href = "#"> Email: <?php echo $_SESSION['useremail']; ?>
</a></li>

        <li><a href = "#"> Current Status: <?php echo
$_SESSION['status']; ?> </a></li>
        <li><a href = "#"> Join date: <?php echo $_SESSION['regdate'];
?> </a></li>
    </ul>
</li>
</ul>
</div>
</nav>
</div>
...

```

And for the threads files, add the following:

```

...
    <ul class = "nav navbar-nav navbar-right">
        <li><a href = "../userprofile.php"> Settings </a></li>
        <li><a href = "../includes/logout.php"><span class = "glyphicon glyphicon-
user"></span> Logout </a></li>
        <li class = "dropdown">
            <a class = "dropdown-toggle" data-toggle = "dropdown" href = "#">
                Hello: <?php echo $_SESSION['username']; ?> <span class =
"caret"></span> </a>
            <ul class = "dropdown-menu">
                <li><a href = "#"> <?php echo "<img src = '../" . $_SESSION['pic'] .
'" height='70' width='70' />"; ?> </a></li>
                <li><a href = "#"> Full Name: <?php echo $_SESSION['firstname'] .
" " . $_SESSION['lastname'] ?> </a></li>
                <li><a href = "#"> Email: <?php echo $_SESSION['useremail']; ?>
</a></li>
                <li><a href = "#"> Current Status: <?php echo $_SESSION['status'];
?> </a></li>
                <li><a href = "#"> Join date: <?php echo $_SESSION['regdate']; ?>
</a></li>
            </ul>
        </li>
    </ul>
...

```

---

**4/11/19**

**Note:** Keep these two things in mind in the future (see notes from 4/9/19):

- After the user has added their new information, it will not be displayed right away, so the user has to re-login (with their new username and password of course if those were changed) to see the new information changed.
- There is more you have to add to the updatestatus.php (such as what if the username and status are not blank and so on)

What you want now, for the loginPage.php and signupPage.php, is add the php code that handles the login and signup process in separate files. Create two files: login.php and signup.php, add them inside the includes folder, and add the respective code in each of the files:

**signup.php**

```
<?php
require 'includes/database.php';

$firstname = $lastname = $email = $username = $password = "";

//error message variable
$printErr = "";

//success message
$printSucc = "";

if($_SERVER["REQUEST_METHOD"] == "POST")
{
    if(isset($_POST["submitButton"]))
    {
        //if all fields are empty
        if(empty($_POST["fname"]) && empty($_POST["lname"]) &&
empty($_POST["email"]) && empty($_POST["uname"]) && empty($_POST["pwd"]))
        {
            $printErr = "Do not leave anything blank!";
        }
        //if only the first name field is filled in, but the others are empty
        else if(empty($_POST["lname"]) && empty($_POST["email"]) &&
empty($_POST["uname"]) && empty($_POST["pwd"]))
        {
            $printErr = "You only filled in the first name field!";
        }
    }
}
```

```

//if only the last name field is filled in, but the others are empty
else if(empty($_POST["fname"]) && empty($_POST["email"]) &&
empty($_POST["uname"]) && empty($_POST["pwd"]))
{
    $printErr = "You only filled in the last name field!";
}
//if only the email field is filled in, but the others are empty
else if(empty($_POST["fname"]) && empty($_POST["lname"]) &&
empty($_POST["uname"]) && empty($_POST["pwd"]))
{
    $printErr = "You only filled in the email field!";
}
//if only the username field is filled in, but the others are empty
else if(empty($_POST["fname"]) && empty($_POST["lname"]) &&
empty($_POST["email"]) && empty($_POST["pwd"]))
{
    $printErr = "You only filled in the username field!";
}
//if only the password is filled in, but the others are empty
else if(empty($_POST["fname"]) && empty($_POST["lname"]) &&
empty($_POST["email"]) && empty($_POST["uname"]))
{
    $printErr = "You only filled in the password field!";
}
else
{
    //the information the user inputs in the signupPage.php
    $firstname = $_POST['fname'];
    $lastname = $_POST['lname'];
    $email = $_POST['email'];
    $username = $_POST['uname'];
    $password = $_POST['pwd'];

    //adds user information into the accounts table of the forumaccounts database.
    //the values are left unspecified, the variables are not shown.
    $sql = "insert into accounts(user_firstname, user_lastname, user_email, user_username,
user_password)
        values(?, ?, ?, ?, ?)";

    //initializes database connection
    $stmt = mysqli_stmt_init($conn);

    //prepares the connection
    if(!mysqli_stmt_prepare($stmt, $sql))//if connection fails for whatever reason
    {
        $printErr = "SQL error!";
    }
}

```

```

    }
    else //if function successfully runs
    {
        //hashes the password (to prevent hackers from seeing password)
        $hashedPwd = password_hash($password, PASSWORD_DEFAULT);

        //since you have 5 different placeholders inside $sql, you will use 5 's'.
        mysqli_stmt_bind_param($stmt, 'sssss', $firstname, $lastname, $email, $username,
$hashedPwd);

        mysqli_stmt_execute($stmt); //executes the database connection

        //upon successful signup, output a message
        $printSucc = "You have successfully signed up! Sign in now to enter forum
discussion!";
    }
}
}
}
?>

```

### signupPage.php

```

<?php
include "includes/signup.php";
?>

<html>
    <head>
        <title> Register an account now! </title>
        <meta charset = "utf-8"/>

        <link rel = "stylesheet" href = "css/signup.css"/>
    </head>

    <body>
        <div id = "headings">
            <h1> Welcome </h1>
            <h4> Please sign up now to join our discussion forum </h4>
        </div>

        <div id = "signupwindow">
            <h3> Member Signup </h3>

            <form method = "POST" action = "<?php echo
htmlspecialchars($_SERVER["PHP_SELF"]);?>">
                <input type = "text" placeholder = "Enter First Name" name = "fname"/>

```

```

        <br/>
        <input type = "text" placeholder = "Enter Last Name" name = "lname"/>
        <br/>
        <input type = "text" placeholder = "Enter your Email" name = "email"/>
        <br/>
        <input type = "text" placeholder = "Enter Desired Username" name = "uname"/>
        <br/>
        <input type = "password" placeholder = "Enter Desired Password" name =
"pwd"/>
        <br/>
        <input type = "submit" value = "Register" name = "submitButton"/>
    </form>

    <a href = 'loginPage.php'> <h5> Already have an account, log in now </h5> </a>

    <p id = "errormessage"> <?php echo $printErr; ?> </p>
    <p id = "successmessage"> <?php echo $printSucc; ?></p>
</div>
</body>
</html>

```

### login.php

```

<?php
require 'includes/database.php'; //needed to connect to forumaccounts database

$loginusername = $loginpassword = "";

//error message variable
$printErr = "";

if($_SERVER["REQUEST_METHOD"] == "POST")
{
    if(isset($_POST["loginButton"]))
    {
        //if both fields are empty
        if(empty($_POST["loginuname"]) && empty($_POST["loginpwd"]))
        {
            $printErr = "Do not leave anything blank and then press the login button!";
        }
        //if the username field is empty
        else if(empty($_POST["loginuname"]))
        {
            $printErr = "You left the username field blank!";
        }
    }
}

```

```

//if the password field is empty
else if(empty($_POST["loginpwd"]))
{
    $printErr = "You left the password field blank!";
}
//else if everything is filled out
else
{
    //obtains the username and login input from the loginwindow form
    $loginusername = $_POST["loginuname"];
    $loginpassword = $_POST["loginpwd"];

    //obtains the username and password data from the accounts table of the
forumaccounts table
    $sql = "select * from accounts where user_username=? or user_password=? and
user_firstname or user_lastname;";

    $stmt = mysqli_stmt_init($conn); //initializes the connection to the forumaccounts
database

    if(!mysqli_stmt_prepare($stmt, $sql)) //if connection fails
    {
        header("Location: loginPage.php?error=sqlerror");
        exit();
    }
    else //if connection is fine
    {
        mysqli_stmt_bind_param($stmt, "ss", $loginusername, $loginusername);
        mysqli_stmt_execute($stmt); //executes the database connection

        $result = mysqli_stmt_get_result($stmt); //obtains the results from the database

        //checks if you did get any result from the database
        if($row = mysqli_fetch_assoc($result))
        {
            //takes the password the user used to log in and the password from the database
and see if they match
            $pwdCheck = password_verify($loginpassword, $row['user_password']);

            if($pwdCheck == false) //if the passwords don't match
            {
                $printErr = "Password is wrong!";
            }
            else if($pwdCheck == true) //else if the passwords do match
            {
                session_start(); // you need to start a session to start the login system
            }
        }
    }
}

```

```

        /*obtain the id, username, email, firstname, lastname and password as well as
profile picture
        and status and registration date, and stores them in session variables*/
$_SESSION['userid'] = $row['user_id']; //id of user (intval gets the integer value)
$_SESSION['useremail'] = $row['user_email']; //email of user
$_SESSION['username'] = $row['user_username']; //username of user
$_SESSION['pwd'] = $row['user_password']; //password of user
$_SESSION['firstname'] = $row['user_firstname']; //firstname of user
$_SESSION['lastname'] = $row['user_lastname']; //lastname of user
$_SESSION['pic'] = $row['profile_picture']; //path of the uploaded profile
picture

$_SESSION['status'] = $row['user_status']; //status of user
$_SESSION['regdate'] = $row['reg_date']; //registration date of user

//sends user to the forum homepage upon successful login
header("Location: forumhome.php?login=successful");
exit();
    }
}
else // if there is no user in the database that matches the inputted password and
username
{
    $printErr = "There is no such account!";
}
}
}
}
}
}
?>

```

### **loginPage.php**

```

<?php
include "includes/login.php";
?>

<html>
<head>
    <title> Login to account </title>
    <meta charset = "utf-8"/>

    <link rel = "stylesheet" href = "css/login.css"/>
    <link href="https://fonts.googleapis.com/css?family=Pacifico" rel="stylesheet">
</head>

<body>

```



```

<div id = "headings">
    <h1> Welcome to the Discussion forums </h1>
    <h4> Please login to enter </h4>
</div>

<div id = "loginwindow">
    <h4> Member Login </h4>

    <form method = "POST" action = "<?php echo
htmlspecialchars($_SERVER["PHP_SELF"]);?>">
        <input type = "text" placeholder = "Enter Username" name = "loginuname"/>
        <br/>
        <input type = "password" placeholder = "Enter Password" name = "loginpwd"/>
        <br/>
        <input type = "submit" value = "Login" name = "loginButton"/>
    </form>

    <p id = "errorMessage"> <?php echo $printErr; ?> </p>

    <a href = "#"><h4> Forgot password? Click here </h4></a>
    <a href = "signupPage.php"><h4> Don't have an account yet? Click here to sign up
for an account </hr></a>
</div>
</body>
</html>

```

Before going on any further, what you want to do now is add the topics of a particular subject in a folder that is named after that subject. For ex, for the subject ‘Culture’, create a folder called ‘Culture’, and store all topics associated with the subject (Philosophy, Religion, Art) in that folder.

You should have the following subject folders inside the Threads folder with the following topic files:

subject folders	topic files
Culture	Art.php Philosophy.php Religion.php
Social Science	Anthropology.php Archeology.php Sociology.php Psychology.php Linguistics.php
Natural Science and Mathematics	Biology.php Chemistry.php

	Physics.php EarthScience.php Astronomy.php Mathematics.php
World News	News.php
Technology	CompHard.php CompSoft.php VidGm.php Phones.php

Because you have these topic files in another folder (in the subject folder, which in turn is inside another folder, which is the threads folder), you have to tweak some of the code to get the correct file paths so that you can properly link the CSS, JavaScript and PHP scripts so that there are no errors that occur. So for all the topics files, make sure to add the following to all (they all have the same code with small differences, mainly in topic names):

```
<?php
    session_start();
?>

<html>
    <head>
        <title> (Topic) Threads </title>
        <meta charset = "utf-8"/>

        <link rel = "stylesheet" href = "../ThreadsCssFiles/bootstrap.css"/>
        <link rel = "stylesheet" href = "../ThreadsCssFiles/bootstrap.min.css"/>
        <link rel = "stylesheet" href = "../ThreadsCssFiles/threads.css"/>
    </head>

    <body id = "Topic">
        <!-- Navbar -->
        <nav class = "navbar navbar-inverse">
            <div class = "container-fluid">
                <div class = "navbar-header">
                    <a href = "#" class = "navbar-brand"> Discussion Forum </a>
                </div>

                <ul class = "nav navbar-nav">
                    <li><a href = "../forumhome.php"> Home </a></li>
                    <li><a href = "#"> Boards </a></li>
                    <li><a href = "#"> Community </a></li>
                </ul>

                <ul class = "nav navbar-nav navbar-right">
                    <li><a href = "../userprofile.php"> Settings </a></li>
```

```

        <li><a href = "../includes/logout.php"><span class = "glyphicon
glyphicon-user"></span> Logout </a></li>
        <li class = "dropdown">
            <a class = "dropdown-toggle" data-toggle = "dropdown" href = "#">
                Hello: <?php echo $_SESSION['username']; ?> <span class =
"caret"></span> </a>
                <ul class = "dropdown-menu">
                    <li><a href = "#"> <?php echo "<img src = '../..' . $_SESSION['pic']
. '" height='70' width='70' />"; ?> </a></li>
                    <li><a href = "#"> Full Name: <?php echo $_SESSION['firstname'] . " " .
$_SESSION['lastname'] ?> </a></li>
                    <li><a href = "#"> Email: <?php echo $_SESSION['useremail']; ?>
</a></li>
                    <li><a href = "#"> Current Status: <?php echo $_SESSION['status']; ?>
</a></li>
                    <li><a href = "#"> Join date: <?php echo $_SESSION['regdate']; ?>
</a></li>
                </ul>
            </li>
        </ul>
    </div>
</nav>

<div class = "container">
    <br/>
    <a href = "../add_topic_window.php"><button> Create new Thread
</button></a>
    <br/><br/>

    <table class = "table" id = "sub-forums">
        <thead>
            <tr>
                <th> Sub-Forums </th>
                <th> Threads/Posts </th>
                <th> Latest Posts</th>
            </tr>
        </thead>

        <tbody>
            <tr>
                <td> Sub-Forum 1 </td>
                <td> Threads: <br/> Posts: </td>
                <td> Last person who posted a thread in this sub-forum </td>
            </tr>
        </tbody>
    </table>

```

```

<br/><br/><br/>

<table class = "table threads-section" id = "threads-section">
  <thead>
    <tr>
      <th> Title/Threads </th>
      <th> Replies/Views </th>
      <th> Latest Posts </th>
    </tr>
  </thead>

  <tbody>
    <tr>
      <td> Thread title </td>
      <td> Replies: <br> Views: </td>
      <td> Last person who posted in this thread </td>
    </tr>
  </tbody>
</table>
</div>

<script src = "../ThreadsJSFiles/jquery-3.3.1.js"/></script>
<script src = "../ThreadsJSFiles/bootstrap.js"/></script>
</body>
</html>

```

Now in your forumhome.php, add the following (which basically sets up the file paths and links):

#### **forumhome.php**

```

...
<!-- Forum window -->
<div class = "container" id = "forumwindow">
  <table class = "table table1">
    <thead>
      <tr>
        <th> Culture </th>
        <th> Latest Posts </th>
      </tr>
    </thead>

    <tbody>
      <tr>
        <td> <a href = "Threads/Culture/Philosophy.php">Philosophy</a> </td>
        <td> thread title of topic <br> by <em>(username of user who created the
subject)</em> </td>

```

```

        </tr>
        <tr>
            <td> <a href = "Threads/Culture/Religion.php">Religion</a> </td>
            <td> thread title of topic </br> by <em>(username of user who created the
subject)</em> </td>
        </tr>
        <tr>
            <td> <a href = "Threads/Culture/Art.php">Art</a> </td>
            <td> thread title of topic </br> by <em>(username of user who created the
subject)</em> </td>
        </tr>
    </tbody>
</table>

```

```

<table class = "table table2">
    <thead>
        <tr>
            <th> Social Science </th>
            <th> Latest Posts </th>
        </tr>
    </thead>

    <tbody>
        <tr>
            <td> <a href = "Threads/Social
Science/Anthropology.php">Anthropology</a> </td>
            <td> thread title of topic </br> by <em>(username of user who created the
subject)</em> </td>
        </tr>
        <tr>
            <td> <a href = "Threads/Social
Science/Archeology.php">Archeology</a> </td>
            <td> thread title of topic </br> by <em>(username of user who created the
subject)</em> </td>
        </tr>
        <tr>
            <td> <a href = "Threads/Social
Science/Sociology.php">Sociology</a></td>
            <td> thread title of topic </br> by <em>(username of user who created the
subject)</em> </td>
        </tr>
        <tr>
            <td> <a href = "Threads/Social
Science/Psychology.php">Psychology</a></td>
            <td> thread title of topic </br> by <em>(username of user who created the
subject)</em> </td>
        </tr>
    </tbody>
</table>

```

```

        </tr>
        <tr>
            <td> <a href = "Threads/Social
Science/Linguistics.php">Linguistics</a></td>
            <td> thread title of topic </br> by <em>(username of user who created the
subject)</em> </td>
        </tr>
    </tbody>
</table>

```

```

<table class = "table table3">
    <thead>
        <tr>
            <th> Natural Science and Mathematics </th>
            <th> Latest Posts </th>
        </tr>
    </thead>

    <tbody>
        <tr>
            <td> <a href = "Threads/Natural Science and
Mathematics/Biology.php">Biology</a> </td>
            <td> thread title of topic </br> by <em>(username of user who created the
subject)</em> </td>
        </tr>
        <tr>
            <td> <a href = "Threads/Natural Science and
Mathematics/Chemistry.php">Chemistry</a> </td>
            <td> thread title of topic </br> by <em>(username of user who created the
subject)</em> </td>
        </tr>
        <tr>
            <td> <a href = "Threads/Natural Science and
Mathematics/Physics.php">Physics</a> </td>
            <td> thread title of topic </br> by <em>(username of user who created the
subject)</em> </td>
        </tr>
        <tr>
            <td> <a href = "Threads/Natural Science and
Mathematics/EarthScience.php">Earth Science</a> </td>
            <td> thread title of topic </br> by <em>(username of user who created the
subject)</em> </td>
        </tr>
        <tr>
            <td> <a href = "Threads/Natural Science and
Mathematics/Astronomy.php">Astronomy</a> </td>

```



```

                <td> <a href = "Threads/Technology/CompHard.php">Computer
Hardware</a> </td>
                <td> thread title of topic </br> by <em>(username of user who created the
subject)</em> </td>
            </tr>
            <tr>
                <td> <a href = "Threads/Technology/VidGm.php">Video Games</a>
</td>
                <td> thread title of topic </br> by <em>(username of user who created the
subject)</em> </td>
            </tr>
            <tr>
                <td> <a href = "Threads/Technology/Phones.php">Smartphones and
Tablets</a> </td>
                <td> thread title of topic </br> by <em>(username of user who created the
subject)</em> </td>
            </tr>
        </tbody>
    </table>
</div>
...

```

What you now want to do is create the threads window, where a user will create a question thread and other users can engage in discussion on the topic. You had a file that does that and it was called “add\_topic\_window.php”. Copy some of the code, but delete the entire file.

You want to create one for each subject (they will have almost the same code, but for that particular subject) So here, you want to create the files that will allow users to create a new topic for each particular subject. You should have the following files for each subject folder:

subject folders	files
Culture	Add_Culture_thread.php
Social Science	Add_SocialScience_thread.php
Natural Science and Mathematics	Add_ScienceMath_thread.php
World News	Add_News_thread.php
Technology	Add_Tech_thread.php



Now, with the code you copied from the add\_topic\_window.php, change some of the code, and add it to each of the files:

### **Add (Subject) thread.php**

```
<?php
    session_start();
?>

<html>
    <head>
        <title> New Topic --- (Subject) </title>
        <meta charset = "utf-8"/>

        <link rel = "stylesheet" href = "../css/bootstrap.css"/>
        <link rel = "stylesheet" href = "../css/bootstrap.min.css"/>
        <link rel = "stylesheet" href = "../css/addWindow.css">
    </head>

    <body>
        <nav class = "navbar navbar-inverse">
            <div class = "container-fluid">
                <div class = "navbar-header">
                    <a href = "#" class = "navbar-brand"> Discussion Forum </a>
                </div>

                <ul class = "nav navbar-nav">
                    <li><a href = "../forumhome.php"> Home </a></li>
                    <li><a href = "#"> Boards </a></li>
                    <li><a href = "#"> Community </a></li>
                </ul>

                <ul class = "nav navbar-nav navbar-right">
                    <li><a href = "../userprofile.php"> Settings </a></li>
                    <li><a href = "../includes/logout.php"><span class = "glyphicon glyphicon-
user"></span> Logout </a></li>
                    <li class = "dropdown">
                        <a class = "dropdown-toggle" data-toggle = "dropdown" href = "#">
                            Hello: <?php echo $_SESSION['username']; ?> <span class =
"caret"></span> </a>
                        <ul class = "dropdown-menu">
                            <li><a href = "#"> <?php echo "<img src = '../" . $_SESSION['pic'] . '"
height='70' width='70'/>"; ?> </a></li>
                            <li><a href = "#"> Full Name: <?php echo $_SESSION['firstname'] . " " .
$_SESSION['lastname'] ?> </a></li>
                            <li><a href = "#"> Email: <?php echo $_SESSION['useremail']; ?>
                        </a></li>
```

```

        <li><a href = "#"> Current Status: <?php echo $_SESSION['status']; ?>
</a></li>
        <li><a href = "#"> Join date: <?php echo $_SESSION['regdate']; ?>
</a></li>
    </ul>
</li>
</ul>
</div>
</nav>

<div id = "addnewtopic">
    <h3> Add New Thread </h3>

    <form method = "GET">
        <label> Author: <?php echo $_SESSION['username']; ?> </label>

        <br/><br/>

        <label>
            Select Topic:
            <select name = "topics-culture">
                <option name = "(subject)- (topic 1)"> (topic 1) </option>
                <option name = "(subject)- (topic 2) "> (topic 2) </option>
                <option name = "(subject)- (topic 3)"> (topic 3)</option>
                ...
            </select>
        </label>

        <br/><br/>

        <label> Title: <input type = "text" name = "title" size = "100"> </label>

        <br/><br/>

        <label> Content: <br/><textarea name = "content" rows = "20" cols =
"70"></textarea> </label>

        <br/><br/>

        <input type = "button" value = "Submit Thread">
    </form>
</div>

<script src = "../..js/jquery-3.3.1.js"/></script>
<script src = "../..js/bootstrap.js"/></script>
<script src = "../..js/jquery-ui.min.js"/></script>

```

```
</body>
</html>
```

Don't forget to add the following in each topic file:

```
...
<a href = "Add_(Subject)_thread.php"><button> Create new (Subject) Thread </button></a>
...
```

---

### *Creating the thread discussion process*

---

#### **4/13/19**

Now that the above is complete, what you now want to do is create the threads window, the actual thread window that contains the question and where user will engage in a discussion regarding the thread at hand.

You want to first create the layout for the thread window. Create a file called "DiscussionWindow.php", save it in the Discussion Forum folder, and add the following:

#### **DiscussionWindow.php**

```
<?php
    session_start();
?>
<html>
    <head>
        <title> Discussion Window </title>
        <meta charset = "utf-8"/>

        <link rel = "stylesheet" href = "css/bootstrap.css"/>
        <link rel = "stylesheet" href = "css/bootstrap.min.css"/>

        <style>
            /*The box-sizing property defines how the width and height of an element are
            calculated:
            should they include padding and borders, or not.*/
            *
            {
                box-sizing: border-box;
            }
            /*sets the text style for all the text in the div thread*/
```

```

#thread
{
    font-family: Arial, Helvetica, sans-serif;
    padding: 60px;
}
/*styles the header*/
header
{
    background-color: #666;
    padding: 10px;
    font-size: 15px;
    color: white;
}
/*creates two columns/boxes that floats next to each other*/
#userinfo
{
    float: left;
    padding: 20px;
    width: 30%;
    height: auto;
    background-color: #ccc;
}
article
{
    float: left;
    padding: 20px;
    width: 70%;
    height: auto;
    background-color: #f1f1f1;
}
/*clear floats after the columns*/
section:after
{
    content: "";
    display: table;
    clear: both;
}
/*style the footer*/
footer
{
    background-color: #777;
    padding: 5px;
    color: white;
}
/* Responsive layout - makes the two columns/boxes stack on top of
each other instead of next to each other, on small screens */

```

```

        @media (max-width: 600px)
        {
            #userinfo, article
            {
                width: 100%;
                height: auto;
            }
        }
    </style>
</head>

<body>
    <!-- Navbar -->
    <nav class = "navbar navbar-inverse">
        <div class = "container-fluid">
            <div class = "navbar-header">
                <a href = "#" class = "navbar-brand"> Discussion Forum </a>
            </div>

            <ul class = "nav navbar-nav">
                <li><a href = "forumhome.php"> Home </a></li>
                <li><a href = "#"> Boards </a></li>
                <li><a href = "#"> Community </a></li>
            </ul>

            <ul class = "nav navbar-nav navbar-right">
                <li> <a href = "userprofile.php"> Settings </a> </li>
                <li> <a href = "includes/logout.php"> <span class = "glyphicon glyphicon-
user"></span> Logout </a> </li>
                <li class = "dropdown">
                    <a class = "dropdown-toggle" data-toggle = "dropdown" href = "#"> Hello:
                    <?php echo $_SESSION['username']; ?> <span class = "caret"></span> </a>
                    <ul class = "dropdown-menu">
                        <li> <a href = "$"> <?php echo "<img src = '' . $_SESSION['pic'] . ''
height='70' width='70'/>"; ?> </a> </li>
                        <li> <a href = "$"> Full Name: <?php echo $_SESSION['firstname'] . " " .
$_SESSION['lastname']; ?> </a> </li>
                        <li> <a href = "$"> Email: <?php echo $_SESSION['useremail']; ?> </a>
                    </ul>
                </li>
                <li> <a href = "$"> Current Status: <?php echo $_SESSION['status']; ?>
                </a> </li>
                <li> <a href = "$"> Join Date: <?php echo $_SESSION['regdate']; ?> </a>
            </ul>
        </div>
    </nav>
</body>
</html>

```

```

        </div>
</nav>

<!-- Discussion thread -->
<div id = "thread">
    <header>
        <h3> Title: </h3>
        <h5> Date created: </h5>
        <h5> Subject: </h5>
        <h5> Topic: </h5>
    </header>

    <section>
        <div id = "userInfo">
            <h4> User Information </h4>

            <p> Profile Picture </p>
            <p> Full Name </p>
            <p> Location </p>
            <p> Join Date </p>
            <p> Current Status </p>
        </div>

        <article>
            <h4> Content </h4>
        </article>
    </section>

    <footer> Famous Quotes </footer>
</div>

<script src = "js/jquery-3.3.1.js"/></script>
<script src = "js/bootstrap.js"/></script>
<script src = "js/jquery-ui.min.js"/></script>
</body>
</html>

```

Keep in mind that you might be end up deleting this in the future, but this is the layout of what the thread discussion will look like for each subject and topic discussion.

**4/14/19**

Before moving on, move the addWindow.css file to the ThreadCssFiles folder and add the following CSS link to each Add\_(subject)\_thread:

```
<link rel = "stylesheet" href = "../ThreadCssFiles/addWindow.css"/>
```

The, create a CSS file called “DiscussionWindow.css”, save it to the ThreadCssFiles folder and add the following code:

**DiscussionWindow.css**

/\*The box-sizing property defines how the width and height of an element are calculated:

should they include padding and borders, or not.\*/

\*

```
{
  box-sizing: border-box;
}
```

/\*sets the text style for all the text in the div thread\*/

#thread

```
{
  font-family: Arial, Helvetica, sans-serif;
  padding: 60px;
}
```

/\*styles the header\*/

header

```
{
  background-color: #666;
  padding: 10px;
  font-size: 15px;
  color: white;
}
```

/\*creates two columns/boxes that floats next to each other\*/

#userinfo

```
{
  float: left;
  padding: 20px;
  width: 30%;
  height: auto;
  background-color: #ccc;
}
```

article

```
{
  float: left;
  padding: 20px;
  width: 70%;
  height: auto;
  background-color: #f1f1f1;
}
```

```
/*clear floats after the columns*/
section:after
{
    content: "";
    display: table;
    clear: both;
}
/*style the footer*/
footer
{
    background-color: #777;
    padding: 5px;
    color: white;
}
/* Responsive layout - makes the two columns/boxes stack on top of
each other instead of next to each other, on small screens */
@media (max-width: 600px)
{
    #userinfo, article
    {
        width: 100%;
        height: auto;
    }
}
```

What you now want to do is copy the code from the DiscussionWindow.php and create discussion windows for each subject. You should have the following files for the subject folders:

subject folders	files
Culture	DiscussionWindow_Culture.php
Social Science	DiscussionWindow_SocialScience.php
Natural Science and Mathematics	DiscussionWindow_ScienceMath.php
World News	DiscussionWindow_WorldNews.php
Technology	DiscussionWindow_Tech.php

You should have the following for each file (they all have similar code, which is really the same exact code from the DiscussionWindow.php file, but with some tweaks):

```
<?php
    session_start();
?>
<html>
    <head>
        <title> Discussion Window </title>
        <meta charset = "utf-8"/>

        <link rel = "stylesheet" href = "../ThreadsCssFiles/bootstrap.css"/>
```



```

<link rel = "stylesheet" href = "../ThreadsCssFiles/bootstrap.min.css"/>
<link rel = "stylesheet" href = "../ThreadsCssFiles/DiscussionWindow.css">
</head>

<body id = "(subjectname)">
  <!-- Navbar -->
  <nav class = "navbar navbar-inverse">
    <div class = "container-fluid">
      <div class = "navbar-header">
        <a href = "#" class = "navbar-brand"> Discussion Forum </a>
      </div>

      <ul class = "nav navbar-nav">
        <li><a href = "../forumhome.php"> Home </a></li>
        <li><a href = "#"> Boards </a></li>
        <li><a href = "#"> Community </a></li>
      </ul>

      <ul class = "nav navbar-nav navbar-right">
        <li> <a href = "../userprofile.php"> Settings </a> </li>
        <li> <a href = "../includes/logout.php"> <span class = "glyphicon glyphicon-
user"></span> Logout </a> </li>
        <li class = "dropdown">
          <a class = "dropdown-toggle" data-toggle = "dropdown" href = "#"> Hello:
            <?php echo $_SESSION['username']; ?> <span class = "caret"></span> </a>
          <ul class = "dropdown-menu">
            <li> <a href = "$"> <?php echo "<img src = '../" . $_SESSION['pic'] . ""
height='70' width='70'/>"; ?> </a> </li>
            <li> <a href = "$"> Full Name: <?php echo $_SESSION['firstname'] . " " .
$_SESSION['lastname']; ?> </a> </li>
            <li> <a href = "$"> Email: <?php echo $_SESSION['useremail']; ?> </a>
</li>
            <li> <a href = "$"> Current Status: <?php echo $_SESSION['status']; ?>
</a> </li>
            <li> <a href = "$"> Join Date: <?php echo $_SESSION['regdate']; ?> </a>
</li>
          </ul>
        </li>
      </ul>
    </div>
  </nav>

  <!-- Discussion thread -->
  <div id = "thread">
    <header>
      <h3> Title: </h3>

```

```

        <h5> Date created: </h5>
        <h5> Subject: </h5>
        <h5> Topic: </h5>
    </header>

    <section>
        <div id = "userInfo">
            <h4> User Information </h4>

            <p> Profile Picture </p>
            <p> Full Name </p>
            <p> Location </p>
            <p> Join Date </p>
            <p> Current Status </p>
        </div>

        <article>
            <h4> Content </h4>
        </article>
    </section>

    <footer> Famous Quotes </footer>
</div>

<script src = "../ThreadsJSFiles/jquery-3.3.1.js"/></script>
<script src = "../ThreadsJSFiles/bootstrap.js"/></script>
</body>
</html>

```

Now add the following in the DiscussionWindow.css:

```

...
/*Sets the backgrounds of the Discussion Windows*/
#Culture
{
    background-image:
url("https://png.pngtree.com/thumb_back/fw800/back_pic/02/67/99/70578f60912f607.jpg");
    background-repeat: no-repeat;
    background-size: cover;
}
#ScienceMath
{
    background-image: url("https://encrypted-
tbn0.gstatic.com/images?q=tbn:ANd9GcS0Y_2asRVMCmOaZTbWj5eS8YUTUTR1j_zV6_
YuT3UrLHFN76pbqw");
    background-repeat: no-repeat;
}

```

```

    background-size: cover;
}
#SocialScience
{
    background-image: url("https://encrypted-
tbn0.gstatic.com/images?q=tbn:ANd9GcRONyx4xEIg7ovmxzLfOrDRRmJ5Jj4fbBG5PkB-
BrFUclmz9AiT");
    background-repeat: no-repeat;
    background-size: cover;
}
#Tech
{
    background-image: url("https://encrypted-
tbn0.gstatic.com/images?q=tbn:ANd9GcTGQnKhHxMAWv4YG7i1DWjLL_xjhr4ZnYRPt4o
KbP88K-BnQIBP");
    background-repeat: no-repeat;
    background-size: cover;
}
#WorldNews
{
    background-image: url("https://ak7.picdn.net/shutterstock/videos/8558317/thumb/1.jpg");
    background-repeat: no-repeat;
    background-size: cover;
}
...

```

#### **4/16/19**

Before moving on, what you now want to do is, re-decorate the entire page, including revamping the navbar using custom CSS and not Bootstrap. This way, you can implement your own code and not use some other framework or jQuery libraries.

So the current navbar code, which is:

```

...
<!-- Navbar -->
<nav class = "navbar navbar-inverse">
    <div class = "container-fluid">
        <div class = "navbar-header">
            <a href = "#" class = "navbar-brand"> Discussion Forum </a>
        </div>

        <ul class = "nav navbar-nav">
            <li><a href = "../forumhome.php"> Home </a></li>
            <li><a href = "#"> Boards </a></li>

```

```

        <li><a href = "#"> Community </a></li>
    </ul>

    <ul class = "nav navbar-nav navbar-right">
        <li> <a href = "../userprofile.php"> Settings </a> </li>
        <li> <a href = "../includes/logout.php"> <span class = "glyphicon glyphicon-
user"></span> Logout </a> </li>
        <li class = "dropdown">
            <a class = "dropdown-toggle" data-toggle = "dropdown" href = "#"> Hello:
                <?php echo $_SESSION['username']; ?> <span class = "caret"></span> </a>
            <ul class = "dropdown-menu">
                <li> <a href = "$"> <?php echo "<img src = '../..' . $_SESSION['pic'] . '"
height='70' width='70'/>"; ?> </a> </li>
                <li> <a href = "$"> Full Name: <?php echo $_SESSION['firstname'] . " " .
$_SESSION['lastname']; ?> </a> </li>
                <li> <a href = "$"> Email: <?php echo $_SESSION['useremail']; ?> </a>
</li>
                <li> <a href = "$"> Current Status: <?php echo $_SESSION['status']; ?>
</a> </li>
                <li> <a href = "$"> Join Date: <?php echo $_SESSION['regdate']; ?> </a>
</li>
            </ul>
        </li>
    </ul>
</div>
</nav>
...

```

Delete it completely, and add the following code:

```

...
<!-- navbar -->
<ul class = "navbar">
    <li class = "left-tabs"><a href = "forumhome.php"> Home </a></li>
    <li class = "left-tabs"><a href = "#"> Board </a></li>
    <li class = "left-tabs"><a href = "#"> Community </a></li>

    <li class = "right-tabs"><a href = "includes/logout.php"> Logout </a></li>
    <li class = "dropdown right-tabs">
        <a href = "#" class = "dropdownButton"> Hello: <?php echo
$_SESSION['username']; ?> </a>
        <div class = "dropdown-content">
            <a href = "#"> <?php echo "<img src = '" . $_SESSION['pic'] . '" height='70'
width='70'/>"; ?> </a>
            <a href = "#"> Full Name: <?php echo $_SESSION['firstname'] . " " .
$_SESSION['lastname']; ?> </a>
            <a href = "#"> Email: <?php echo $_SESSION['useremail']; ?> </a>

```

```

        <a href = "#"> Current Status: <?php echo $_SESSION['status']; ?> </a>
        <a href = "#"> Join Date: <?php echo $_SESSION['regdate']; ?> </a>
        <a href = "userprofile.php"> Go to account settings </a>
    </div>
</li>
</ul>

```

...

Add this code to the following files: forumhome.php, userprofile.php

For the Threads files, add this code instead:

```

...
<!-- navbar -->
<ul class = "navbar">
    <li class = "left-tabs"><a href = "../forumhome.php"> Home </a></li>
    <li class = "left-tabs"><a href = "#"> Board </a></li>
    <li class = "left-tabs"><a href = "#"> Community </a></li>

    <li class = "right-tabs"><a href = "../includes/logout.php"> Logout </a></li>
    <li class = "dropdown right-tabs">
        <a href = "#" class = "dropdownButton"> Hello: <?php echo
$_SESSION['username']; ?> </a>
        <div class = "dropdown-content">
            <a href = "#"> <?php echo "<img src = '../" . $_SESSION['pic'] . "' height='70'
width='70'/>"; ?> </a>
            <a href = "#"> Full Name: <?php echo $_SESSION['firstname'] . " " .
$_SESSION['lastname']; ?> </a>
            <a href = "#"> Email: <?php echo $_SESSION['useremail']; ?> </a>
            <a href = "#"> Current Status: <?php echo $_SESSION['status']; ?> </a>
            <a href = "#"> Join Date: <?php echo $_SESSION['regdate']; ?> </a>
            <a href = "../userprofile.php"> Go to account settings </a>
        </div>
    </li>
</ul>

```

...

In order for the codes above to work, create a css file called “Navbar.css”, add it to both the css and ThreadCssFiles folders and add the code:

### **Navbar.css**

```

.navbar
{
    list-style-type: none;
    margin: 0;
    padding: 0;

```

```
overflow: hidden;
background-color: #333;
border-radius: 10px;
}
.navbar .left-tabs
{
float: left;
}
.navbar .right-tabs
{
float: right;
}
.navbar li a, .dropdownButton
{
display: inline-block;
color: white;
text-decoration: none;
text-align: center;
padding: 14px 16px;
}
.navbar li a:hover, .dropdown:hover .dropdownButton
{
background-color: blue;
}
.navbar li.dropdown
{
display: inline-block;
}
.navbar .dropdown .dropdown-content
{
display: none;
position: absolute;
background-color: #f9f9f9;
min-width: 160px;
box-shadow: 0px 8px 16px 0px rgba(0,0,0,0.2);
z-index: 1;
}
.navbar .dropdown:hover .dropdown-content
{
display: block;
}
.navbar .dropdown .dropdown-content a
{
color: black;
padding: 12px 16px;
text-decoration: none;
```

```

display: block;
text-align: left;
}
.navbar .dropdown .dropdown-content a:hover
{
background-color: #f1f1f1;
}

```

What you now want to do is create a file in Discussion Forum called “Unused files”, and this is where all your Bootstrap and jQuery files will go, since you will be doing all the code without using any CSS framework or JavaScript library. Also delete any Bootstrap and jQuery links from all php files since you won’t be using them.

Doing so, the layout will be completely messed up, and the tables will be completely disorganized.

Starting with the forumhome.php and main.css, add the following:

#### **forumhome.php**

```

...
<!-- Forum window -->
    <div id = "forumwindow">
        <table class = "table">
            <thead>
                <tr>
                    <th class = "leftside"> Culture </th>
                    <th class = "rightside"> Latest Posts </th>
                </tr>
            </thead>

            <tbody>
                <tr>
                    <td class = "leftside"> <a href =
"Threads/Culture/Philosophy.php">Philosophy</a> </td>
                    <td class = "rightside"> thread title of topic </br> by <em>(username of user
who created the subject)</em> </td>
                </tr>
                <tr>
                    <td class = "leftside"> <a href = "Threads/Culture/Religion.php">Religion</a>
</td>
                    <td class = "rightside"> thread title of topic </br> by <em>(username of user
who created the subject)</em> </td>
                </tr>
                <tr>
                    <td class = "leftside"> <a href = "Threads/Culture/Art.php">Art</a> </td>
                    <td class = "rightside"> thread title of topic </br> by <em>(username of user
who created the subject)</em> </td>
                </tr>

```

```
</tbody>
</table>
```

```
<table class = "table">
```

```
<thead>
```

```
<tr>
```

```
<th class = "leftside"> Social Science </th>
```

```
<th class = "rightside"> Latest Posts </th>
```

```
</tr>
```

```
</thead>
```

```
<tbody>
```

```
<tr>
```

```
<td class = "leftside"> <a href = "Threads/Social
Science/Anthropology.php">Anthropology</a> </td>
```

```
<td class = "rightside"> thread title of topic </br> by <em>(username of user
who created the subject)</em> </td>
```

```
</tr>
```

```
<tr>
```

```
<td class = "leftside"> <a href = "Threads/Social
Science/Archeology.php">Archeology</a> </td>
```

```
<td class = "rightside"> thread title of topic </br> by <em>(username of user
who created the subject)</em> </td>
```

```
</tr>
```

```
<tr>
```

```
<td class = "leftside"> <a href = "Threads/Social
Science/Sociology.php">Sociology</a></td>
```

```
<td class = "rightside"> thread title of topic </br> by <em>(username of user
who created the subject)</em> </td>
```

```
</tr>
```

```
<tr>
```

```
<td class = "leftside"> <a href = "Threads/Social
Science/Psychology.php">Psychology</a></td>
```

```
<td class = "rightside"> thread title of topic </br> by <em>(username of user
who created the subject)</em> </td>
```

```
</tr>
```

```
<tr>
```

```
<td class = "leftside"> <a href = "Threads/Social
Science/Linguistics.php">Linguistics</a></td>
```

```
<td class = "rightside"> thread title of topic </br> by <em>(username of user
who created the subject)</em> </td>
```

```
</tr>
```

```
</tbody>
```

```
</table>
```

```
<table class = "table">
```



```

<thead>
  <tr>
    <th class = "leftside"> Natural Science and Mathematics </th>
    <th class = "rightside"> Latest Posts </th>
  </tr>
</thead>

<tbody>
  <tr>
    <td class = "leftside"> <a href = "Threads/Natural Science and
Mathematics/Biology.php">Biology</a> </td>
    <td class = "rightside"> thread title of topic </br> by <em>(username of user
who created the subject)</em> </td>
  </tr>
  <tr>
    <td class = "leftside"> <a href = "Threads/Natural Science and
Mathematics/Chemistry.php">Chemistry</a> </td>
    <td class = "rightside"> thread title of topic </br> by <em>(username of user
who created the subject)</em> </td>
  </tr>
  <tr>
    <td class = "leftside"> <a href = "Threads/Natural Science and
Mathematics/Physics.php">Physics</a> </td>
    <td class = "rightside"> thread title of topic </br> by <em>(username of user
who created the subject)</em> </td>
  </tr>
  <tr>
    <td class = "leftside"> <a href = "Threads/Natural Science and
Mathematics/EarthScience.php">Earth Science</a> </td>
    <td class = "rightside"> thread title of topic </br> by <em>(username of user
who created the subject)</em> </td>
  </tr>
  <tr>
    <td class = "leftside"> <a href = "Threads/Natural Science and
Mathematics/Astronomy.php">Astronomy</a> </td>
    <td class = "rightside"> thread title of topic </br> by <em>(username of user
who created the subject)</em> </td>
  </tr>
  <tr>
    <td class = "leftside"> <a href = "Threads/Natural Science and
Mathematics/Mathematics.php">Mathematics</a> </td>
    <td class = "rightside"> thread title of topic </br> by <em>(username of user
who created the subject)</em> </td>
  </tr>
</tbody>
</table>

```

```

<table class = "table">
  <thead>
    <tr>
      <th class = "leftside"> Things happening outside the Internet </th>
      <th class = "rightside"> Latest Posts </th>
    </tr>
  </thead>

  <tbody>
    <tr>
      <td class = "leftside"> <a href = "Threads/World
News/News.php">International News</a> </td>
      <td class = "rightside"> thread title of topic </br> by <em>(username of user
who created the subject)</em> </td>
    </tr>
  </tbody>
</table>

<table class = "table">
  <thead>
    <tr>
      <th class = "leftside"> What's going on now in Tech </th>
      <th class = "rightside"> Latest Posts </th>
    </tr>
  </thead>

  <tbody>
    <tr>
      <td class = "leftside"> <a href =
"Threads/Technology/CompSoft.php">Computer Software/Programs</a> </td>
      <td class = "rightside"> thread title of topic </br> by <em>(username of user
who created the subject)</em> </td>
    </tr>
    <tr>
      <td class = "leftside"> <a href =
"Threads/Technology/CompHard.php">Computer Hardware</a> </td>
      <td class = "rightside"> thread title of topic </br> by <em>(username of user
who created the subject)</em> </td>
    </tr>
    <tr>
      <td class = "leftside"> <a href = "Threads/Technology/VidGm.php">Video
Games</a> </td>
      <td class = "rightside"> thread title of topic </br> by <em>(username of user
who created the subject)</em> </td>
    </tr>
  </tbody>
</table>

```

```

        <tr>
        <td class = "leftside"> <a href =
"Threads/Technology/Phones.php">Smartphones and Tablets</a> </td>
        <td class = "rightside"> thread title of topic </br> by <em>(username of user
who created the subject)</em> </td>
        </tr>
    </tbody>
</table>
</div>

```

...

### main.css

```

body
{
    background-image:
url("https://png.pngtree.com/thumb_back/fh260/back_pic/04/00/23/4757fc9e77387db.jpg");
    background-repeat: no-repeat;
    background-size: cover;
}

#searchwindow
{
    padding-left: 25%;
    padding-top: 10%;
}

#searchwindow input[type=text]
{
    width: 60%;
    height: 30px;

    background-color: #F5F5DC;
    font-family: Times New Roman;
}

#searchwindow input[type=submit]
{
    width: 60%;
    height: 30px;

    background-color: #F5FFFA;
    font-family: Times New Roman;
}

#searchwindow p
{
    padding-right: 30%;
}

```

```

font-family: "Lobster", cursive;
font-size: 200%;
}

#forumwindow
{
padding-top: 70px;
padding-left: 5%;
}

#searchwindow p, #forumwindow .table th, #forumwindow .table td, #forumwindow .table td
a
{
color: #D3D3D3;
text-decoration: none;
}

#forumwindow .table th, #forumwindow .table td
{
font-family: Times New Roman;
border-radius: 5px;
padding: 7px;
}

#forumwindow .table .leftside
{
width: 8%;
}

#forumwindow .table .rightside
{
width: 20%;
}

#forumwindow .table thead
{
background-color: purple;
}

#forumwindow .table tbody
{
background-color: #2F4F4F;
}

```

Now for the `userprofile.php` and `userprofile.css`

### **userprofile.php**

```

...
<!-- Displays profile image and settings for changing picture -->

```

```

<div id = "profileimage-window">
    <?php echo "<img src = '' . $_SESSION['pic'] . '' height='150' width='150'/>"; ?>
    <br/>
    <form action = '<?php echo htmlspecialchars($_SERVER["PHP_SELF"]);?>' method
= 'POST' enctype = 'multipart/form-data'>
        <h3> Select new image to upload: </h3>

        <input type = 'file' name = 'getFile'/>
        <input type = 'submit' value = 'Upload Photo' name = 'submit-pic'/>
    </form>

    <p> <?php echo $printErr; ?></p>
    <p> <?php echo $printSucc; ?></p>
</div>

<!-- Settings for changing user info -->
<div id = "settings-window">
    <h3> Change Info </h3>
    <form action = '<?php echo htmlspecialchars($_SERVER["PHP_SELF"]);?>'
method = "POST">
        <br/><br/>
        <label class = "userstats"> Username: <input type = "text" name =
"changeUsername"/> </label>
        <br/><br/>
        <label class = "userstats"> Password: <input type = "text" name =
"changePassword"/> </label>
        <br/><br/>
        <label class = "userstats"> Email: <input type = "text" name = "changeEmail"/>
</label>
        <br/><br/>
        <label class = "userstats"> Current Status: <input type = "text" name = "status"/>
</label>
        <br/><br/>
        <input type = "submit" name = "new-info" value = "Set info"/>
    </form>

    <p> <?php echo $errMessage; ?></p>
    <p> <?php echo $succMessage; ?></p>
</div>

<!-- Displays messages from other users and messages sent -->
<div id = "message-window">
    <!-- Messages from other users -->
    <div id = "messageinbox">
        <h4> Your message inbox: </h4>
        <br/>

```

```

<table>
  <tr>
    <th> From </th>
    <th> Subject </th>
    <th> Topic </th>
    <th> Date sent </th>
    <th> Status </th>
    <th> Message </th>
  </tr>
  <tr>
    <td> from </td>
    <td> subject </td>
    <td> topic </td>
    <td> date sent </td>
    <td> status </td>
    <td> message </td>
  </tr>
</table>
</div>
<br/>
<!-- Messages you sent -->
<div id = "sentmessages">
  <h4> Messages you have sent: </h4>
  <br/>
  <table>
    <tr>
      <th> To </th>
      <th> Subject </th>
      <th> Topic </th>
      <th> Date sent </th>
      <th> Status </th>
      <th> Message </th>
    </tr>
    <tr>
      <td> to </td>
      <td> subject </td>
      <td> topic </td>
      <td> date sent </td>
      <td> status </td>
      <td> message </td>
    </tr>
  </table>
</div>
</div>

<!-- Option for sending private message -->

```

```

<div id = "privatemessage-window">
  <h3> Send Private Message </h3>

  <form method = "POST" id = "sendpm">
    <label> Enter username of forum member: <input type = "text"/> </label>
    <br/>
    <label> Enter subject: <input type = "text"/> </label>
    <br/>
    <label> Enter topic: <input type = "text"/> </label>
    <br/><br/>
    <label> Enter Message: </label>
    <br/>
    <textarea name = "pm-message"></textarea>
    <br/><br/>
    <input type = "submit" value = "Send PM"/>
  </form>
</div>

<!-- List of threads you last made -->
<div id = "user-posts">
  <table>
    <tr>
      <th> Last thread you posted: </th>
      <th> Subject of last thread: </th>
      <th> Date of thread posted: </th>
    </tr>
    <tr>
      <td> </td>
      <td> </td>
      <td> </td>
    </tr>
  </table>
</div>
</body>

```

...

#### userprofile.css

```

#profileimage-window, #settings-window, #message-window, #privatemessage-window,
#user-posts

```

```

{
  border: 1px solid black;
  margin-bottom: 20px;
}

```

```

/*styles the profileimage-window widow*/

```

```

#profileimage-window
{

```

```
margin-top: 10px;
padding: 30px;
}
#profileimage-window input[type=submit], #profileimage-window input[type=file]
{
    font-family: serif;
    background-color: lightgreen;
    border: none;
    border-radius: 3px;
}

/*styles the settings-window window*/
#settings-window
{
    padding: 30px;
}
#settings-window h3
{
    text-align: center;
}
#settings-window .userstats
{
    font-family: sans-serif;
}
#settings-window .userstats input[type=text]
{
    font-family: serif;
    width: 50%;
    height: 30px;
    border: 1px solid #ccc;
    border-radius: 4px;
}
#settings-window input[type=submit]
{
    font-family: serif;
    margin-left: 40%;
    background-color: lightgreen;
    border: none;
    border-radius: 3px;
}

/*styles the message-window window*/
#message-window
{
    padding: 0 10px 20px 10px;
}
```



```

#message-window #messageinbox table, #message-window #sentmessages table
{
    font-family: arial, sans-serif;
    border-collapse: collapse;
    width: 100%;
}
#message-window #messageinbox td, #message-window #messageinbox th, #message-
window #sentmessages td, #message-window #sentmessages th
{
    border: 1px solid #dddddd;
    text-align: left;
    padding: 8px;
}

/*styles the privatemessgae-window window*/
#privatemessage-window
{
    padding: 30px;
}
#privatemessage-window label
{
    font-family: sans-serif;
}
#privatemessage-window input[type=text]
{
    font-family: serif;
    width: 50%;
    height: 30px;
    border: 1px solid #ccc;
    border-radius: 4px;
}
#privatemessage-window textarea
{
    font-family: serif;
    width: 70%;
    height: 30%;
    border: 1px solid #ccc;
    border-radius: 4px;
}
#privatemessage-window input[type=submit]
{
    font-family: serif;
    margin-left: 40%;
    background-color: lightgreen;
    border: none;
    border-radius: 3px;
}

```

```

}

/*styles the user-posts window*/
#user-posts
{
    padding: 20px;
}
#user-posts table
{
    font-family: arial, sans-serif;
    border-collapse: collapse;
    width: 100%;
}
#user-posts th, #user-posts td
{
    border: 1px solid #dddddd;
    text-align: left;
    padding: 8px;
}

```

Now for the topic files in the subject folders, it is cumbersome and tedious to change all of them, but most topic files should look like the following:

```

...
    <br/>
    <a href = "Add_(Subject)_thread.php"><button> Create new (Subject name) Thread
</button></a>
    <br/><br/>

    <div id = "threadwindow">
        <table class = "table">
            <thead>
                <tr>
                    <th> Sub-Forums </th>
                    <th> Threads/Posts </th>
                    <th> Latest Posts</th>
                </tr>
            </thead>

            <tbody>
                <tr>
                    <td> Sub-Forum 1 </td>
                    <td> Threads: <br/> Posts: </td>
                    <td> Last person who posted a thread in this sub-forum </td>
                </tr>
            </tbody>
        </table>
    </div>

```

```

</table>

<br/><br/><br/>

<table class = "table">
  <thead>
    <tr>
      <th> Title/Threads </th>
      <th> Replies/Views </th>
      <th> Latest Posts </th>
    </tr>
  </thead>

  <tbody>
    <tr>
      <td> Thread title </td>
      <td> Replies: </br> Views: </td>
      <td> Last person who posted in this thread </td>
    </tr>
  </tbody>
</table>
</div>
...

```

Now the threads.css should look like this (it will decorate all the topic files)

### **threads.css**

```

body
{
  background-color: lightblue;
}

button
{
  width: 50%;
  height: 30px;
  border: none;
  border-radius: 4px;
  cursor: pointer;
  background-color: #FFDEAD;
  font-family: serif;
}

#threadwindow
{
  padding-top: 70px;
  padding-left: 5%;
}

```

```
}

#threadwindow .table thead
{
    background-color: purple;
}

#threadwindow .table tbody
{
    background-color: #2F4F4F;
}

#threadwindow .table th, #threadwindow .table td
{
    font-family: Times New Roman;
    border-radius: 5px;
    padding: 7px;
    color: #D3D3D3;
}

#Anthropology
{
    background-image: url("https://www.kevinmd.com/blog/wp-content/uploads/shutterstock_262498211.jpg");
    background-repeat: no-repeat;
    background-size: cover;
}

#Archeology
{
    background-image:
url("https://previews.123rf.com/images/violin/violin1711/violin171100171/89935702-old-egypt-scriptures-archeology-background.jpg");
    background-repeat: no-repeat;
    background-size: cover;
}

#Art
{
    background-image:
url("https://previews.123rf.com/images/grthirteen/grthirteen1407/grthirteen140700153/29721561-abstract-art-background-hand-painted-background-self-made-.jpg");
    background-repeat: no-repeat;
    background-size: cover;
}
```

#### #Astronomy

```
{
  background-image: url("https://image.freepik.com/free-vector/space-background-illustration_1284-4406.jpg");
  background-repeat: no-repeat;
  background-size: cover;
}
```

#### #Biology

```
{
  background-image:
url("https://us.123rf.com/450wm/zagory/zagory1606/zagory160600042/58467218-seamless-
background-with-hand-drawn-icons-on-the-theme-of-biology-light-outline-on-a-dark-
background.jpg?ver=6");
  background-repeat: no-repeat;
  background-size: cover;
}
```

#### #Chemistry

```
{
  background-image:
url("https://previews.123rf.com/images/xiver/xiver1104/xiver110400042/9302822-chemistry-
background.jpg");
  background-repeat: no-repeat;
  background-size: cover;
}
```

#### #EarthScience

```
{
  background-image:
url("https://png.pngtree.com/thumb_back/fw800/back_pic/00/06/31/695628f664c006c.jpg");
  background-repeat: no-repeat;
  background-size: cover;
}
```

#### #Linguistics

```
{
  background-image: url("http://www.littlelanguagesite.com/wp-
content/uploads/2015/03/linguistics-660x330.jpg");
  background-repeat: no-repeat;
  background-size: cover;
}
```

#### #Mathematics

```
{
```

```
background-image:
url("https://previews.123rf.com/images/ihor_seamless/ihor_seamless0809/ihor_seamless0809
00008/3524160-seamlessly-vector-wallpaper-mathematics-on-black.jpg");
background-repeat: no-repeat;
background-size: cover;
}
```

#### #Physics

```
{
background-image:
url("https://us.123rf.com/450wm/saintantonio21/saintantonio211603/saintantonio2116030007
5/53595262-natuurkunde-i-en-sciense-naadloze-patroon-met-schets-elementen-hand-
getekende-doodles-achtergrond-ve.jpg?ver=6");
background-repeat: no-repeat;
background-size: cover;
}
```

#### #Philosophy

```
{
background-image:
url("https://athensinsiders.com/content/uploads/2017/03/Greek_philosopher_busts.jpg");
background-repeat: no-repeat;
background-size: cover;
}
```

#### #Psychology

```
{
background-image:
url("https://d19d5sz0wkl0lu.cloudfront.net/dims4/default/bfa4f1f/2147483647/resize/800x%3
E/quality/90/?url=https%3A%2F%2Ffatd-
brightspot.s3.amazonaws.com%2Fyourbrainfw.png");
background-repeat: no-repeat;
background-size: cover;
}
```

#### #Religion

```
{
background-image: url("http://religious-symbols.net/Pictures/religious-symbols_2.png");
background-repeat: no-repeat;
background-size: cover;
}
```

#### #Sociology

```
{
```

```
background-image: url("https://media.istockphoto.com/vectors/network-design-vector-id477648318?k=6&m=477648318&s=612x612&w=0&h=Ia8UGLXD9LANK9FL6Uf32yCD BjpgFu9ZEFnWN4aa_T4Y=");
background-repeat: no-repeat;
background-size: cover;
}

#InterNews
{
background-image: url("https://i.ytimg.com/vi/jtpLpOYbE9M/maxresdefault.jpg");
background-repeat: no-repeat;
background-size: cover;
}

#CompSoft
{
background-image: url("https://thumbs.dreamstime.com/t/html-code-programming-software-background-d-render-abstract-technology-developer-computer-script-102733125.jpg");
background-repeat: no-repeat;
background-size: cover;
}

#CompHard
{
background-image:
url("https://i.pining.com/originals/8a/6c/2e/8a6c2e45cfb9e32859e9e7fa80350ad1.jpg");
background-repeat: no-repeat;
background-size: cover;
}

#VidGm
{
background-image:
url("https://i.pining.com/originals/9b/9c/ef/9b9cefa3d0b19285a39163230b8edb09.jpg");
background-repeat: no-repeat;
background-size: cover;
}

#Phones
{
background-image: url("https://cdn.vox-cdn.com/thumbor/nWjiPR-hSCy1_WXvFxr6pAjZYYy=/0x0:2040x1360/1200x800/filters:focal(857x517:1183x843)/cdn.vox-cdn.com/uploads/chorus_image/image/61520647/jbareham_180424_2499_0007.0.jpg");
background-repeat: no-repeat;
background-size: cover;
}
```

**4/21/19**

What you want to do now is, for the DiscussionWindow\_(Subject).php files, you want to delete the footer element that contains the “Famous Quotes”, because that is useless. You can if course add it later on in the future.

You also want to add a comment box to each of the Discussion Window files (else, how are users going to respond to a thread):

```
...
<!-- Comment box -->
<div id = "comment">
  <form method = "POST">
    <textarea rows = "10" cols = "40" name = "commentbox"></textarea>
    <input type = "submit" name = "submit-comment" value = "Submit Comment!">
  </form>
</div>...
```

Add the following to the DiscussionWindow.css

**DiscussionWindow.css**

/\*The box-sizing property defines how the width and height of an element are calculated: should they include padding and borders, or not.\*/

```
*
{
  box-sizing: border-box;
}
/*sets the text style for all the text in the div thread*/
```

```
#thread
{
  font-family: Arial, Helvetica, sans-serif;

  margin-top: 5%;
  margin-left: 5%;
  margin-right: 5%;
}
```

/\*styles the header\*/

```
#thread header
{
  background-color: #666;

  padding: 10px;
  font-size: 15px;
  color: white;
}
```

/\*creates two columns/boxes that floats next to each other\*/

```
#thread section #userinfo
{
```



```
float: left;
padding: 20px;
width: 30%;
height: auto;

background-color: #ccc;
}
#thread section article
{
float: left;
padding: 20px;
width: 70%;
height: auto;

background-color: #f1f1f1;
}
/*clear floats after the columns*/
#thread section:after
{
content: "";
display: table;
clear: both;
}

#comment
{
text-align: center;
background-color: grey;

border-style: inset;
border-radius: 5px;
}
#comment form
{
margin-top: 30px;
}
#comment form textarea
{
border-style: inset;
border-radius: 5px;

font-family: serif;
}
#comment form input[type=submit]
{
margin-top: 10px;
```

```

border: none;
border-radius: 5px;

font-family: serif;
font-size: 15px;

cursor: pointer;
}
/* Responsive layout - makes the two columns/boxes stack on top of
each other instead of next to each other, on small screens */
@media (max-width: 600px)
{
    #userinfo, article
    {
        width: 100%;
        height: auto;
    }
}

```

---

**5/8/19**

You'll want to now fix the DiscussionWindow\_(Subject).php of each Subject discussion window for the upcoming procedure:

**DiscussionWindow (Subject).php**

```

<?php
    session_start();
?>
<html>
    <head>
        <title> (Subject) Discussion </title>
        <meta charset = "utf-8"/>

        <link rel = "stylesheet" href = "../ThreadsCssFiles/Navbar.css"/>
        <link rel = "stylesheet" href = "../ThreadsCssFiles/DiscussionWindow.css"/>
    </head>

    <body id = "(Subject)">
        <!-- navbar -->
        <ul class = "navbar">
            <li class = "left-tabs"><a href = "../forumhome.php"> Home </a></li>
            <li class = "left-tabs"><a href = "#"> Board </a></li>
            <li class = "left-tabs"><a href = "#"> Community </a></li>

```

```

        <li class = "right-tabs"><a href = "../includes/logout.php"> Logout </a></li>
        <li class = "dropdown right-tabs">
            <a href = "#" class = "dropdownButton"> Hello: <?php echo
$_SESSION['username']; ?> </a>
            <div class = "dropdown-content">
                <a href = "#"> <?php echo "<img src = '../.." . $_SESSION['pic'] . "' height='70'
width='70'/>"; ?> </a>
                <a href = "#"> Full Name: <?php echo $_SESSION['firstname'] . " " .
$_SESSION['lastname']; ?> </a>
                <a href = "#"> Email: <?php echo $_SESSION['useremail']; ?> </a>
                <a href = "#"> Current Status: <?php echo $_SESSION['status']; ?> </a>
                <a href = "#"> Join Date: <?php echo $_SESSION['regdate']; ?> </a>
                <a href = "../userprofile.php"> Go to account settings </a>
            </div>
        </li>
    </ul>

    <!-- thread window -->
    <div id = "thread">
        <header>
            <h3> Thread title: </h3>
            <h5> Date created: </h5>
            <h5> Subject: </h5>
            <h5> Topic: </h5>
        </header>

        <section>
            <div class = "userInfo">
                <p> Profile Picture </p>
                <p> Full Name </p>
                <p> Location </p>
                <p> Join Date </p>
                <p> Current Status </p>
            </div>

            <article>
                <h4> Content </h4>
            </article>
        </section>
    </div>

    <br/><br/><br/>

    <!-- replies window -->
    <div id = "replies">
        <header>

```

```

    <h5 style = "text-align: center"> Reply </h5>
  </header>

  <section>
    <div class = "userInfo">
      <p> Profile Picture </p>
      <p> Full Name </p>
      <p> Location </p>
      <p> Join Date </p>
      <p> Current Status </p>
    </div>

    <article>
      <h4> Content </h4>
    </article>
  </section>
</div>

<br/><br/><br/>

<!-- Comment box -->
<div id = "comment">
  <form method = "POST">
    <textarea rows = "10" cols = "70" name = "commentbox"></textarea> <br/>
    <input type = "submit" name = "submit-comment" value = "Submit comment!"/>
  </form>
</div>
</body>
</html>

```

### **DiscussionWindow.css**

```

...
/*The box-sizing property defines how the width and height of an element are calculated:
should they include padding and borders, or not.*/
*
{
  box-sizing: border-box;
}

/*sets the text style for all the text in the thread window and replies window*/
#thread, #replies
{
  font-family: Arial, Helvetica, sans-serif;

  margin-top: 5%;
  margin-left: 5%;
  margin-right: 5%;
}

```

```
}
/*styles the header*/
#thread header, #replies header
{
    background-color: #666;

    padding: 10px;
    font-size: 15px;
    color: white;
}
/*creates two columns/boxes that floats next to each other*/
#thread section .userinfo, #replies section .userinfo
{
    float: left;
    padding: 20px;
    width: 30%;
    height: auto;

    background-color: #ccc;
}
#thread section article, #replies section article
{
    float: left;
    padding: 20px;
    width: 70%;
    height: auto;

    background-color: #f1f1f1;
}
/*clear floats after the columns*/
#thread section:after, #replies section:after
{
    content: "";
    display: table;
    clear: both;
}
...
```

**6/6/19**

Okay, after a long break from this project because you had to learn a lot of SQL and how to manipulate and retrieve data from databases and what a foreign key is, now time to continue with this project and not spend a year completing it.

What you now want to do, now that you know what to do, is implement the code that allows users to create a thread on a particular subject, post that thread question on the page, then display the replies to the thread question on the page as well.

But before doing this (yes, you have to do so many things before you can actually get to the most important part), you have to do a lot of refactoring of the code. In the includes folder, you want to switch from procedural PHP to object oriented PHP. This way, you don't have to create so much php files to do a particular functionality. You can do it all in a single class.

What you want to do is, in the includes folder, create a php file called "AccountHandler.php", and this will handle the login and signup processes, so you can go ahead and delete the login.php and signup.php files and do it all in one file.

Another thing is, because you are not using the phpMyAdmin and storing all databases in the MySQL workbench, you will not be using MySQL via XAMPP (though you will still have to use the Apache web server that is included in XAMPP), you have to change the MySQL connection to the MySQL workbench. But before this (yes, a lot of steps), you will have to create a brand new database for the discussion forum that will contain the accounts, the created threads and replies (more on these both). This new database is called "discussionforum". Now, in the MySQL workbench, input the following SQL code to create the new database:

```
CREATE DATABASE discussionforum;
```

After creating the new database, next is to create the accounts table which will contain the user accounts:

```
create table accounts
(
    user_id int NOT NULL AUTO_INCREMENT,
    user_firstname varchar(100) NOT NULL,
    user_lastname varchar(100) NOT NULL,
    user_email varchar(100) NOT NULL,
    user_username varchar(100) NOT NULL,
    user_password varchar(100) NOT NULL,
    profile_pic varchar(255), /* stores the path of the uploaded profile pic */
    user_status varchar(100),
    reg_date TIMESTAMP, /* registration date when user first registered */
    PRIMARY KEY(user_id)
);
```

Now, in the database.php file, delete everything and instead, input the following:

**database.php**

```
<?php
class Connection
{
    private $servername;
    private $username;
    private $password;
    private $db;

    private $conn;

    public function __construct()
    {
        $this->servername = "127.0.0.1";
        $this->username = "root";
        $this->password = "ricmysql";
        $this->db = "discussionforum";
    }

    public function connect()
    {
        return $this->conn = new mysqli($this->servername, $this->username, $this->password, $this->db);
    }
}
?>
```

Now that the whole above thing is done, now continue with the AccountHandler.php file:

**AccountHandler.php**

```
<?php
include_once 'database.php';

/* This class will handle the login, logout and signup processes */
class Accounts
{
    private $dbConn;
    private $sql1; //for logging in new users to the forumhome.php
    private $sql2; //for adding new users to the database

    /* initialize connection to the discussionforum database upon calling the object of the class */
    public function __construct()
    {
        $this->dbConn = new Connection();
        $this->dbConn = $this->dbConn->connect();
    }
}
```

```

}

/* This method is for logging in new users into the forum */
public function login($username, $password)
{
    if(empty($username) && empty($password))
    {
        echo "<p id = 'errormessage'> Do not submit blank information! </p>";
    }
    else if(empty($username))
    {
        echo "<p id = 'errormessage'> You must enter your username! </p>";
    }
    else if(empty($password))
    {
        echo "<p id = 'errormessage'> You must enter your password! </p>";
    }
    else
    {
        $this -> sql1 = "SELECT * FROM accounts WHERE user_username=? OR
user_password=?";

        $stmt = $this -> dbConn -> stmt_init();

        if(!$stmt -> prepare($this -> sql1))
        {
            echo "<script> alert('SQL ERROR: " . $stmt -> error . "'); </script>";
        }
        else
        {
            $stmt -> bind_param("ss", $username, $password);
            $stmt -> execute();

            $result = $stmt -> get_result();

            if($row = $result -> fetch_assoc()) //store the results in an associative array
            {
                session_start();

                /*obtain the id, username, email, firstname, lastname as well as profile picture
                and status and registration date, and stores them in session variables*/
                $_SESSION['userid'] = $row['user_id'];
                $_SESSION['firstname'] = $row['user_firstname'];
                $_SESSION['lastname'] = $row['user_lastname'];
                $_SESSION['useremail'] = $row['user_email'];
                $_SESSION['username'] = $row['user_username'];
            }
        }
    }
}

```



```

        $_SESSION['pic'] = $row['profile_pic'];
        $_SESSION['status'] = $row['user_status'];
        $_SESSION['regdate'] = $row['reg_date'];

        //sends user to the forum homepage upon successful login
        header("Location: forumhome.php?login=successful");
        exit();
    }
    else
    {
        echo "<p id = 'errormessage'> There is no such account! </p>";
    }
}
}

/* This method is for registering new users and adding their information to the accounts
table */
public function signup($firstname, $lastname, $email, $username, $password)
{
    if(empty($firstname) && empty($lastname) && empty($email) && empty($username)
&& empty($password))
    {
        echo "<p id = 'errormessage'> Do not submit empty information! </p>";
    }
    else if(empty($lastname) && empty($email) && empty($username) &&
empty($password))
    {
        echo "<p id = 'errormessage'> You only filled in the first name field! </p>";
    }
    else if(empty($firstname) && empty($email) && empty($username) &&
empty($password))
    {
        echo "<p id = 'errormessage'> You only filled in the last name field! </p>";
    }
    else if(empty($firstname) && empty($lastname) && empty($username) &&
empty($password))
    {
        echo "<p id = 'errormessage'> You only filled in the email field! </p>";
    }
    else if(empty($firstname) && empty($lastname) && empty($email) &&
empty($password))
    {
        echo "<p id = 'errormessage'> You only filled out the username field! </p>";
    }
}

```

```

        else if(empty($firstname) && empty($lastname) && empty($email) &&
empty($username))
        {
            echo "<p id = 'errormessage'> You only filled out the password field! </p>";
        }
        else
        {
            $this -> sql2 = "INSERT INTO accounts(user_firstname, user_lastname, user_email,
user_username, user_password)
                VALUES(?, ?, ?, ?, ?);";

            $stmt = $this -> dbConn -> stmt_init();

            if(!$stmt -> prepare($this -> sql2))
            {
                echo "<script> alert('SQL ERROR: " . $stmt -> error . "'); </script>";
            }
            else
            {
                $hashedPwd = password_hash($password, PASSWORD_DEFAULT);

                $stmt -> bind_param("sssss", $firstname, $lastname, $email, $username,
$hashedPwd);
                $stmt -> execute();

                echo "<p id = 'successmessage'> You have successfully signed up! Sign in now to
enter forum discussion! </p>";
            }
        }
    }
}
?>

```

After that, now you want to add the following to the loginPage.php and signupPage.php:

### loginPage.php

```
<?php
include "includes/AccountHandler.php";
$account = new Accounts();
?>
<html>
  <head>
    <title> Login to account </title>
    <meta charset = "utf-8"/>

    <link rel = "stylesheet" href = "css/login.css"/>
    <link href="https://fonts.googleapis.com/css?family=Pacifico" rel="stylesheet">
  </head>

  <body>
    <div id = "headings">
      <h1> Welcome to the Discussion forums </h1>
      <h4> Please login to enter </h4>
    </div>

    <div id = "loginwindow">
      <h4> Member Login </h4>

      <form method = "POST" action = "loginPage.php">
        <input type = "text" placeholder = "Enter Username" name = "loginuname"/>
        <br/>
        <input type = "password" placeholder = "Enter Password" name = "loginpwd"/>
        <br/>
        <input type = "submit" value = "Login" name = "loginButton"/>
      </form>

      <?php
        if(isset($_POST['loginButton']))
        {
          $uname = $_POST['loginuname'];
          $pwd = $_POST['loginpwd'];

          $account -> login($uname, $pwd);
        }
      ?>

      <a href = "#"><h4> Forgot password? Click here </h4></a>
      <a href = "signupPage.php"><h4> Don't have an account yet? Click here to sign up
for an account </hr></a>
    </div>
```

```
</body>
</html>
```

### signupPage.php

```
<?php
include "includes/AccountHandler.php";
$account = new Accounts();
?>
<html>
  <head>
    <title> Register an account now! </title>
    <meta charset = "utf-8"/>

    <link rel = "stylesheet" href = "css/signup.css"/>
  </head>

  <body>
    <div id = "headings">
      <h1> Welcome </h1>
      <h4> Please sign up now to join our discussion forum </h4>
    </div>

    <div id = "signupwindow">
      <h3> Member Signup </h3>

      <form method = "POST" action = "signupPage.php">
        <input type = "text" placeholder = "Enter First Name" name = "fname"/>
        <br/>
        <input type = "text" placeholder = "Enter Last Name" name = "lname"/>
        <br/>
        <input type = "text" placeholder = "Enter your Email" name = "email"/>
        <br/>
        <input type = "text" placeholder = "Enter Desired Username" name = "uname"/>
        <br/>
        <input type = "password" placeholder = "Enter Desired Password" name =
"pwd"/>
        <br/>
        <input type = "submit" value = "Register" name = "submitButton"/>
      </form>

      <?php
        if(isset($_POST['submitButton']))
        {
          $fname = $_POST['fname'];
          $lname = $_POST['lname'];
          $email = $_POST['email'];
          $uname = $_POST['uname'];
```

```

        $pwd = $_POST['pwd'];

        $account -> signup($fname, $lname, $email, $uname, $pwd);
    }
?>

    <a href = 'loginPage.php'> <h5> Already have an account, log in now </h5>
</a></div>
</body>
</html>

```

### **6/7/19 – 6/8/19**

What you now want to do is create a php file in the includes folder called “ProfileHandler.php” and this file will take place of the updatestatus.php and uploadPics.php, meaning the ProfileHandler.php will both update the information of the user and upload a profile picture (remember, you want to change everything from procedural code to object oriented code for organization and to reduce the number of php files). You also want to fix the userprofile.php page afterwards.

#### **ProfileHandler.php**

```

<?php
    include_once 'database.php';

    /* This class updates user information such as name, email and profile picture */
    class ProfileHandler
    {
        private $db; //to connect to the discussionforum database

        /* These variables are reserved for the uploadPics() method */
        private $sql1; //for uploading profile pictures
        private $photo = ""; //obtains the photo to be uploaded (initialize to nothing until user
selects a picture to upload)
        private $imageFileType = ""; //obtain the type of file (".jpg", ".png", etc...) (initialize to
nothing until user selects a picture to upload)
        private $folder; //the folder to store the uploaded pictures
        private $check; //checks the size of the image

        /* These variables are reserved for the updateStatus() method */
        private $sql2; //for updating user information
        private $newUsername = ""; //sets new user name
        private $newPassword = ""; //sets new user password
        private $newEmail = ""; //sets new user email
        private $setStatus = ""; //sets new user status

        /* Initialize connection to the discussionforum database upon creating the object of this
class */

```

```

public function __construct()
{
    $this -> db = new Connection();
    $this -> db = $this -> db -> connect();
}

/* This method uploads new profile pictures */
public function uploadPics($id)
{
    $this -> folder = "ProfilePics/";
    $this -> photo = $this -> folder . basename($_FILES["getFile"]["name"]); //obtains the
photo to be updated and store them in the ProfilePics folder
    $this -> imageFileType = strtolower(pathinfo($this -> photo,
PATHINFO_EXTENSION)); //obtain the path of the type of file (".jpg", ".png", etc...)
    $this -> check = getimagesize($_FILES["getFile"]["tmp_name"]); //checks the size of
the image

    //check if the file is a real image or not
    if($this -> check !== false)
    {
        //if the image file is not jpg, jpeg, png and gif files
        if($this -> imageFileType != "jpg" && $this -> imageFileType != "png" && $this ->
imageFileType != "jpeg" && $this -> imageFileType != "gif")
        {
            echo "<p id = 'errMessage'> Sorry, only JPG, JPEG, PNG, and GIF files are allowed!
</p>";
        }
        else //else if the image file is a jpg, jpeg, png, and/or gif
        {
            //if the file is too large
            if($_FILES["getFile"]["size"] > 500000)
            {
                echo "<p id = 'errMessage'> Sorry, your file is too big to upload! </p>";
            }
            else //else if file is not too large
            {
                $pic = $this -> photo; //this variable is just a placeholder to hold the photo variable

                //upload the image
                if(move_uploaded_file($_FILES["getFile"]["tmp_name"], $this -> photo))
                {
                    $userid = intval($id); //gets the user id session variable. intval gets the integer
value (the id)

                    $this -> sql1 = "UPDATE accounts SET profile_pic='$pic' WHERE
user_id='$id'";

```

```

        if(!$this->db->query($this->sql1))
        {
            echo "<script> alert('SQL ERROR: " . $this->db->error . "'); </script>";
        }
        else
        {
            echo "<p id = 'succMessage'> Your picture was successfully uploaded! <br/>
                In order to view your new picture, you will have to logout, then log back
in!
                </p>";

        }
    }
    //if the image could not be uploaded
    else
    {
        echo "<p id = 'errMessage'> Sorry, there was a problem uploading your image
file </p>";
    }
}
}
}
else
{
    echo "<p id = 'errMessage'> The image you selected is not a file! </p>";
}
}

/* This method updates user information such as name, email, password, etc */
public function updateInfo($userid, $newUsername, $newPassword, $newEmail,
$setStatus)
{
    //if the username field is not empty, but the others are, change only the username
    if(empty($newPassword) && empty($newEmail) && empty($setStatus))
    {
        $this->newUsername = $newUsername; //private $newUsername =
updateInfo($newUsername)

        $id = intval($userid);

        $this->sql2 = "UPDATE accounts SET user_username='$this->newUsername'
WHERE user_id='$id'";

```

```

        if(!$this -> db -> query($this -> sql2))
        {
            echo "<script> alert('SQL ERROR: " . $this -> db -> error . "'); </script>";
        }
        else
        {
            echo "<p id = 'succMessage'> Username has been changed! </p>";
        }
    }
    //if the password field is not empty, but the others are, change only the password
    else if(empty($newUsername) && empty($newEmail) && empty($setStatus))
    {
        $this -> newPassword = $newPassword; //private $newPassword =
updateInfo($newPassword)
        $hashedPwd = password_hash($this -> newPassword, PASSWORD_DEFAULT);
//must hash the password to prevent hackers from seeing user password

        $id = intval($userid);

        $this -> sql2 = "UPDATE accounts SET user_password='$hashedPwd' WHERE
user_id='$id'";

        if(!$this -> db -> query($this -> sql2))
        {
            echo "<script> alert('SQL ERROR: " . $this -> db -> error . "'); </script>";
        }
        else
        {
            echo "<p id = 'succMessage'> Password has been changed! </p>";
        }
    }
    //if the email field is not empty, but the others are, change only the email
    else if(empty($newUsername) && empty($newPassword) && empty($setStatus))
    {
        $this -> newEmail = $newEmail; //private $newEmail = updateInfo($newEmail)

        $id = intval($userid);

        $this -> sql2 = "UPDATE accounts SET user_email='$this->newEmail' WHERE
user_id='$id'";

        if(!$this -> db -> query($this -> sql2))
        {
            echo "<script> alert('SQL ERROR: " . $this -> db -> error . "'); </script>";
        }
        else
    }

```



```

    {
        echo "<p id = 'succMessage'> Email has been changed! </p>";
    }
}
//if the status is not empty, but the others are, change only the status
else if(empty($newUsername) && empty($newPassword) && empty($newEmail))
{
    $this->setStatus = $setStatus; //private $setStatus = updateInfo($setStatus)

    $id = intval($userid);

    $this->sql2 = "UPDATE accounts SET user_status='$this->setStatus' WHERE
user_id='$id'";

    if(!$this->db->query($this->sql2))
    {
        echo "<script> alert('SQL ERROR: " . $this->db->error . "'); </script>";
    }
    else
    {
        echo "<p id = 'succMessage'> Status has been updated! </p>";
    }
}
//if only the username field is blank, but the rest is not, change the password, email and
status
else if(empty($newUsername))
{
    $this->newPassword = $newPassword;
    $this->newEmail = $newEmail;
    $this->setStatus = $setStatus;

    $hashedPwd = password_hash($this->newPassword, PASSWORD_DEFAULT);

    $id = intval($userid);

    $this->sql2 = "UPDATE accounts SET user_password='$hashedPwd',
user_email='$this->newEmail', user_status='$this->setStatus' WHERE user_id='$id'";

    if(!$this->db->query($this->sql2))
    {
        echo "<script> alert('SQL ERROR: " . $this->db->error . "'); </script>";
    }
    else
    {
        echo "<p id = 'succMessage'> The password and email have been changed and the
status has been updated! </p>";
    }
}

```

```

    }
  }
  //if only the password field is blank, but the rest is not, change the username, email and
status
  else if(empty($newPassword))
  {
    $this -> newUsername = $newUsername;
    $this -> newEmail = $newEmail;
    $this -> setStatus = $setStatus;

    $sid = intval($userid);

    $this -> sql2 = "UPDATE accounts SET user_username='$this->newUsername',
user_email='$this->newEmail', user_status='$this->setStatus' WHERE user_id='$sid'";

    if(!$this -> db -> query($this -> sql2))
    {
      echo "<script> alert('SQL ERROR: " . $this -> db -> error . "'); </script>";
    }
    else
    {
      echo "<p id = 'succMessage'> The username and email have been changed and the
status has been updated! </p>";
    }
  }
  //if only the email field is blank, but the rest is not, change the username, password and
status
  else if(empty($newEmail))
  {
    $this -> newUsername = $newUsername;
    $this -> newPassword = $newPassword;
    $this -> setStatus = $setStatus;

    $hashedPwd = password_hash($this -> newPassword, PASSWORD_DEFAULT);

    $sid = intval($userid);

    $this -> sql2 = "UPDATE accounts SET user_username='$this->newUsername',
user_password='$hashedPwd', user_status='$this->setStatus' WHERE user_id='$sid'";

    if(!$this -> db -> query($this -> sql2))
    {
      echo "<script> alert('SQL ERROR: " . $this -> db -> error . "'); </script>";
    }
    else
    {

```

```

        echo "<p id = 'succMessage'> The username and password have been changed and
the status has been updated! </p>";
    }
}
//if only the status field is blank, but the rest is not, change the username, password and
email
else if(empty($setStatus))
{
    $this -> newUsername = $newUsername;
    $this -> newEmail = $newEmail;
    $this -> newPassword = $newPassword;

    $hashedPwd = password_hash($this -> newPassword, PASSWORD_DEFAULT);

    $id = intval($userid);

    $this -> sql2 = "UPDATE accounts SET user_username='$this->newUsername',
user_email='$this->newEmail', user_password='$hashedPwd' WHERE user_id='$id'";

    if(!$this -> db -> query($this -> sql2))
    {
        echo "<script> alert('SQL ERROR: " . $this -> db -> error . "'); </script>";
    }
    else
    {
        echo "<p id = 'succMessage'> The username, password and email have been
changed! </p>";
    }
}
//change only the username and password if the email and status fields are blank
else if(empty($newEmail) && empty($setStatus))
{
    $this -> newUsername = $newUsername;
    $this -> newPassword = $newPassword;

    $hashedPwd = password_hash($this -> newPassword, PASSWORD_DEFAULT);

    $id = intval($userid);

    $this -> sql2 = "UPDATE accounts SET user_username='$this->newUsername',
user_password='$hashedPwd' WHERE user_id='$id'";

    if(!$this -> db -> query($this -> sql2))
    {
        echo "<script> alert('SQL ERROR: " . $this -> db -> error . "'); </script>";
    }
}

```

```

else
{
    echo "<p id = 'succMessage'> The username and password has been changed! </p>";
}
}
//change only the username and email if the password and status fields are empty
else if(empty($newPassword) && empty($setStatus))
{
    $this->newUsername = $newUsername;
    $this->newEmail = $newEmail;

    $id = intval($userid);

    $this->sql2 = "UPDATE accounts SET user_username='$this->newUsername',
user_email='$this->newEmail' WHERE user_id='$id'";

    if(!$this->db->query($this->sql2))
    {
        echo "<script> alert('SQL ERROR: " . $this->db->error . "'); </script>";
    }
    else
    {
        echo "<p id = 'succMessage'> The username and email has been changed! </p>";
    }
}
//change only the username and status if the password and email fields are empty
else if(empty($newPassword) && empty($newEmail))
{
    $this->newUsername = $newUsername;
    $this->setStatus = $setStatus;

    $id = intval($userid);

    $this->sql2 = "UPDATE accounts SET user_username='$this->newUsername',
user_status='$this->setStatus' WHERE user_id='$id'";

    if(!$this->db->query($this->sql2))
    {
        echo "<script> alert('SQL ERROR: " . $this->db->error . "'); </script>";
    }
    else
    {
        echo "<p id = 'succMessage'> The username and status has been changed! </p>";
    }
}
//change only the password and email if the username and status fields are empty

```

```

else if(empty($newUsername) && empty($newEmail))
{
    $this -> newPassword = $newPassword;
    $this -> newEmail = $newEmail;

    $hashedPwd = password_hash($this -> newPassword, PASSWORD_DEFAULT);

    $id = intval($userid);

    $this -> sql2 = "UPDATE accounts SET user_password='$hashedPwd',
user_email='$this->newEmail' WHERE user_id='$id'";

    if(!$this -> db -> query($this -> sql2))
    {
        echo "<script> alert('SQL ERROR: " . $this -> db -> error . "'); </script>";
    }
    else
    {
        echo "<p id = 'succMessage'> The password and email has been changed! </p>";
    }
}
//change only the password and status if the username and email fields are empty
else if(empty($newUsername) && empty($newEmail))
{
    $this -> newPassword = $newPassword;
    $this -> setStatus = $setStatus;

    $hashedPwd = password_hash($this -> newPassword, PASSWORD_DEFAULT);

    $id = intval($userid);

    $this -> sql2 = "UPDATE accounts SET user_password='$hashedPwd',
user_status='$this->setStatus' WHERE user_id='$id'";

    if(!$this -> db -> query($this -> sql2))
    {
        echo "<script> alert('SQL ERROR: " . $this -> db -> error . "'); </script>";
    }
    else
    {
        echo "<p id = 'succMessage'> The password and status has been changed! </p>";
    }
}
//change only the email and status if the username and password fields are empty
else if(empty($newUsername) && empty($newPassword))
{

```

```

$this->newEmail = $newEmail;
$this->setStatus = $setStatus;

$id = intval($userid);

$this->sql2 = "UPDATE accounts SET user_email='$this->newEmail',
user_status='$this->setStatus' WHERE user_id='$id'";

if(!$this->db->query($this->sql2))
{
    echo "<script> alert('SQL ERROR: " . $this->db->error . "'); </script>";
}
else
{
    echo "<p id = 'succMessage'> The email and status has been changed! </p>";
}
}
}
}
?>

```

Now add the following to the `userprofile.php` and `userprofile.css` pages:

```

userprofile.php
<?php
    session_start(); //needed for starting session
?>

<html>
    <head>
        <title> Welcome to your profile </title>
        <meta charset = "utf-8"/>

        <link rel = "stylesheet" href = "css/Navbar.css"/>
        <link rel = "stylesheet" href = "css/userprofile.css"/>
    </head>

    <body>
        <!-- navbar -->
        <ul class = "navbar">
            <li class = "left-tabs"><a href = "forumhome.php"> Home </a></li>
            <li class = "left-tabs"><a href = "#"> Board </a></li>
            <li class = "left-tabs"><a href = "#"> Community </a></li>

            <li class = "right-tabs"><a href = "includes/logout.php"> Logout </a></li>
            <li class = "dropdown right-tabs">

```

```

        <a href = "#" class = "dropdownButton"> Hello: <?php echo
$_SESSION['username']; ?> </a>
        <div class = "dropdown-content">
            <a href = "#"> <?php echo "<img src = '" . $_SESSION['pic'] . "'" height='70'
width='70'/>"; ?> </a>
            <a href = "#"> Full Name: <?php echo $_SESSION['firstname'] . " " .
$_SESSION['lastname']; ?> </a>
            <a href = "#"> Email: <?php echo $_SESSION['useremail']; ?> </a>
            <a href = "#"> Current Status: <?php echo $_SESSION['status']; ?> </a>
            <a href = "#"> Join Date: <?php echo $_SESSION['regdate']; ?> </a>
            <a href = "userprofile.php"> Go to account settings </a>
        </div>
    </li>
</ul>

<!-- Displays profile image and settings for changing picture -->
<div id = "profileimage-window">
    <?php echo "<img src = '" . $_SESSION['pic'] . "'" height='150' width='150'/>"; ?>
    <br/>
    <form action = 'userprofile.php' method = 'POST' enctype = 'multipart/form-data'>
        <h3> Select new image to upload: </h3>

        <input type = 'file' name = 'getFile'/>
        <input type = 'submit' value = 'Upload Photo' name = 'submit-pic'/>
    </form>

    <?php
        include 'includes/ProfileHandler.php'; //needed for uploading user information
and profile picture
        $update = new ProfileHandler();

        $userid = $_SESSION['userid'];

        if(isset($_POST['submit-pic']))
        {
            $update -> uploadPics($userid);
        }
    ?>
</div>

<!-- Settings for changing user info -->
<div id = "settings-window">
    <h3> Change Info </h3>
    <form action = 'userprofile.php' method = "POST">
        <br/><br/>

```

```

        <label class = "userstats"> Username: <input type = "text" name =
"changeUsername"/> </label>
        <br/><br/>
        <label class = "userstats"> Password: <input type = "text" name =
"changePassword"/> </label>
        <br/><br/>
        <label class = "userstats"> Email: <input type = "text" name = "changeEmail"/>
</label>
        <br/><br/>
        <label class = "userstats"> Current Status: <input type = "text" name = "status"/>
</label>
        <br/><br/>
        <input type = "submit" name = "new-info" value = "Set info"/>
</form>

<?php
    if(isset($_POST['new-info']))
    {
        $newuname = $_POST['changeUsername'];
        $newpwd = $_POST['changePassword'];
        $newemail = $_POST['changeEmail'];
        $newstatus = $_POST['status'];

        $update -> updateInfo($userid, $newuname, $newpwd, $newemail,
$newstatus);
    }
?>
</div>

<!-- Displays messages from other users and messages sent -->
<div id = "message-window">
    <!-- Messages from other users -->
    <div id = "messageinbox">
        <h4> Your message inbox: </h4>
        <br/>
        <table>
            <tr>
                <th> From </th>
                <th> Subject </th>
                <th> Topic </th>
                <th> Date sent </th>
                <th> Status </th>
                <th> Message </th>
            </tr>
            <tr>
                <td> from </td>

```



```

        <td> subject </td>
        <td> topic </td>
        <td> date sent </td>
        <td> status </td>
        <td> message </td>
    </tr>
</table>
</div>
<br/>
<!-- Messages you sent -->
<div id = "sentmessages">
    <h4> Messages you have sent: </h4>
    <br/>
    <table>
        <tr>
            <th> To </th>
            <th> Subject </th>
            <th> Topic </th>
            <th> Date sent </th>
            <th> Status </th>
            <th> Message </th>
        </tr>
        <tr>
            <td> to </td>
            <td> subject </td>
            <td> topic </td>
            <td> date sent </td>
            <td> status </td>
            <td> message </td>
        </tr>
    </table>
</div>
</div>

<!-- Option for sending private message -->
<div id = "privatemessage-window">
    <h3> Send Private Message </h3>

    <form method = "POST" id = "sendpm">
        <label> Enter username of forum member: <input type = "text"/> </label>
        <br/>
        <label> Enter subject: <input type = "text"/> </label>
        <br/>
        <label> Enter topic: <input type = "text"/> </label>
        <br/><br/>
        <label> Enter Message: </label>
    </form>
</div>

```

```

        <br/>
        <textarea name = "pm-message"></textarea>
        <br/><br/>
        <input type = "submit" value = "Send PM"/>
    </form>
</div>

<!-- List of threads you last made -->
<div id = "user-posts">
    <table>
        <tr>
            <th> Last thread you posted: </th>
            <th> Subject of last thread: </th>
            <th> Date of thread posted: </th>
        </tr>
        <tr>
            <td> </td>
            <td> </td>
            <td> </td>
        </tr>
    </table>
</div>
</body>
</html>

```

### **userprofile.css**

```

#profileimage-window, #settings-window, #message-window, #privatemessage-window,
#user-posts
{
    border: 1px solid black;
    margin-bottom: 20px;
}

/*styles the profileimage-window widow*/
#profileimage-window
{
    margin-top: 10px;
    padding: 30px;
}
#profileimage-window input[type=submit], #profileimage-window input[type=file]
{
    font-family: serif;
    background-color: lightgreen;
    border: none;
    border-radius: 3px;
}
#profileimage-window #errorMessage, #settings-window #errorMessage

```

```

{
    color: red;
    font-family: serif;
}
#profileimage-window #succMessage, #settings-window #succMessage
{
    color: green;
    font-family: serif;
}

/*styles the settings-window window*/
#settings-window
{
    padding: 30px;
}
#settings-window h3
{
    text-align: center;
}
#settings-window .userstats
{
    font-family: sans-serif;
}
#settings-window .userstats input[type=text]
{
    font-family: serif;
    width: 50%;
    height: 30px;
    border: 1px solid #ccc;
    border-radius: 4px;
}
#settings-window input[type=submit]
{
    font-family: serif;
    margin-left: 40%;
    background-color: lightgreen;
    border: none;
    border-radius: 3px;
}

/*styles the message-window window*/
#message-window
{
    padding: 0 10px 20px 10px;
}
#message-window #messageinbox table, #message-window #sentmessages table

```

```

{
    font-family: arial, sans-serif;
    border-collapse: collapse;
    width: 100%;
}
#message-window #messageinbox td, #message-window #messageinbox th, #message-
window #sentmessages td, #message-window #sentmessages th
{
    border: 1px solid #dddddd;
    text-align: left;
    padding: 8px;
}

/*styles the privatemessgae-window window*/
#privatemessage-window
{
    padding: 30px;
}
#privatemessage-window label
{
    font-family: sans-serif;
}
#privatemessage-window input[type=text]
{
    font-family: serif;
    width: 50%;
    height: 30px;
    border: 1px solid #ccc;
    border-radius: 4px;
}
#privatemessage-window textarea
{
    font-family: serif;
    width: 70%;
    height: 30%;
    border: 1px solid #ccc;
    border-radius: 4px;
}
#privatemessage-window input[type=submit]
{
    font-family: serif;
    margin-left: 40%;
    background-color: lightgreen;
    border: none;
    border-radius: 3px;
}

```

```
/*styles the user-posts window*/
#user-posts
{
    padding: 20px;
}
#user-posts table
{
    font-family: arial, sans-serif;
    border-collapse: collapse;
    width: 100%;
}
#user-posts th, #user-posts td
{
    border: 1px solid #dddddd;
    text-align: left;
    padding: 8px;
}
```

You can now delete the `updatestatus.php` and `uploadPics.php` files since you don't need them.

---

### **6/9/19 – 6/11/19**

Now getting back to the task on hand and that is implementing the code that will allow users to create threads on a particular subject and allowing other users to reply to their thread. This will allow a lot of thinking beforehand, so don't go jumping straight into coding.

**6/9/19, 6/10/19**

**Step 1:** Create two new tables in the discussionforum database: threads and replies

- threads
  - user\_id (references the user who made the thread)
  - thread\_id
  - thread\_author (contains the username of the user who created the thread)
  - thread\_subject
  - thread\_topic
  - thread\_title
  - thread\_content
- replies
  - user\_id (references the user who made the reply)
  - replies\_id
  - thread\_id (references the user who made the thread)
  - replies\_content

```
CREATE TABLE threads
(
    user_id int NOT NULL,
    thread_id int NOT NULL AUTO_INCREMENT,
    thread_author varchar(100) NOT NULL,
    thread_subject varchar(100) NOT NULL,
    thread_topic varchar(100) NOT NULL,
    thread_title varchar(100) NOT NULL,
    thread_content text NOT NULL,
    PRIMARY KEY(thread_id),
    FOREIGN KEY(user_id) REFERENCES accounts(user_id)
);

CREATE TABLE replies
(
    user_id int NOT NULL,
    thread_id int NOT NULL,
    replies_id int NOT NULL,
    replies_content text NOT NULL,
    PRIMARY KEY(replies_id),
    FOREIGN KEY(user_id) REFERENCES accounts(user_id),
    FOREIGN KEY(thread_id) REFERENCES threads(thread_id)
);
```

You should now have the following tables in the *discussionforum* database:

- threads

user_id	thread_id	thread_author	thread_subject	thread_topic	thread_title	thread_content
---------	-----------	---------------	----------------	--------------	--------------	----------------

- replies

user_id	thread_id	replies_id	replies_content
---------	-----------	------------	-----------------

Step 2: Change the name of the "Thread" folder to "Subjects" (because it's kind of misleading and you're going to use a similar named folder later on)

- It will contain the same subject folders: Culture, Natural Science and Mathematics, Social Science, Technology, World News.
- It will contain the same CSS and JavaScript folders: ThreadsCssFiles, ThreadsJSFiles.

You will have to change the links in the forumhome.php, from "Threads/..." to "Subjects/..."

However, inside each of the subject folders of the Subjects folder, you want to have the following folders for each particular subject folder:

- Culture -> Culture Threads
- Natural Science and Mathematics -> SciMath Threads
- Social Science -> SoSci Threads
- Technology -> Tech Threads
- World News -> International Threads

These "(Subject) Thread" folders will contain the created threads of the particular subject. Meaning, when a user creates a thread for a particular subject in the Add\_(subject)\_thread.php page, that thread gets stored in the "(Subject) Thread" folder.

**6/10/19**

The 'DiscussionWindow\_(topic)\_(title).php' page describes the layout of what the thread window will look like.

Thus the file should be named: “*DiscussionWindow\_(topic)\_(title).php*”, and must be stored in their respective (Subject) folders.

**Step 3:** Create the file called "ThreadHandler.php" and save it inside the includes folder.

This file will contain the ThreadHandler class, which will handle the thread creation process.

It will contain two methods: public function createThread() and public function displayThread():

- The createThread() method will be responsible for creating the thread for that particular subject and storing them in the "(Subject) Thread" folders.

How this method will work:

- The method will then get that thread information from the threads table, then append that data into a thread file called '*DiscussionWindow\_(topic)\_(title).php*', and will store that thread in its respective subject folder.
- This method will then store the username of the user who created the topic thread, the subject of the topic thread, the title of the thread, and the content of the thread in the threads table of the discussionforum database.

- The displayThread() method will be responsible for displaying the number of threads on the forum pages.

How this will work is:

- It will look at the thread table and through the subject folders, see how many threads are in that, then display that number in the topic.php pages, as well as displaying all the threads contained the subject folder on the topic pages. It will also check to see the last thread that was created and display that on the forumhome.php page.



**6/11/19 – 6/13/19**

Another thing you want to do is create another column for the threads table:

```
ALTER TABLE threads
ADD thread_creation TIMESTAMP;
```

You should have the following threads table:

user_id	thread_id	thread_author	thread_subject	thread_topic	thread_title	thread_content	thread_creation
---------	-----------	---------------	----------------	--------------	--------------	----------------	-----------------

The thread\_creation is the time the thread was created.

Now continue with the ThreadHandler.php:

**ThreadHandler.php**

```
<?php
    include_once 'database.php';

    /* This class will handle the processes of both creating the topic thread of a particular
    subject
    and displaying it on the forumhome.php and the subject pages */
    class ThreadHandler
    {
        private $db;

        //SQL variables
        private $sql1; //for the createThread() method. For storing the thread information in the
        thread table.
        private $sql2; //for the displayThread() method. For obtaining the thread information from
        the thread table and displaying it in their respective topic thread file

        public function __construct()
        {
            $this -> db = new Connection();
            $this -> db = $this -> db -> connect();
        }

        /* This method is responsible for obtaining the thread information and storing that
        information in the threads table */
        public function createThread($directory, $id, $author, $subject, $topic, $title, $content)
        {
            /* Before adding the thread information into the database, first append that information
            into a file called 'DiscussionWindow_(topic)_(title).php' */
            $threadfile = fopen("$directory/DiscussionWindow_$topic" . "_" . $title.php", "w");
            $profilepic = "<?php echo " . "\"<img src = '../..../". $_SESSION['pic'] . "' height='70'
            width='70'/>"; ?>";

            //set the thread page layout for the 'DiscussionWindow_(topic)_(title).php' page
```

```

$threadlayout = "<?php
    session_start();
?>
<html>
    <head>
        <title> $title </title>
        <meta charset = 'utf-8'/>

        <link rel = 'stylesheet' href = '../ThreadsCssFiles/Navbar.css'/>
        <link rel = 'stylesheet' href =
'../ThreadsCssFiles/DiscussionWindow.css'/>
    </head>

    <body id = '$subject'>
        <!-- Navbar (shortened version because too hard to add the whole
navbar)-->
        <ul class = 'navbar'>
            <li class = 'left-tabs'> <a href = '../forumhome.php'> Home </a>
</li>

            <li class = 'left-tabs'> <a href = '#'> Board </a> </li>
            <li class = 'left-tabs'> <a href = '#'> Community </a> </li>

            <li class = 'right-tabs'> <a href = '../userprofile.php'> Go to
profile </a> </li>
            <li class = 'right-tabs'> <a href = '../includes/logout.php'> Logout
</a> </li>

        </ul>

        <!-- Thread window -->
        <div id = 'thread'>
            <header>
                <h3> $title </h3>
                <h5> Date created: <?php echo " . "$" . "_SESSION" .
"['thread_creation'];" . " ?> </h5>
                <h5> $subject </h5>
                <h5> $topic </h5>
            </header>

            <section>
                <div class = 'userinfo'>
                    <p> <?php echo " . "$" . "_SESSION" . "['username']; ?>" .
"</p>

                    <p> $profilepic </p>
                    <p> Join date: <?php echo " . "$" . "_" .
"SESSION['regdate']; ?>" . "</p>

```

```

        <p> <?php echo " . "$" . "_" . "SESSION" . "["status"]; ?>" .
" </p>

        </div>

        <article>
            $content
        </article>
    </section>

    <br/><br/><br/>

    <!-- Comment box -->
    <div id = 'comment'>
        <form method = 'POST'>
            <textarea rows = '10' cols = '70' name =
'commentbox'></textarea> <br/>
            <input type = 'submit' name = 'submit-comment' value =
'Submit comment!'/>
        </form>
    </div>
</div>
</body>
</html>";

    fwrite($threadfile, $threadlayout);

    /* Now add the thread information to the database */
    $this -> sql1 = "INSERT INTO threads(user_id, thread_author, thread_subject,
thread_topic, thread_title, thread_content)
        VALUES('$id', '$author', '$subject', '$topic', '$title', '$content');"

    $stmt1 = $this -> db -> stmt_init();

    if(!$stmt1 -> prepare($this -> sql1))
    {
        echo "<script> alert('SQL ERROR: ". $stmt1 -> error . "'); </script>";
    }
    else
    {
        $stmt1 -> execute();
        echo "<script> alert('Thread file has been created!'); </script>";
    }
}

```

```

    /* This method is responsible for displaying the topic thread on the forumhome.php page
    and the subject pages */
    public function displayThread()
    {

    }
}
?>

```

Now what you want to do is, in each of the Add\_(Subject)\_thread.php file in the subject folders, you want to add the following PHP code segment:

#### **Add (Subject) thread.php**

```

<?php
    session_start();
?>
<html>
    <head>
        <title> New Topic --- (Subject) </title>
        <meta charset = "utf-8"/>

        <link rel = "stylesheet" href = "../ThreadsCssFiles/Navbar.css"/>
        <link rel = "stylesheet" href = "../ThreadsCssFiles/addWindow.css"/>
    </head>

    <body>
        <!-- navbar -->
        <ul class = "navbar">
            <li class = "left-tabs"><a href = "../forumhome.php"> Home </a></li>
            <li class = "left-tabs"><a href = "#"> Board </a></li>
            <li class = "left-tabs"><a href = "#"> Community </a></li>

            <li class = "right-tabs"><a href = "../includes/logout.php"> Logout </a></li>
            <li class = "dropdown right-tabs">
                <a href = "#" class = "dropdownButton"> Hello: <?php echo
$_SESSION['username']; ?> </a>
                <div class = "dropdown-content">
                    <a href = "#"> <?php echo "<img src = '../.." . $_SESSION['pic'] . "' height='70'
width='70'/>"; ?> </a>
                    <a href = "#"> Full Name: <?php echo $_SESSION['firstname'] . " " .
$_SESSION['lastname']; ?> </a>
                    <a href = "#"> Email: <?php echo $_SESSION['useremail']; ?> </a>
                    <a href = "#"> Current Status: <?php echo $_SESSION['status']; ?> </a>
                    <a href = "#"> Join Date: <?php echo $_SESSION['regdate']; ?> </a>
                    <a href = "../userprofile.php"> Go to account settings </a>
                </div>
            </li>
        </ul>
    </body>
</html>

```

```

</li>
</ul>

<div id = "addnewtopic">
  <h3> Add New Thread </h3>

  <form method = "POST" action = "Add_(Subject)_thread.php">
    <label> Author: <?php echo $_SESSION['username']; ?> </label>

    <br/><br/>

    <!-- Topics selections -->
    <label>
      Select Topic:
      <select name = "topics-(Subject)">
        <option name = "(Subject)-(topic1)"> (topic1) </option>
        <option name = "(Subject)-(topic2)"> (topic2) </option>
        <option name = "(Subject)-(topic3)"> (topic3) </option>
        ...
      </select>
    </label>

    <br/><br/>

    <label> Title: <input type = "text" name = "title" size = "100"> </label>

    <br/><br/>

    <label> Content: <br/><textarea name = "content" rows = "20" cols =
"70"></textarea> </label>

    <br/><br/>

    <input type = "submit" name = 'submit-info' value = "Submit Thread">
  </form>

  <?php
    include_once '../includes/ThreadHandler.php';

    if(isset($_POST['submit-info']))
    {
      $Directory = "(Subject) Threads";
      $uid = $_SESSION['userid'];
      $Author = $_SESSION['username'];
      $Subject = "(Subject)";
      $TopicSelection = "";
    }
  }

```

```

$Title = $_POST['title'];
$Content = $_POST['content'];

/* For the Topic selections */
if(isset($_POST['topics-(Subject)']))
{
    $Topic = $_POST['topics-(Subject)'];

    switch($Topic)
    {
        case '(topic1)': //if the user chooses '(topic1)' in the Topics selection
            $TopicSelection = "(topic1)";
            break;
        case '(topic2)': //else if the user chooses '(topic2)' in the Topics selection
            $TopicSelection = "(topic2)";
            break;
        case '(topic3)': //else if the user chooses '(topic3)' in the Topics selection
            $TopicSelection = "(topic3)";
            break;
        ...(continue with more cases if there are more topic selections)
        default: //else, display an error message if the user did not choose
            anything
            echo "<p style = 'color: red;'>You must choose one of the following
topic selections!</p>";
            break;
        }
    }

    $createFile = new ThreadHandler(); //object of the ThreadHandler class,
named 'createFile'
    $createFile -> createThread($Directory, $uid, $Author, $Subject,
$TopicSelection, $Title, $Content);
}
?>
</div>
</body>
</html>

```

When the user types in the topic thread information of a particular subject, when clicking the “Submit Thread” button, it will create the file, store it in the subject folder, then store the thread information in the threads table.

**6/14/19 – 6/15/19**

Above, you did it only for the Add\_Culture\_thread.php file. You want to do the above with the other ones.

What you want to do now is work on the displayThread() method. How this will work is it will loop through the threads table and display all the threads in the following order: “(Thread title) by (Username)” in the topic files of each subject folder (Art.php, Philosophy.php, etc). You have to link the “(Thread title) by (Username)” link with the topic thread file that matches the (Thread title). For example, if the thread title is: “Thread on Philosophy”, then you want to link that title with the following file: “DiscussionWindow\_Philosophy\_Thread on Philosophy.php”.

### **ThreadHandler.php**

```
...
/* This method is responsible for displaying the topic thread on the forumhome.php page and
the subject pages */
    public function displayThread($id, $directory , $subject, $topic)
    {
        $this -> sql2 = "SELECT user_id, thread_author, thread_subject, thread_topic,
thread_title
                        FROM threads
                        WHERE user_id='$id' AND thread_subject='$subject' AND
thread_topic='$topic';";

        $stmt2 = $this -> db -> stmt_init();

        if(!$stmt2 -> prepare($this -> sql2))
        {
            echo "<script> alert('SQL ERROR: ". $stmt2 -> error . "'); </script>";
        }
        else
        {
            $stmt2 -> execute();

            $result = $stmt2 -> get_result();

            if($result -> num_rows > 0)
            {
                while($row = $result -> fetch_assoc())
                {
                    echo "<tr>
                        <td><a href = '$directory/DiscussionWindow_" . $row['thread_topic'] . "_" .
$row['thread_title'] . ".php'" . $row['thread_title'] . "</a></td>
                        <td> Replies: <br/> Views: </td>
                        <td> By: <em>" . $row['thread_author'] . "</em> </td>
                    </tr>";
                }
            }
        }
    }
}
```

```

else
{
    echo "<i> There are no current threads posted. </i>";
}
}
}...

```

Now in each topic page in each subject folder (ex/ Art.php, Philosophy.php, Biology.php, etc), in the threadwindow of each topic page:

```

...
<div id = "threadwindow">
    <table class = "table">
        <thead>
            <tr>
                <th> Sub-Forums </th>
                <th> Threads/Posts </th>
                <th> Latest Posts</th>
            </tr>
        </thead>

        <tbody>
            <tr>
                <td> Sub-Forum 1 </td>
                <td> Threads: <br/> Posts: </td>
                <td> thread title of topic <br/> by <em>(username of user who created the
subject)</em> </td>
            </tr>
        </tbody>
    </table>

    <br/><br/><br/>

    <table class = "table">
        <thead>
            <tr>
                <th> Title/Threads </th>
                <th> Replies/Views </th>
                <th> Latest Posts </th>
            </tr>
        </thead>

        <tbody>
            <tr>
                <td> Thread title </td>
                <td> Replies: <br/> Views: </td>

```



```

        <td> thread title of topic </br> by <em>(username of user who created the
subject)</em> </td>
    </tr>
</tbody>
</table>
</div>
...

```

Add the following php code segment in the “*table class = ‘table’*”:

```

<div id = "threadwindow">
    <table class = "table">
        <thead>
            <tr>
                <th> Sub-Forums </th>
                <th> Threads/Posts </th>
                <th> Latest Posts</th>
            </tr>
        </thead>

        <tbody>
            <tr>
                <td> Sub-Forum 1 </td>
                <td> Threads: <br/> Posts: </td>
                <td> thread title of topic </br> by <em>(username of user who created the
subject)</em> </td>
            </tr>
        </tbody>
    </table>

    <br/><br/><br/>

    <table class = "table">
        <thead>
            <tr>
                <th> Title/Threads </th>
                <th> Replies/Views </th>
                <th> Latest Posts </th>
            </tr>
        </thead>

        <tbody>
            <tr>
                <?php
                include '../includes/ThreadHandler.php';

```

```

        $suid = $_SESSION['userid'];
        $Directory = "(Subject) Threads";
        $Subject = "(Subject)";
        $Topic = "(Topic)";

        $displayThreads = new ThreadHandler();
        $displayThreads -> displayThread($suid, $Directory, $Subject, $Topic);
    ?>
</tr>
</tbody>
</table>
</div>

```

Now each of the topic pages will display the threads of that particular thread.

What you now want to do is, in all the topics pages, delete all the sub forums windows, because they are absolutely useless.

---

## **6/16/19**

What you want to do now is, after users create their threads on a topic of a particular subject and display those threads on their respective pages, when other users click on that thread, you want to implement the code that will allow users to reply to that thread.

How this will work is, after the user has replied to the thread they replied on, that reply will be added to the replies table of the discussionforum database. Then, on the thread page, you'll want to find a way to display that reply on that thread page. Each time a user create a reply on that particular topic thread, you'll want to find a way to create some loop that will output all replies from the replies table each time one is created.

First, you'll want to revamp the comment field in the ThreadHandler.php:

### **ThreadHandler.php**

```

...
        <!-- Comment box -->
        <table id = 'commentfield'>
            <tr>
                <td>
                    <button id = 'bold'> <abbr title = 'Bold'><b>B</b></abbr>
</button><br/><br/>
                    <button id = 'italic'> <abbr title =
'Italic'><em>I</em></abbr> </button><br/><br/>
                    <button id = 'underline'> <abbr title =
'Underline'><u>U</u></abbr> </button><br/><br/>

```

```

        <button id = 'subscript'> <abbr title =
'Subscript'>A<sub>2</sub></abbr> </button><br/><br/>
        <button id = 'superscript'> <abbr title =
'Superscript'>A<sup>2</sup></abbr> </button><br/><br/>
        <button id = 'quote'> <abbr title = 'Quote'>Q</abbr>
</button>

        </td>
        <td>
        <form method = 'POST'>
        <textarea rows = '15' cols = '70' id = 'commentbox' name =
'commentbox'></textarea> <br/>
        <input type = 'submit' name = 'submit-comment' value =
'Submit comment!'/>
        </form>
        </td>
    </tr>
</table>
...

```

And in the DiscussionWindow.css:

```

...
/*styles the comment field*/
#commentfield
{
    background-color: grey;

    padding: 5% 5% 0 5%;
}
#commentfield tr abbr
{
    text-decoration: none;
}

#commentfield form
{
    text-align: center;
}
#commentfield form textarea
{
    border-radius: 5px;

    resize: none;
}
#commentfield form input[type=submit]
{

```

```

border: none;
border-radius: 5px;

padding: 5px;

cursor: pointer;

font-family: serif;
}
...

```

Before moving on with the comments bit, one thing you want to do is in the ThreadHandler.php, after the user has created a topic thread, you want to send that user to the topic page which will shows the threads created for that topic (including of course, the one the user created). To do this, implement the following code in the ThreadHandler.php page in the createThread() method:

### **ThreadHandler.php**

```

...
/* This method is responsible for obtaining the thread information and storing that information
in the threads table */
    public function createThread($directory, $id, $author, $subject, $topic, $title, $content,
$topicPage)
    {
        /* Before adding the thread information into the database, first append that information
into a file called 'DiscussionWindow_(topic)_(title).php' */
        $threadfile = fopen("$directory/DiscussionWindow_$topic" . "_" . $title.php", "w");
        $profilepic = "<?php echo " . " \<img src = '../..../". $_SESSION['pic'] . "' height='70'
width='70'/>\"; ?>";

        //set the thread page layout for the 'DiscussionWindow_(topic)_(title).php' page
        $threadlayout = "<?php
            session_start();
            ?>
            <html>
                <head>
                    <title> $title </title>
                    <meta charset = 'utf-8'/>

                    <link rel = 'stylesheet' href = '../..../ThreadsCssFiles/Navbar.css'/>
                    <link rel = 'stylesheet' href =
'../..../ThreadsCssFiles/DiscussionWindow.css'/>
                </head>

                <body id = '$subject'>
                    <!-- Navbar (shortened version because too hard to add the whole
navbar)-->

```

```

        <ul class = 'navbar'>
            <li class = 'left-tabs'> <a href = '../././forumhome.php'> Home </a>
</li>

            <li class = 'left-tabs'> <a href = '#'> Board </a> </li>
            <li class = 'left-tabs'> <a href = '#'> Community </a> </li>

            <li class = 'right-tabs'> <a href = '../././userprofile.php'> Go to
profile </a> </li>

            <li class = 'right-tabs'> <a href = '../././includes/logout.php'> Logout
</a> </li>

        </ul>

        <!-- Thread window -->
        <div id = 'thread'>
            <header>
                <h3> $title </h3>
                <h5> Date created: <?php echo " . "$" . "_SESSION" .
"['thread_creation'];" . " ?> </h5>
                <h5> $subject </h5>
                <h5> $topic </h5>
            </header>

            <section>
                <div class = 'userinfo'>
                    <p> <?php echo " . "$" . "_SESSION" . "['username']; ?>" .
"</p>

                    <p> $profilepic </p>
                    <p> Join date: <?php echo " . "$" . "_" .
"SESSION['regdate']; ?>" . "</p>

                    <p> <?php echo " . "$" . "_" . "SESSION" . "['status']; ?>" .
"</p>

                </div>

                <article>
                    $content
                </article>
            </section>

            <br/><br/><br/>

            <!-- Comment box -->
            <table id = 'commentfield'>
                <tr>
                    <td>
                        <button id = 'bold'> <abbr title = 'Bold'><b>B</b></abbr>
</button><br/><br/>

```

```

                                <button id = 'italic'> <abbr title =
'Italic'><em>I</em></abbr> </button><br/><br/>
                                <button id = 'underline'> <abbr title =
'Underline'><u>U</u></abbr> </button><br/><br/>
                                <button id = 'subscript'> <abbr title =
'Subscript'>A<sub>2</sub></abbr> </button><br/><br/>
                                <button id = 'superscript'> <abbr title =
'Superscript'>A<sup>2</sup></abbr> </button><br/><br/>
                                <button id = 'quote'> <abbr title = 'Quote'>Q</abbr>
</button>
                                </td>
                                <td>
                                    <form method = 'POST'>
                                        <textarea rows = '15' cols = '70' id = 'commentbox' name =
'commentbox'></textarea> <br/>
                                        <input type = 'submit' name = 'submit-comment' value =
'Submit comment!'/>
                                    </form>
                                </td>
                            </tr>
                        </table>
                    </div>
                </body>
            </html>";

```

```

fwrite($threadfile, $threadlayout);

```

```

/* Now add the thread information to the database */
$this -> sql1 = "INSERT INTO threads(user_id, thread_author, thread_subject,
thread_topic, thread_title, thread_content)
                VALUES('$id', '$author', '$subject', '$topic', '$title', '$content')";

```

```

$stmt1 = $this -> db -> stmt_init();

```

```

if(!$stmt1 -> prepare($this -> sql1))
{
    echo "<script> alert('SQL ERROR: ". $stmt1 -> error . "'); </script>";
}
else
{
    $stmt1 -> execute();
    header("Location: $topicPage?threadcreation=success");
}
}
}

```

```

...

```

Then in the Add\_(Subject)\_thread.php page, add the following:

**Add (Subject) thread.php**

```
<?php
    session_start();
?>
<html>
    <head>
        <title> New Topic --- (Subject) </title>
        <meta charset = "utf-8"/>

        <link rel = "stylesheet" href = "../ThreadsCssFiles/Navbar.css"/>
        <link rel = "stylesheet" href = "../ThreadsCssFiles/addWindow.css"/>
    </head>

    <body>
        <!-- navbar -->
        <ul class = "navbar">
            <li class = "left-tabs"><a href = "../forumhome.php"> Home </a></li>
            <li class = "left-tabs"><a href = "#"> Board </a></li>
            <li class = "left-tabs"><a href = "#"> Community </a></li>

            <li class = "right-tabs"><a href = "../includes/logout.php"> Logout </a></li>
            <li class = "dropdown right-tabs">
                <a href = "#" class = "dropdownButton"> Hello: <?php echo
$_SESSION['username']; ?> </a>
                <div class = "dropdown-content">
                    <a href = "#"> <?php echo "<img src = '../.." . $_SESSION['pic'] . "' height='70'
width='70'/>"; ?> </a>
                    <a href = "#"> Full Name: <?php echo $_SESSION['firstname'] . " " .
$_SESSION['lastname']; ?> </a>
                    <a href = "#"> Email: <?php echo $_SESSION['useremail']; ?> </a>
                    <a href = "#"> Current Status: <?php echo $_SESSION['status']; ?> </a>
                    <a href = "#"> Join Date: <?php echo $_SESSION['regdate']; ?> </a>
                    <a href = "../userprofile.php"> Go to account settings </a>
                </div>
            </li>
        </ul>

        <div id = "addnewtopic">
            <h3> Add New Thread </h3>

            <form method = "POST" action = "Add_(Subject)_thread.php">
                <label> Author: <?php echo $_SESSION['username']; ?> </label>

                <br/><br/>
```

```

<!-- Topics selections -->
<label>
    Select Topic:
    <select name = "topics-(Subject)">
        <option name = "(Subject)-(topic1)"> (topic1) </option>
        <option name = "(Subject)-(topic2)"> (topic2) </option>
        <option name = "(Subject)-(topic3)"> (topic3) </option>
        ...
    </select>
</label>

<br/><br/>

<label> Title: <input type = "text" name = "title" size = "100"> </label>

<br/><br/>

<label> Content: <br/><textarea name = "content" rows = "20" cols =
"70"></textarea> </label>

<br/><br/>

<input type = "submit" name = 'submit-info' value = "Submit Thread">
</form>

<?php
include_once '../includes/ThreadHandler.php';

if(isset($_POST['submit-info']))
{
    $Directory = "(Subject) Threads";
    $uid = $_SESSION['userid'];
    $Author = $_SESSION['username'];
    $Subject = "(Subject)";
    $TopicSelection = "";
    $Title = $_POST['title'];
    $Content = $_POST['content'];
    $TopicPage = "";

    /* For the Topic selections */
    if(isset($_POST['topics-(Subject)']))
    {
        $Topic = $_POST['topics-(Subject)'];

        switch($Topic)
        {

```



```

        case '(topic1)': //if the user chooses '(topic1)' in the Topics selection
            $TopicSelection = "(topic1)";
            $TopicPage = "(topic1).php";
            break;
        case '(topic2)': //else if the user chooses '(topic2)' in the Topics selection
            $TopicSelection = "(topic2)";
            $TopicPage = "(topic2).php";
            break;
        case '(topic3)': //else if the user chooses '(topic3)' in the Topics selection
            $TopicSelection = "(topic3)";
            $TopicPage = "(topic3).php";
            break;
        ...(continue with more cases if there are more topic selections)
        default: //else, display an error message if the user did not choose anything
            echo "<p style = 'color: red;'>You must choose one of the following topic
selections!</p>";
            break;
    }
}

    $createFile = new ThreadHandler(); //object of the ThreadHandler class, named
'createFile'
    $createFile -> createThread($Directory, $uid, $Author, $Subject,
$TopicSelection, $Title, $Content, $TopicPage);
}
?>
</div>
</body>
</html>

```

Now when the user finishes creating their thread on a particular topic, they will be sent to the topic page showing the thread they created along with others.

What you also want to do now is delete the user id session variable on the displayThread() method of the ThreadHandler.php page as well as the topics pages that link the file because it will only show the threads associated with user logged in.

### **ThreadHandler.php**

```

...
/* This method is responsible for displaying the topic thread on the forumhome.php page and
the subject pages */
    public function displayThread($directory , $subject, $topic)
    {
        $this -> sql2 = "SELECT thread_author, thread_subject, thread_topic, thread_title
                        FROM threads
                        WHERE thread_subject='$subject' AND thread_topic='$topic';";
    }
}

```

...

Remove the \$id from the parameters and user\_id in the SQL statement.

Now in each topics page:

```
<?php
    include '../includes/ThreadHandler.php';

    $Directory = "Culture Threads";
    $Subject = "Culture";
    $Topic = "Art";

    $displayThreads = new ThreadHandler();
    $displayThreads -> displayThread($Directory, $Subject, $Topic);
?>
```

This way, the page will now display threads created by all users, not just one user.

---

Next thing is for the thread table, create a new table called “thread\_file”. This will contain the directory of the thread files:

```
ALTER TABLE threads
ADD thread_file varchar(255);
```

In the ThreadHandler.php, add the following:

```
ThreadHandler.php
<?php
    include_once 'database.php';

    /* This class will handle the processes of both creating the topic thread of a particular
    subject
    and displaying it on the forumhome.php and the subject pages */
    class ThreadHandler
    {
        private $db;

        //SQL variables
        private $sql1; //for the createThread() method. For storing the thread information in the
        thread table.
        private $sql2; //for the displayThread() method. For obtaining the thread information from
        the thread table and displaying it in their respective topic thread file
        private $sql3; //for the createReply() method. For obtaining the user comment and storing
        it in the replies table
```

```

public function __construct()
{
    $this->db = new Connection();
    $this->db = $this->db->connect();
}

/* This method is responsible for obtaining the thread information and storing that
information in the threads table */
public function createThread($directory, $id, $author, $subject, $topic, $title, $content,
$topicPage)
{
    /* Before adding the thread information into the database, first append that information
into a file called 'DiscussionWindow_(topic)_(title).php' */
    $threadfile = fopen("$directory/DiscussionWindow_$topic" . "_" . $title.php", "w");
    $profilepic = "<?php echo " . "\"<img src = '../..../". $_SESSION['pic'] . "' height='70'
width='70'/>\"; ?>";

    $threadPath = "$directory/DiscussionWindow_$topic" . "_" . $title.php";

    //set the thread page layout for the 'DiscussionWindow_(topic)_(title).php' page
    $threadlayout = "<?php
        session_start();
        ?>
        <html>
        <head>
            <title> $title </title>
            <meta charset = 'utf-8'/>

            <link rel = 'stylesheet' href = '../..../ThreadsCssFiles/Navbar.css'/>
            <link rel = 'stylesheet' href =
'../..../ThreadsCssFiles/DiscussionWindow.css'/>
        </head>

        <body id = '$subject'>
            <!-- Navbar (shortened version because too hard to add the whole
navbar)-->

            <ul class = 'navbar'>
                <li class = 'left-tabs'> <a href = '../..../forumhome.php'> Home </a>
</li>

                <li class = 'left-tabs'> <a href = '#'> Board </a> </li>
                <li class = 'left-tabs'> <a href = '#'> Community </a> </li>

                <li class = 'right-tabs'> <a href = '../..../userprofile.php'> Go to
profile </a> </li>

```

```

        <li class = 'right-tabs'> <a href = '../../includes/logout.php'> Logout
    </a> </li>

</ul>

<!-- Thread window -->
<div id = 'thread'>
    <header>
        <h3> $title </h3>
        <h5> Date created: <?php echo " . "$" . "_SESSION" .
"['thread_creation'];" . " ?> </h5>
        <h5> $subject </h5>
        <h5> $topic </h5>
    </header>

    <section>
        <div class = 'userinfo'>
            <p> <?php echo " . "$" . "_SESSION" . "['username']; ?>" .
" </p>

            <p> $profilepic </p>
            <p> Join date: <?php echo " . "$" . "_" .
"SESSION['regdate']; ?>" . " </p>
            <p> <?php echo " . "$" . "_" . "SESSION" . "['status']; ?>" .
" </p>

        </div>

        <article>
            $content
        </article>
    </section>

    <br/><br/><br/>

    <!-- Comment box -->
    <table id = 'commentfield'>
        <tr>
            <td>
                <button id = 'bold'> <abbr title = 'Bold'><b>B</b></abbr>
</button><br/><br/>

                <button id = 'italic'> <abbr title =
'Italic'><em>I</em></abbr> </button><br/><br/>

                <button id = 'underline'> <abbr title =
'Underline'><u>U</u></abbr> </button><br/><br/>

                <button id = 'subscript'> <abbr title =
'Subscript'>A<sub>2</sub></abbr> </button><br/><br/>

                <button id = 'superscript'> <abbr title =
'Superscript'>A<sup>2</sup></abbr> </button><br/><br/>

```

```

        <button id = 'quote'> <abbr title = 'Quote'>Q</abbr>
</button>

        </td>
        <td>
            <form method = 'POST'>
                <textarea rows = '15' cols = '70' id = 'commentbox' name =
'commentbox'></textarea> <br/>
                <input type = 'submit' name = 'submit-comment' value =
'Submit comment!'/>
            </form>
        </td>
    </tr>
</table>
</div>
</body>
</html>";

    fwrite($threadfile, $threadlayout);

    /* Now add the thread information to the database */
    $this -> sql1 = "INSERT INTO threads(user_id, thread_author, thread_subject,
thread_topic, thread_title, thread_content, thread_file)
        VALUES('$id', '$author', '$subject', '$topic', '$title', '$content',
'$threadPath');";

    $stmt1 = $this -> db -> stmt_init();

    if(!$stmt1 -> prepare($this -> sql1))
    {
        echo "<script> alert('SQL ERROR: ". $stmt1 -> error . "'); </script>";
    }
    else
    {
        $stmt1 -> execute();
        header("Location: $topicPage?threadcreation=success");
    }
}

/* This method is responsible for displaying the topic thread on the forumhome.php page
and the subject pages */
public function displayThread($subject, $topic)
{
    $this -> sql2 = "SELECT thread_author, thread_subject, thread_topic, thread_title,
thread_file
        FROM threads

```

```

WHERE thread_subject='$subject' AND thread_topic='$topic';";

$stmt2 = $this->db->stmt_init();

if(!$stmt2->prepare($this->sql2))
{
    echo "<script> alert('SQL ERROR: ". $stmt2->error . "'); </script>";
}
else
{
    $stmt2->execute();

    $result = $stmt2->get_result();

    if($result->num_rows > 0)
    {
        while($row = $result->fetch_assoc())
        {
            echo "<tr>
                <td><a href = '". $row['thread_file'] .' ">". $row['thread_title'] . "</a></td>
                <td> Replies: <br/> Views: </td>
                <td> By: <em>" . $row['thread_author'] . "</em> </td>
            </tr>";
        }
    }
    else
    {
        echo "<i> There are no current threads posted. </i>";
    }
}
}
?>

```

Then in each topic page, add the following:

```
<?php
    include '../includes/ThreadHandler.php';

    $Subject = "(Subject)";
    $Topic = "(Topic)";

    $displayThreads = new ThreadHandler();
    $displayThreads -> displayThread($Subject, $Topic);
?>
```

This makes it easier to link to the topic thread.

---

### 6/17/19, 6/21/19

Now you want to work on the replies bit.

On the topic thread page, you'll want to find a way to display that reply on that thread page. Each time a user create a reply on that particular topic thread, you'll want to find a way to create some loop that will output all replies from the replies table each time one is created. But the replies must be for that particular topic thread.

Hint: Since you created the thread\_file column in the threads table, you can use that write to that topic thread file (that is, writing the reply to that topic thread) using the fwrite() function.

What you want to do now is, revamp the whole replies table with the following columns:

- **user\_id** (references the id of the user who is replying to a thread)
- **user\_username** (references the username who is replying to the thread)
- **user\_status** (references the status of the user who is replying to the thread)
- **profile\_pic** (the profile picture of the user who is replying to the thread)
- **thread\_title** (references the thread\_title of the thread table, the title of the thread you are replying)
- **thread\_file** (references the thread\_file of the thread table the user is replying to)
- **replies\_id** (the id of the reply)
- **replies\_content** (what the reply is)

Therefore, delete the old replies table and instead create a new one using the following SQL code:

```
CREATE TABLE replies
(
    user_id int NOT NULL,
    user_username varchar(100) NOT NULL, /* name of the user replying to a thread */
    user_status varchar(100),
    profile_pic varchar(255),
    thread_title varchar(100) NOT NULL, /* title of the thread being replied to */
    thread_file varchar(255) NOT NULL,
    replies_id int NOT NULL AUTO_INCREMENT,
    replies_creation varchar(100) NOT NULL, /* when the reply was created */
    replies_content text NOT NULL,
    PRIMARY KEY(replies_id),
    FOREIGN KEY(user_id) REFERENCES accounts(user_id),
    FOREIGN KEY(thread_file) REFERENCES threads(thread_file)
);
```

But in order to make the above work, you'll have to delete the old accounts and threads table and input the following SQL code (separately):

```
CREATE TABLE accounts
(
    user_id int(11) NOT NULL AUTO_INCREMENT,
    reg_date varchar(100) NOT NULL, /* registration date when user first registered */
    user_firstname varchar(100) NOT NULL,
    user_lastname varchar(100) NOT NULL,
    user_email varchar(100) NOT NULL,
    user_username varchar(100) NOT NULL UNIQUE,
    user_password varchar(100) NOT NULL,
    user_status varchar(255) UNIQUE,
    profile_pic varchar(255) UNIQUE, /* stores the path of the uploaded profile pic */
    PRIMARY KEY(user_id)
);
```

```
CREATE TABLE threads
(
    user_id int NOT NULL,
    thread_id int NOT NULL AUTO_INCREMENT,
    thread_author varchar(100) NOT NULL,
    thread_subject varchar(100) NOT NULL,
    thread_topic varchar(100) NOT NULL,
    thread_title varchar(100) NOT NULL UNIQUE,
    thread_content text NOT NULL,
    thread_file varchar(255) UNIQUE, /* the filename of the thread */
    thread_creation varchar(255), /* When the thread was created */
    PRIMARY KEY(thread_id),
```



```
FOREIGN KEY(user_id) REFERENCES accounts(user_id)
);
```

Remember, foreign keys must reference the primary or unique keys of the parent class. The parent class being threads and the child class being replies.

Now you should have the following columns in the newly revamped accounts, threads, and replies table:

- accounts:

user_id	reg_date	user_firstname	user_lastname	user_email	user_username	user_password	user_status	profile_pic
---------	----------	----------------	---------------	------------	---------------	---------------	-------------	-------------

- threads:

user_id	thread_id	thread_author	thread_subject	thread_topic	thread_title	thread_content	thread_file	thread_creation
---------	-----------	---------------	----------------	--------------	--------------	----------------	-------------	-----------------

- replies:

user_id	user_username	user_status	profile_pic	thread_title	thread_file	replies_id	replies_content
---------	---------------	-------------	-------------	--------------	-------------	------------	-----------------

Also, in the AccountHandler.php, because you added the line “reg\_date varchar(100) NOT NULL” which is the registration date of the user, in the signup method of the Accounts class, add the following:

#### **AccountHandler.php**

```
...
/* This method is for registering new users and adding their information to the accounts table
*/
    public function signup($accountCreated, $firstname, $lastname, $email, $username,
$password)
    {
        if(empty($firstname) && empty($lastname) && empty($email) && empty($username)
&& empty($password))
        {
            echo "<p id = 'errormessage'> Do not submit empty information! </p>";
        }
        else if(empty($firstname) || empty($lastname) || empty($email) || empty($username) ||
empty($password))
        {
            echo "<p id = 'errormessage'> You must fill out everything! </p>";
        }
        else
        {
            $this->sql2 = "INSERT INTO accounts(reg_date, user_firstname, user_lastname,
user_email, user_username, user_password)
                        VALUES(?, ?, ?, ?, ?, ?);";

            $stmt = $this->dbConn->stmt_init();
```

```

        if(!$stmt -> prepare($this -> sql2))
        {
            echo "<script> alert('SQL ERROR: " . $stmt -> error . "'); </script>";
        }
        else
        {
            $hashedPwd = password_hash($password, PASSWORD_DEFAULT);

            $stmt -> bind_param("sssss", $accountCreated, $firstname, $lastname, $email,
$username, $hashedPwd);
            $stmt -> execute();

            echo "<p id = 'successmessage'> You have successfully signed up! Sign in now to
enter forum discussion! </p>";
        }
    }
}
...

```

Then in the signupPage.php:

### signupPage.php

```

...
<div id = "signupwindow">
    <h3> Member Signup </h3>

    <form method = "POST" action = "signupPage.php">
        <input type = "text" placeholder = "Enter First Name" name = "fname"/>
        <br/>
        <input type = "text" placeholder = "Enter Last Name" name = "lname"/>
        <br/>
        <input type = "text" placeholder = "Enter your Email" name = "email"/>
        <br/>
        <input type = "text" placeholder = "Enter Desired Username" name = "uname"/>
        <br/>
        <input type = "password" placeholder = "Enter Desired Password" name =
"pwd"/>
        <br/>
        <input type = "submit" value = "Register" name = "submitButton"/>
    </form>

    <?php
    if(isset($_POST['submitButton']))
    {
        $accountCreatedDate = date("m/d/Y"); //date of when the user first created
his/her account
    }
}

```

```

        $fname = $_POST['fname'];
        $lname = $_POST['lname'];
        $email = $_POST['email'];
        $uname = $_POST['uname'];
        $pwd = $_POST['pwd'];

        $account -> signup($accountCreatedDate, $fname, $lname, $email, $uname,
$pwd);
    }
?>

    <a href = 'loginPage.php'> <h5> Already have an account, log in now </h5> </a>
</div>
...

```

---

**6/22/19**

Now that all that is done, going back to the replies bit, there will be two methods in the ThreadHandler that will handle the reply creation process:

1. **createReply()**: Obtains the reply from the user on a particular thread and store that comment in the replies table.
2. **displayReply()**: Obtains the user reply on a particular thread from the replies table and display all the user replies on that particular thread.

#### **ThreadHandler.php**

```

<?php
    include_once 'database.php';

    /* This class will handle the processes of both creating the topic thread of a particular
subject
and displaying it on the forumhome.php and the subject pages */
    class ThreadHandler
    {
        private $db;

        //SQL variables
        private $sql1; //for the createThread() method. For storing the thread information in the
thread table.
        private $sql2; //for the displayThread() method. For obtaining the thread information from
the thread table and displaying it in their respective topic thread file
        private $sql3; //for the createReply() method. For obtaining the user comment and
storing it in the replies table
        private $sql4; //for the displayReply() method. For obtaining the user comment from
the replies table and displaying it in the respective topic thread

```

```

public function __construct()
{
    $this->db = new Connection();
    $this->db = $this->db->connect();
}

/* This method is responsible for obtaining the thread information and storing that
information in the threads table */
public function createThread($directory, $id, $author, $subject, $topic, $title, $content,
$topicPage)
{
    /* Before adding the thread information into the database, first append that
information into a file called 'DiscussionWindow_(topic)_(title).php' */
    $threadfile = fopen("$directory/DiscussionWindow_$topic" . "_" . " $title.php",
"w");

    $uregdate = $_SESSION['regdate']; //registration date of user who made the
thread
    $upic = "<img src = '../..'/' . $_SESSION['pic'] .'" height='70' width='70'/>";

    $threadPath = "$directory/DiscussionWindow_$topic" . "_" . " $title.php"; //full
thread file path
    $regularPath = "DiscussionWindow_$topic" . "_" . " $title.php"; //half thread file
path
    $threadCreated = date("m/d/Y");

    //set the thread page layout for the 'DiscussionWindow_(topic)_(title).php' page
    $threadlayout = "<?php
        session_start();
        include '../..../includes/ThreadHandler.php';
    ?>
    <html>
        <head>
            <title> $title </title>
            <meta charset = 'utf-8'/>

            <link rel = 'stylesheet' href = '../..../ThreadsCssFiles/Navbar.css'/>
            <link rel = 'stylesheet' href =
'../..../ThreadsCssFiles/DiscussionWindow.css'/>
        </head>

        <body id = '$subject'>
            <!-- Navbar (shortened version because too hard to add the whole
navbar)-->
            <ul class = 'navbar'>

```

```

</li>
<li class = 'left-tabs'> <a href = '../../forumhome.php'> Home </a>

<li class = 'left-tabs'> <a href = '#'> Board </a> </li>
<li class = 'left-tabs'> <a href = '#'> Community </a> </li>

<li class = 'right-tabs'> <a href = '../../userprofile.php'> Go to
profile </a> </li>

<li class = 'right-tabs'> <a href = '../../includes/logout.php'> Logout
</a> </li>

</ul>

<!-- Thread window -->
<div id = 'thread'>
  <header>
    <h3> $title </h3>
    <h5> Date created: $threadCreated </h5>
    <h5> $subject </h5>
    <h5> $topic </h5>
  </header>

  <section>
    <div class = 'userinfo'>
      <p> <?php echo ".$$_SESSION"."['username'];"."
?> </p>

      <p> <?php echo "\"$upic\""; ?> </p>
      <p> Join date: $uregdate </p>
      <p> Current Status: <?php echo
".$$_SESSION"."['status'];"." ?> </p>
    </div>

    <article>
      $content
    </article>
  </section>

  <br/><br/><br/>

  <!-- Replies window -->
  <?php
    ".$$_SESSION"."replies = new ThreadHandler();

    ".$$_SESSION"."Directory = \"$threadPath\";

    ".$$_SESSION"."replies -> displayReply(".$$_SESSION"."Directory);
  ?>

```

```

<br/><br/><br/>

<!-- Comment box -->
<form method = 'POST' action = '$regularPath'>
  <table id = 'commentfield'>
    <tr>
      <td>
        <button id = 'bold'> <abbr title =
'Bold'><b>B</b></abbr> </button><br/><br/>
        <button id = 'italic'> <abbr title =
'Italic'><em>I</em></abbr> </button><br/><br/>
        <button id = 'underline'> <abbr title =
'Underline'><u>U</u></abbr> </button><br/><br/>
        <button id = 'subscript'> <abbr title =
'Subscript'>A<sub>2</sub></abbr> </button><br/><br/>
        <button id = 'superscript'> <abbr title =
'Superscript'>A<sup>2</sup></abbr> </button><br/><br/>
        <button id = 'quote'> <abbr title = 'Quote'>Q</abbr>
</button>
      </td>
      <td>
        <textarea rows = '15' cols = '70' id = 'commentbox'
name = 'commentbox'></textarea> <br/>
        <input type = 'submit' name = 'submit-comment' value
= 'Submit comment!'/>
      </td>
    </tr>
  </table>
</form>

<?php
if(isset($_POST['submit-comment']))
{
  $_SESSION['Replies'] = new ThreadHandler();

  $_SESSION['uid'] = $_SESSION['userid']; //id of the
user who made the comment/reply
  $_SESSION['uname'] = $_SESSION['username'];
//username of the user who made the comment/reply
  $_SESSION['upic'] = $_SESSION['pic']; //profile pic
of user who made the comment/reply
  $_SESSION['Title'] = $_POST['title']; //title of the thread who the
user is replying to
  $_SESSION['Directory'] = $_POST['threadPath']; //full file path
of the thread file
  $_SESSION['replyCreated'] = date('m/d/Y'); //date the
reply was created
  $_SESSION['Reply'] = $_POST['commentbox'];
//the reply

```

```

        ". "$" ."Replies -> createReply(".$".uid,
        ".$".uname, ".$".upic, ".$".Title, ".$".Directory, ".$".replyCreated,
        ".$".Reply);

```

```

    }
    ?>
</td>
</tr>
</table>
</form>
</div>
</body>
</html>";

```

```

fwrite($threadfile, $threadlayout);

```

```

/* Now add the thread information to the database */
$this -> sql1 = "INSERT INTO threads(user_id, thread_author, thread_subject,
thread_topic, thread_title, thread_content, thread_file)
VALUES('$id', '$author', '$subject', '$topic', '$title', '$content',
'$threadPath')";

```

```

$stmt1 = $this -> db -> stmt_init();

```

```

if(!$stmt1 -> prepare($this -> sql1))
{
    echo "<script> alert('SQL ERROR: ". $stmt1 -> error . "'); </script>";
}
else
{
    $stmt1 -> execute();
    header("Location: $topicPage?threadcreation=success");
}
}

```

```

/* This method is responsible for displaying the topic thread on the forumhome.php page
and the subject pages */

```

```

public function displayThread($subject, $topic)
{
    $this -> sql2 = "SELECT thread_author, thread_subject, thread_topic, thread_title,
thread_file
FROM threads
WHERE thread_subject='$subject' AND thread_topic='$topic'";

    $stmt2 = $this -> db -> stmt_init();

```

```

if(!$stmt2 -> prepare($this -> sql2))
{
    echo "<script> alert('SQL ERROR: ". $stmt2 -> error . "'); </script>";
}
else
{
    $stmt2 -> execute();

    $result = $stmt2 -> get_result();

    if($result -> num_rows > 0)
    {
        while($row = $result -> fetch_assoc())
        {
            echo "<tr>
                <td><a href = '". $row['thread_file'] ."'>" . $row['thread_title'] . "</a></td>
                <td> Replies: <br/> Views: </td>
                <td> By: <em>" . $row['thread_author'] . "</em> </td>
            </tr>";
        }
    }
    else
    {
        echo "<i style = 'color: red; background-color: grey;'> There are no current threads
        posted. </i>";
    }
}
}

/* This method creates the comment a user made on a particular topic thread */
public function createReply($id, $username, $pic, $title, $directory, $dateCreated,
$comment)
{
    /* Store the comment in the database with the respective thread file */
    if(empty($comment))
    {
        echo "<p id = 'errorMessage'> You must enter a comment! </p>";
    }
    else
    {
        $this -> sql3 = "INSERT INTO replies(user_id, user_username, profile_pic,
thread_title, thread_file, replies_creation, replies_content)
            VALUES('$id', '$username', '$pic', '$title', '$directory',
'$dateCreated', '$comment')";

        $stmt3 = $this -> db -> stmt_init();
    }
}

```



```

        if(!$stmt3 -> prepare($this -> sql3))
        {
            echo "<script> alert('SQL ERROR: ". $stmt3 -> error . "'); </script>";
        }
        else
        {
            $stmt3 -> execute();
            echo "<script> alert('Reply created!'); </script>";
        }
    }
}

/* This method displays the comments that users made on a particular topic thread
*/
public function displayReply($directory)
{
    /* Get only the replies of that particular topic thread */
    $this -> sql4 = "SELECT user_username, user_status, profile_pic, thread_file,
replies_creation, replies_content
                    FROM replies
                    WHERE thread_file = '$directory'";

    $stmt4 = $this -> db -> stmt_init();

    if(!$stmt4 -> prepare($this -> sql4))
    {
        echo "<script> alert('SQL ERROR: ". $stmt4 -> error . "'); </script>";
    }
    else
    {
        $stmt4 -> execute();

        $result = $stmt4 -> get_result();

        if($result -> num_rows > 0)
        {
            while($row = $result -> fetch_assoc())
            {
                echo "<div id = 'replies'>";
                echo "    <section>
                        <div class = 'userInfo'>";
                echo        "<img src = '" . $row['profile_pic'] . "' height='70'
width='70' /> <br/>";
                echo        "<p>". $row['user_username'] . "</p>";
                echo        "<p> Reply created: ". $row['replies_creation'] . "</p>";
            }
        }
    }
}

```

```

        echo "        "<p> Current Status: ". $row['user_status'] ." </p>";
        echo "        </div>

                <article>";
        echo "        "<p>". $row['replies_content'] ."</p>";
        echo "        "</article>";
        echo "    </section>
        </div>";
    }
}
else
{
    echo "<i style = 'color: red; background-color: grey;'> There are no current
replies posted. </i>";
}
}
}
}
?>

```

If users try to update their profile picture or status, that updated information will have to show up in the thread pages themselves whenever users visit or revisit them, so the previous SQL query will not work and will result in an error the next time a user tries to create a reply to a thread. So in order to prevent this, if user updates information in the accounts table, then update the replies table as well (since the user\_username, user\_status, and profile\_pic columns are child columns of the ones in the accounts table). You also want to update the thread\_author column in the threads table (which relates to the user\_username columns of the accounts and replies table). So in the ProfileHandler.php file, modify the following:

```

ProfileHandler.php
<?php
    include_once 'database.php';

    /* This class updates user information such as name, email and profile picture */
    class ProfileHandler
    {
        private $db; //to connect to the discussionforum database

        /* These variables are reserved for the uploadPics() method */
        private $sql1; //for uploading profile pictures
        private $photo = ""; //obtains the photo to be uploaded (initialize to nothing until user
selects a picture to upload)
        private $imageFileType = ""; //obtain the type of file (".jpg", ".png", etc...) (initialize to
nothing until user selects a picture to upload)
    }
}

```

```

private $folder; //the folder to store the uploaded pictures
private $check; //checks the size of the image

/* These variables are reserved for the updateStatus() method */
private $sql2; //for updating user information
private $newUsername = ""; //sets new user name
private $newPassword = ""; //sets new user password
private $newEmail = ""; //sets new user email
private $setStatus = ""; //sets new user status

/* Initialize connection to the discussionforum database upon creating the object of this
class */
public function __construct()
{
    $this -> db = new Connection();
    $this -> db = $this -> db -> connect();
}

/* This method uploads new profile pictures */
/* This will update the profile_pic column in both the accounts and replies table */
public function uploadPics($id)
{
    $this -> folder = "ProfilePics/";
    $this -> photo = $this -> folder . basename($_FILES["getFile"]["name"]); //obtains the
photo to be updated and store them in the ProfilePics folder
    $this -> imageFileType = strtolower(pathinfo($this -> photo,
PATHINFO_EXTENSION)); //obtain the path of the type of file (.jpg, .png, etc...)
    $this -> check = getimagesize($_FILES["getFile"]["tmp_name"]); //checks the size of
the image

    //check if the file is a real image or not
    if($this -> check !== false)
    {
        //if the image file is not jpg, jpeg, png and gif files
        if($this -> imageFileType != "jpg" && $this -> imageFileType != "png" && $this ->
imageFileType != "jpeg" && $this -> imageFileType != "gif")
        {
            echo "<p id = 'errMessage'> Sorry, only JPG, JPEG, PNG, and GIF files are allowed!
</p>";
        }
        else //else if the image file is a jpg, jpeg, png, and/or gif
        {
            //if the file is too large
            if($_FILES["getFile"]["size"] > 500000)
            {
                echo "<p id = 'errMessage'> Sorry, your file is too big to upload! </p>";
            }
        }
    }
}

```

```

    }
    else //else if file is not too large
    {
        $pic = $this -> photo; //this variable is just a placeholder to hold the photo variable

        //upload the image
        if(move_uploaded_file($_FILES["getFile"]["tmp_name"], $this -> photo))
        {
            /* update the profile_pic columns in both the accounts and replies table */
            $this -> sql1 = "UPDATE accounts SET profile_pic='$pic' WHERE
user_id='$id'";
            $this -> sql1 .= "UPDATE replies SET profile_pic='$pic' WHERE
user_id='$id'";

            if(!$this -> db -> multi_query($this -> sql1))
            {
                echo "<script> alert('Multi Query failed: " . $this -> db -> error . "');
</script>";
            }
            else
            {
                echo "<p id = 'succMessage'> Your picture was successfully uploaded!
<br/>
                In order to view your new picture, you will have to logout, then log
back in!
                </p>";
            }
        }
        //if the image could not be uploaded
        else
        {
            echo "<p id = 'errMessage'> Sorry, there was a problem uploading your image
file </p>";
        }
    }
}
}
else
{
    echo "<p id = 'errMessage'> The image you selected is not a file! </p>";
}
}

```

```

/* This method updates user information such as name, email, password, etc */
public function updateInfo($userid, $newUsername, $newPassword, $newEmail,
    $setStatus)
{
    //if the username field is not empty, but the others are, change only the username
    if(empty($newPassword) && empty($newEmail) && empty($setStatus))
    {
        $this->newUsername = $newUsername; //private $newUsername =
updateInfo($newUsername)

        $sid = intval($userid);

        $this->sql2 = "UPDATE accounts SET user_username='$this->newUsername'
WHERE user_id='$sid'";
        $this->sql2 .= "UPDATE replies SET user_username='$this->newUsername'
WHERE user_id='$sid'";
        $this->sql2 .= "UPDATE threads SET thread_author='$this->newUsername'
WHERE user_id='$sid'";

        if(!$this->db->multi_query($this->sql2))
        {
            echo "<script> alert('Multi Query failed! : " . $this->db->error . "');
</script>";
        }
        else
        {
            echo "<p id = 'succMessage'> Username has been changed! </p>";
        }
    }
    //if the password field is not empty, but the others are, change only the password
    else if(empty($newUsername) && empty($newEmail) && empty($setStatus))
    {
        $this->newPassword = $newPassword; //private $newPassword =
updateInfo($newPassword)
        $hashedPwd = password_hash($this->newPassword, PASSWORD_DEFAULT);
//must hash the password to prevent hackers from seeing user password

        $sid = intval($userid);

        $this->sql2 = "UPDATE accounts SET user_password='$hashedPwd' WHERE
user_id='$sid'";

        if(!$this->db->query($this->sql2))
        {
            echo "<script> alert('SQL ERROR: " . $this->db->error . "'); </script>";
        }
    }
}

```

```

else
{
    echo "<p id = 'succMessage'> Password has been changed! </p>";
}
}
//if the email field is not empty, but the others are, change only the email
else if(empty($newUsername) && empty($newPassword) && empty($setStatus))
{
    $this->newEmail = $newEmail; //private $newEmail = updateInfo($newEmail)

    $id = intval($userid);

    $this->sql2 = "UPDATE accounts SET user_email='$this->newEmail' WHERE
user_id='$id'";

    if(!$this->db->query($this->sql2))
    {
        echo "<script> alert('SQL ERROR: " . $this->db->error . "'); </script>";
    }
    else
    {
        echo "<p id = 'succMessage'> Email has been changed! </p>";
    }
}
//if the status is not empty, but the others are, change only the status
else if(empty($newUsername) && empty($newPassword) && empty($newEmail))
{
    $this->setStatus = $setStatus; //private $setStatus = updateInfo($setStatus)

    $id = intval($userid);

    $this->sql2 = "UPDATE accounts SET user_status='$this->setStatus' WHERE
user_id='$id'";
    $this->sql2 .= "UPDATE replies SET user_status='$this->setStatus' WHERE
user_id='$id'";

    if(!$this->db->multi_query($this->sql2))
    {
        echo "<script> alert('Multi Query failed!: " . $this->db->error . "');
</script>";
    }
    else
    {
        echo "<p id = 'succMessage'> Status has been updated! </p>";
    }
}
}

```

```

//if only the username field is blank, but the rest is not, change the password, email and
status
else if(empty($newUsername))
{
    $this -> newPassword = $newPassword;
    $this -> newEmail = $newEmail;
    $this -> setStatus = $setStatus;

    $hashedPwd = password_hash($this -> newPassword, PASSWORD_DEFAULT);

    $id = intval($userid);

    $this -> sql2 = "UPDATE accounts SET user_password='$hashedPwd',
user_email='$this->newEmail', user_status='$this->setStatus' WHERE user_id='$id'";
    $this -> sql2 .= "UPDATE replies SET user_status='$this->setStatus' WHERE
user_id='$id'";

    if(!$this -> db -> multi_query($this -> sql2))
    {
        echo "<script> alert('Multi Query failed!: " . $this -> db -> error . "');
</script>";
    }
    else
    {
        echo "<p id = 'succMessage'> The password and email have been changed and
the status has been updated! </p>";
    }
}
//if only the password field is blank, but the rest is not, change the username, email and
status
else if(empty($newPassword))
{
    $this -> newUsername = $newUsername;
    $this -> newEmail = $newEmail;
    $this -> setStatus = $setStatus;

    $id = intval($userid);

    $this -> sql2 = "UPDATE accounts SET user_username='$this->newUsername',
user_email='$this->newEmail', user_status='$this->setStatus' WHERE user_id='$id'";
    $this -> sql2 .= "UPDATE replies SET user_username='$this->newUsername',
user_status='$this->setStatus' WHERE user_id='$id'";
    $this -> sql2 .= "UPDATE threads SET thread_author='$this->newUsername'
WHERE user_id='$id'";

    if(!$this -> db -> multi_query($this -> sql2))

```

```

        {
            echo "<script> alert('Multi Query failed!: " . $this->db->error . "');
</script>";
        }
        else
        {
            echo "<p id = 'succMessage'> The username and email have been changed and
the status has been updated! </p>";
        }
    }
    //if only the email field is blank, but the rest is not, change the username, password and
status
    else if(empty($newEmail))
    {
        $this->newUsername = $newUsername;
        $this->newPassword = $newPassword;
        $this->setStatus = $setStatus;

        $hashedPwd = password_hash($this->newPassword, PASSWORD_DEFAULT);

        $id = intval($userid);

        $this->sql2 = "UPDATE accounts SET user_username='$this->newUsername',
user_password='$hashedPwd', user_status='$this->setStatus' WHERE user_id='$id'";
        $this->sql2 .= "UPDATE replies SET user_username='$this->newUsername',
user_status='$this->setStatus' WHERE user_id='$id'";
        $this->sql2 .= "UPDATE threads SET thread_author='$this->newUsername'
WHERE user_id='$id'";

        if(!$this->db->multi_query($this->sql2))
        {
            echo "<script> alert('Multi Query failed!: " . $this->db->error . "');
</script>";
        }
        else
        {
            echo "<p id = 'succMessage'> The username and password have been changed
and the status has been updated! </p>";
        }
    }
    //if only the status field is blank, but the rest is not, change the username, password and
email
    else if(empty($setStatus))
    {
        $this->newUsername = $newUsername;
        $this->newEmail = $newEmail;

```



```

$this->newPassword = $newPassword;

$hashedPwd = password_hash($this->newPassword, PASSWORD_DEFAULT);

$хid = intval($userid);

    $this->sql2 = "UPDATE accounts SET user_username='$this->newUsername',
user_email='$this->newEmail', user_password='$hashedPwd' WHERE user_id='$хid';";
    $this->sql2 .= "UPDATE replies SET user_username='$this->newUsername'
WHERE user_id='$хid';";
    $this->sql2 .= "UPDATE threads SET thread_author='$this->newUsername'
WHERE user_id='$хid';";

    if(!$this->db->multi_query($this->sql2))
    {
        echo "<script> alert('Multi Query failed!: " . $this->db->error . "');
</script>";
    }
    else
    {
        echo "<p id = 'succMessage'> The username, password and email have been
changed! </p>";
    }
}
//change only the username and password if the email and status fields are blank
else if(empty($newEmail) && empty($setStatus))
{
    $this->newUsername = $newUsername;
    $this->newPassword = $newPassword;

    $hashedPwd = password_hash($this->newPassword, PASSWORD_DEFAULT);

    $хid = intval($userid);

    $this->sql2 = "UPDATE accounts SET user_username='$this->newUsername',
user_password='$hashedPwd' WHERE user_id='$хid';";
    $this->sql2 .= "UPDATE replies SET user_username='$this->newUsername'
WHERE user_id='$хid';";
    $this->sql2 .= "UPDATE threads SET thread_author='$this->newUsername'
WHERE user_id='$хid';";

    if(!$this->db->multi_query($this->sql2))
    {
        echo "<script> alert('Multi Query failed!: " . $this->db->error . "');
</script>";
    }
}

```

```

else
{
    echo "<p id = 'succMessage'> The username and password has been changed!
</p>";
}
}
//change only the username and email if the password and status fields are empty
else if(empty($newPassword) && empty($setStatus))
{
    $this -> newUsername = $newUsername;
    $this -> newEmail = $newEmail;

    $id = intval($userid);

    $this -> sql2 = "UPDATE accounts SET user_username='$this->newUsername',
user_email='$this->newEmail' WHERE user_id='$id'";
    $this -> sql2 .= "UPDATE replies SET user_username='$this->newUsername'
WHERE user_id='$id'";
    $this -> sql2 .= "UPDATE threads SET thread_author='$this->newUsername'
WHERE user_id='$id'";

    if(!$this -> db -> multi_query($this -> sql2))
    {
        echo "<script> alert('Multi Query failed!: " . $this -> db -> error . "');
</script>";
    }
    else
    {
        echo "<p id = 'succMessage'> The username and email has been changed!
</p>";
    }
}
//change only the username and status if the password and email fields are empty
else if(empty($newPassword) && empty($newEmail))
{
    $this -> newUsername = $newUsername;
    $this -> setStatus = $setStatus;

    $id = intval($userid);

    $this -> sql2 = "UPDATE accounts SET user_username='$this->newUsername',
user_status='$this->setStatus' WHERE user_id='$id'";
    $this -> sql2 .= "UPDATE replies SET user_username='$this->newUsername',
user_status='$this->setStatus' WHERE user_id='$id'";
    $this -> sql2 .= "UPDATE threads SET thread_author='$this->newUsername'
WHERE user_id='$id'";

```

```

        if(!$this -> db -> multi_query($this -> sql2))
        {
            echo "<script> alert('Multi Query failed!: " . $this -> db -> error . "');
</script>";
        }
        else
        {
            echo "<p id = 'succMessage'> The username and status has been changed!
</p>";
        }
    }
    //change only the password and email if the username and status fields are empty
    else if(empty($newUsername) && empty($newEmail))
    {
        $this -> newPassword = $newPassword;
        $this -> newEmail = $newEmail;

        $hashedPwd = password_hash($this -> newPassword, PASSWORD_DEFAULT);

        $id = intval($userid);

        $this -> sql2 = "UPDATE accounts SET user_password='$hashedPwd',
user_email='$this->newEmail' WHERE user_id='$id'";

        if(!$this -> db -> query($this -> sql2))
        {
            echo "<script> alert('SQL ERROR: " . $this -> db -> error . "'); </script>";
        }
        else
        {
            echo "<p id = 'succMessage'> The password and email has been changed! </p>";
        }
    }
    //change only the password and status if the username and email fields are empty
    else if(empty($newUsername) && empty($newEmail))
    {
        $this -> newPassword = $newPassword;
        $this -> setStatus = $setStatus;

        $hashedPwd = password_hash($this -> newPassword, PASSWORD_DEFAULT);

        $id = intval($userid);

        $this -> sql2 = "UPDATE accounts SET user_password='$hashedPwd',
user_status='$this->setStatus' WHERE user_id='$id'";

```

```

    $this->sql2 .= "UPDATE replies SET user_status='$this->setStatus' WHERE
user_id='$id'";

    if(!$this->db->multi_query($this->sql2))
    {
        echo "<script> alert('Multi Query failed!: " . $this->db->error . "');
</script>";
    }
    else
    {
        echo "<p id = 'succMessage'> The password and status has been changed!
</p>";
    }
}
//change only the email and status if the username and password fields are empty
else if(empty($newUsername) && empty($newPassword))
{
    $this->newEmail = $newEmail;
    $this->setStatus = $setStatus;

    $id = intval($userid);

    $this->sql2 = "UPDATE accounts SET user_email='$this->newEmail',
user_status='$this->setStatus' WHERE user_id='$id'";
    $this->sql2 .= "UPDATE replies SET user_status='$this->setStatus' WHERE
user_id='$id'";

    if(!$this->db->multi_query($this->sql2))
    {
        echo "<script> alert('Multi Query failed!: " . $this->db->error . "');
</script>";
    }
    else
    {
        echo "<p id = 'succMessage'> The email and status has been changed! </p>";
    }
}
}
?>

```

## 6/24/19 - ...

Now that the part where you have to create the code where users have to create a topic thread on a particular subject and the code where users have to reply to those threads is over, now is time to move on to the following stuff:

## 6/25/19 – 6/27/19

1. You want to restyle the topic thread page (including the thread window, the reply window and the comment window, including styling the comment such as bolding it or italicizing it and etc...). You also want to add some animation effects and other dynamic contents. This is where your CSS and JavaScript skills come into play.

### ThreadHandler.php

```
<?php
    include_once 'database.php';

    /* This class will handle the processes of both creating the topic thread of a particular
    subject
    and displaying it on the forumhome.php and the subject pages */
    class ThreadHandler
    {
        private $db;

        //SQL variables
        private $sql1; //for the createThread() method. For storing the thread information in the
        thread table.
        private $sql2; //for the displayThread() method. For obtaining the thread information from
        the thread table and displaying it in their respective topic thread file
        private $sql3; //for the createReply() method. For obtaining the user comment and storing
        it in the replies table
        private $sql4; //for the displayReply() method. For obtaining the user comment from the
        replies table and displaying it in the respective topic thread

        public function __construct()
        {
            $this -> db = new Connection();
            $this -> db = $this -> db -> connect();
        }

        /* This method is responsible for obtaining the thread information and storing that
        information in the threads table */
        public function createThread($directory, $id, $author, $subject, $topic, $title, $content,
        $topicPage)
        {
            /* Before adding the thread information into the database, first append that information
            into a file called 'DiscussionWindow_(topic)_(title).php' */
            $threadfile = fopen("$directory/DiscussionWindow_{$topic}" . "_" . $title . ".php", "w");
```

```

    $threadPath = "$directory/DiscussionWindow_$topic" . "_" . " $title.php"; //full thread
file path
    $regularPath = "DiscussionWindow_$topic" . "_" . " $title.php"; //half thread file path
    $threadCreated = date("m/d/Y");

//set the thread page layout for the 'DiscussionWindow_(topic)_(title).php' page
    $threadlayout = "<?php
        session_start();
        include '../..../includes/ThreadHandler.php';
    ?>
    <html>
        <head>
            <title> $title </title>
            <meta charset = 'utf-8'/>

            <link rel = 'stylesheet' href = '../..../ThreadsCssFiles/Navbar.css'/>
            <link rel = 'stylesheet' href =
'../..../ThreadsCssFiles/DiscussionWindow.css'/>
        </head>

        <body id = '$subject'>
            <!-- Navbar -->
            <ul class = 'navbar'>
                <li class = 'left-tabs'> <a href = '../..../forumhome.php'> Home </a>
</li>

                <li class = 'left-tabs'> <a href = '#'> Board </a> </li>
                <li class = 'left-tabs'> <a href = '#'> Community </a> </li>

                <?php ".$$. "upic = ".$$. " _SESSION['pic']; ?>

                <li class = 'right-tabs'><a href = '../..../includes/logout.php'> Logout
</a></li>

                <li class = 'dropdown right-tabs'>
                    <a href = '#' class = 'dropdownButton'> Hello: <?php echo
".$$. " _SESSION['username']; ?> </a>
                    <div class = 'dropdown-content'>
                        <a href = '#'> <?php echo \"<img src = '../..../\"".$$. "upic'
height='70' width='70'/>\"; ?> </a>
                        <a href = '#'> Full Name: <?php echo
".$$. " _SESSION['firstname']".".\" \". ".$$. " _SESSION['lastname']; ?> </a>
                        <a href = '#'> Email: <?php echo ".$$. " _SESSION['useremail'];
?> </a>

                        <a href = '#'> Current Status: <?php echo
".$$. " _SESSION['status']; ?> </a>

```

```

        <a href = '#'> Join Date: <?php echo
"."$"."_SESSION['regdate']; ?> </a>
        <a href = '../../userprofile.php'> Go to account settings </a>
    </div>
</li>
</ul>

<!-- Thread window -->
<div class = 'threadWindow'>
    <div class = 'userInfo'>
        <div id = 'userPicture'> <?php echo \"<img src = '../../\".\"$\".\"upic'
height='70' width='70'/>\"; ?> </div>
        <h2> <?php echo \".\"$\".\"_SESSION['username']; ?> </h2>
        <h4> Join date: <?php echo \".\"$\".\"_SESSION['regdate']; ?> </h4>
        <p> Current Status: <?php echo \".\"$\".\"_SESSION['status']; ?>
</p>
    </div>
    <div class = 'threadContent'>
        <h1> $title </h1>
        <h4> Thread created: $threadCreated </h4>
        <h5> Topic: $topic </h5>
        <div class = 'underline' style = 'border-bottom: 1px solid
black;'></div>
        <p>
            $content
        </p>
    </div>
</div>

<br/><br/><br/>

<!-- Replies window -->
<?php
    "."$"."replies = new ThreadHandler();

    "."$"."Directory = \"$threadPath\";

    "."$"."replies -> displayReply(\".\"$\".\"Directory);
?>

<br/><br/><br/>

<!-- Comment window -->
<div class = 'commentWindow'>
    <div class = 'styleButtons'>

```

```

        <button id = 'bold' onclick = 'styleText(\"<b>\", \"</b>\")'> <abbr
title = 'Bold'> <b>B</b> </abbr> </button> <br/> <br/>
        <button id = 'italic' onclick = 'styleText(\"<em>\", \"</em>\")'>
<abbr title = 'Italic'> <em>i</em> </abbr> </button> <br/> <br/>
        <button id = 'underline' onclick = 'styleText(\"<u>\", \"</u>\")'>
<abbr title = 'Underline'> <u>U</u> </abbr> </button> <br/> <br/>
        <button id = 'subscript' onclick = 'styleText(\"<sub>\",
\"</sub>\")'> <abbr title = 'Subscript'> A<sub>2</sub> </abbr> </button> <br/> <br/>
        <button id = 'superscript' onclick = 'styleText(\"<sup>\",
\"</sup>\")'> <abbr title = 'Superscript'> A<sup>2</sup> </abbr> </button> <br/>
<br/>
        <button id = 'quote' onclick = 'styleText(\"<blockquote>\",
\"</blockquote>\")'> <abbr title = 'Quote'> \"Q\" </abbr> </button>
</div>
<div class = 'commentField'>
    <form method = 'POST' action = '$regularPath'>
        <textarea name = 'commentbox' id = 'commentbox' rows = '15'
cols = '90'></textarea> <br/>
        <input type = 'submit' name = 'submit-comment' value =
'Submit comment'>
    </form>
<?php
    if(isset(\".\"$\".\"_POST['submit-comment']))
    {
        \".\"$\".\"Replies = new ThreadHandler();

        \".\"$\".\"uid = \".\"$\".\"_SESSION['userid']; //id of the user who
made the comment/reply
        \".\"$\".\"uname = \".\"$\".\"_SESSION['username']; //username of
the user who made the comment/reply
        \".\"$\".\"upic = \".\"$\".\"_SESSION['pic']; //profile pic of user who
made the comment/reply
        \".\"$\".\"Title = \".\"$\".\"title\"; //title of the thread who the user is
replying to
        \".\"$\".\"Directory = \".\"$\".\"threadPath\"; //full file path of the thread
file
        \".\"$\".\"replyCreated = date('m/d/Y'); //date the reply was
created
        \".\"$\".\"Reply = \".\"$\".\"_POST['commentbox']; //the reply

        \".\"$\".\"Replies -> createReply(\".\"$\".\"uid, \".\"$\".\"uname,
\".\"$\".\"upic, \".\"$\".\"Title, \".\"$\".\"Directory, \".\"$\".\"replyCreated, \".\"$\".\"Reply);
    }
?>
</div>
</div>

```



```

        <script src = '../ThreadsJSFiles/DiscussionWindow.js'></script>
    </body>
</html>";

fwrite($threadfile, $threadlayout);

/* Now add the thread information to the database */
$this -> sql1 = "INSERT INTO threads(user_id, thread_author, thread_subject,
thread_topic, thread_title, thread_content, thread_file)
                VALUES('$id', '$author', '$subject', '$topic', '$title', '$content',
'$threadPath')";

$stmt1 = $this -> db -> stmt_init();

if(!$stmt1 -> prepare($this -> sql1))
{
    echo "<script> alert('SQL ERROR: ". $stmt1 -> error . "'); </script>";
}
else
{
    $stmt1 -> execute();
    header("Location: $topicPage?threadcreation=success");
}
}

/* This method is responsible for displaying the topic thread on the forumhome.php page
and the subject pages */
public function displayThread($subject, $topic)
{
    $this -> sql2 = "SELECT thread_author, thread_subject, thread_topic, thread_title,
thread_file
                    FROM threads
                    WHERE thread_subject='$subject' AND thread_topic='$topic'";

    $stmt2 = $this -> db -> stmt_init();

    if(!$stmt2 -> prepare($this -> sql2))
    {
        echo "<script> alert('SQL ERROR: ". $stmt2 -> error . "'); </script>";
    }
    else
    {
        $stmt2 -> execute();
    }
}

```

```

$result = $stmt2 -> get_result();

if($result -> num_rows > 0)
{
    while($row = $result -> fetch_assoc())
    {
        echo "<tr>
            <td><a href = '". $row['thread_file'] ."'>" . $row['thread_title'] . "</a></td>
            <td> Replies: <br/> Views: </td>
            <td> By: <em>" . $row['thread_author'] . "</em> </td>
        </tr>";
    }
}
else
{
    echo "<b id = 'warningMessage'> There are no current threads posted. </b>";
}
}

/* This method creates the comment a user made on a particular topic thread */
public function createReply($id, $username, $pic, $title, $directory, $dateCreated,
$comment)
{
    /* Store the comment in the database with the respective thread file */
    if(empty($comment))
    {
        echo "<p id = 'errormessage'> You must enter a comment! </p>";
    }
    else
    {
        $this -> sql3 = "INSERT INTO replies(user_id, user_username, profile_pic,
thread_title, thread_file, replies_creation, replies_content)
VALUES('$id', '$username', '$pic', '$title' , '$directory', '$dateCreated',
'$comment')";

        $stmt3 = $this -> db -> stmt_init();

        if(!$stmt3 -> prepare($this -> sql3))
        {
            echo "<script> alert('SQL ERROR: ". $stmt3 -> error . "'); </script>";
        }
        else
        {
            $stmt3 -> execute();
            echo "<script> alert('Reply created!'); </script>";
        }
    }
}

```

```

    }
  }
}

/* This method displays the comments that users made on a particular topic thread */
public function displayReply($directory)
{
  /* Get only the replies of that particular topic thread */
  $this -> sql4 = "SELECT user_username, user_status, profile_pic, thread_file,
replies_creation, replies_content
                  FROM replies
                  WHERE thread_file = '$directory'";

  $stmt4 = $this -> db -> stmt_init();

  if(!$stmt4 -> prepare($this -> sql4))
  {
    echo "<script> alert('SQL ERROR: ". $stmt4 -> error . "'); </script>";
  }
  else
  {
    $stmt4 -> execute();

    $result = $stmt4 -> get_result();

    if($result -> num_rows > 0)
    {
      while($row = $result -> fetch_assoc())
      {
        echo "<div class = 'repliesWindow'>
              <div class = 'userInfo'>
                <img src = '" . $row['profile_pic'] . "' height='70'
width='70/'> <br/>
                <h2>" . $row['user_username'] . "</h2>
                <p> Current Status: ". $row['user_status'] . "</p>
              </div>
              <div class = 'replyContent'>
                <p>" . $row['replies_content'] . "</p>
              </div>
            </div> <br/><br/>";
      }
    }
    else
    {
      echo "<b id = 'warningMessage'> There are no current replies posted. </b>";
    }
  }
}

```

```
}  
}  
}  
?>
```

### DiscussionWindow.css

**/\* Style the thread window \*/**

**.threadWindow**

```
{  
  margin-top: 5%;  
  display: flex; /* This will keep the divs in the threadWindow the same size and in a  
single row */  
}
```

**.threadWindow .userInfo, .threadWindow .threadContent**

```
{  
  background-color: #FFFFFF0; /* Beige background */
```

**border: 1px solid black;**

**padding: 3%;**

**text-align: center;**

**font-family: 'Noticia Text', serif;**

```
}
```

**.threadWindow .userInfo**

```
{  
  margin-left: 25%;
```

**border-radius: 5px 0 0 5px;**

```
}
```

**.threadWindow .threadContent**

```
{  
  border-radius: 0 5px 5px 0;  
}
```

**/\* Style the replies window \*/**

**.repliesWindow**

```
{  
  display: flex; /* This will keep the divs in the repliesWindow the same size and in a  
single row */  
}
```

**.repliesWindow .userInfo, .repliesWindow .replyContent**

```
{  
  background-color: #FFFFFF0; /* Beige background */
```

**border: 1px solid black;**

```
padding: 3%;

text-align: center;
font-family: 'Noticia Text', serif;
}
.repliesWindow .userInfo
{
margin-left: 25%;

border-radius: 5px 0 0 5px;
}
.repliesWindow .replyContent
{
border-radius: 0 5px 5px 0;
}

/* Style the comment window */
.commentWindow
{
display: flex; /* This will keep the divs in the commentWindow the same size and in a
single row */

background-color: lightgrey;

border-radius: 5px;
border-style: inset;

margin-left: 8%;
margin-right: 8%;

padding-top: 20px;
}
.commentWindow .styleButtons
{
margin-left: 30%;
}
.commentWindow .styleButtons button
{
border: none;
border-radius: 3px;
border-style: inset;
padding: 8px;

cursor: pointer;
}
```

```
.commentWindow .commentField
{
    text-align: center;
}
.commentWindow .commentField textarea
{
    border-radius: 5px;
    border-style: inset;

    resize: none;

    font-family: 'Noticia Text', serif;
}
.commentWindow .commentField input[type=submit]
{
    font-family: 'Noticia Text', serif;

    padding: 10px;

    cursor: pointer;

    border: none;
    border-radius: 5px;
    border-style: inset;
}

/* Style the warning message */
#warningMessage
{
    background-color: red;
    opacity: 0.6;

    text-align: center;

    border-radius: 5px;
    padding: 10px;

    margin-left: 30%;
}

/*Sets the backgrounds of the Discussion Windows*/
#Culture
{
    background-image:
url("https://png.pngtree.com/thumb_back/fw800/back_pic/02/67/99/70578f60912f607.jpg");
    background-repeat: no-repeat;
```

```

    background-size: cover;
}
#ScienceMath
{
    background-image: url("https://encrypted-
tbn0.gstatic.com/images?q=tbn:ANd9GcS0Y_2asRVMCmOaZTbWj5eS8YUTUTR1j_zV6_
YuT3UrLHFN76pbqw");
    background-repeat: no-repeat;
    background-size: cover;
}
#SocialScience
{
    background-image: url("https://encrypted-
tbn0.gstatic.com/images?q=tbn:ANd9GcRONyx4xEIg7ovmxzLfOrDRRmJ5Jj4fbBG5PkB-
BrFUclmz9AiT");
    background-repeat: no-repeat;
    background-size: cover;
}
#Tech
{
    background-image: url("https://encrypted-
tbn0.gstatic.com/images?q=tbn:ANd9GcTGQnKhHxMAWv4YG7i1DWjLL_xjhr4ZnYRPt4o
KbP88K-BnQIBP");
    background-repeat: no-repeat;
    background-size: cover;
}
#WorldNews
{
    background-image: url("https://ak7.picdn.net/shutterstock/videos/8558317/thumb/1.jpg");
    background-repeat: no-repeat;
    background-size: cover;
}

```

Also in the threads.css file, you want to add the following:

```

threads.css
...
/* Style the warning message */
#warningMessage
{
    background-color: red;
    opacity: 0.6;

    text-align: center;

    border-radius: 5px;

```

```
padding: 10px;
}
...
```

Finally, you want to create a JavaScript file called “DiscussionWindow.js”, add it to the ThreadJSFiles folder in the Subjects folder. Add the following:

#### **DiscussionWindow.js**

```
//use the window.getSelection() method to obtain highlighted text in the comment box
//Use following link as reference: https://jsfiddle.net/qpw1eemr/1/

function styleText(x, y)
{
    var commentbox = document.getElementById("commentbox"); //obtains the textarea
    comment box

    var startText = commentbox.selectionStart; //gets the beginning of the selected portion of the
    comment text in the textarea
    var endText = commentbox.selectionEnd; //gets the end of the selected portion of the
    comment text in the textarea
    var allText = commentbox.value; //gets all the comment text input from the comment box
    textarea

    var selectedText = allText.substring(startText, endText); //obtains the selected text from start
    letter to end letter
    var newText; //the new comment input to be styled

    /* inside the newText variable translates to: "go from the beginning text all the way to just
    before the beginning of the selected text,
    then add a beginning style tag (x) before the beginning of the selected text, then add the end
    style tag (y) after the end of the selected text,
    then continue on from the end of the selected text to the end of the text" */
    if(x == "<b>" && y == "</b>") //bold the selected text
    {
        newText = allText.substring(0, startText) + "<b>" + selectedText + "</b>" +
        allText.substring(endText, allText.length);
    }
    else if(x == "<em>" && y == "</em>") //italicize the selected text
    {
        newText = allText.substring(0, startText) + "<em>" + selectedText + "</em>" +
        allText.substring(endText, allText.length);
    }
    else if(x == "<u>" && y == "</u>") //underline the selected text
    {
        newText = allText.substring(0, startText) + "<u>" + selectedText + "</u>" +
        allText.substring(endText, allText.length);
    }
}
```



```

else if(x == "<sub>" && y == "</sub>") //subscript the selected text
{
    newText = allText.substring(0, startText) + "<sub>" + selectedText + "</sub>" +
allText.substring(endText, allText.length);
}
else if(x == "<sup>" && y == "</sup>") //superscript the selected text
{
    newText = allText.substring(0, startText) + "<sup>" + selectedText + "</sup>" +
allText.substring(endText, allText.length);
}
else if(x == "<blockquote>" && y == "</blockquote>") //quote a text from another user
{
    newText = allText.substring(0, startText) + "<blockquote style = 'background-color:
lightgrey;'" + selectedText + "</blockquote>" + allText.substring(endText, allText.length);
}

commentbox.value = newText; //replace the original comment input with the comment input
to be bolded
}

```

---

**6/28/19**

2. Whenever users update their information such as profile picture, status or username, that updated information will not show up on any page (such as the navbar or thread pages) until the users has logged out and logged back in. You want to find a way to display that information right away when the user has finished updating their information.

First, fix the uploadPics() method in the ProfileHandler.php page so that new uploaded user profile picture can show up immediately after uploading new picture:

```

ProfileHandler.php
...
/* This method uploads new profile pictures */
/* This will update the profile_pic column in both the accounts and replies table */
public function uploadPics($id)
{
    $this -> folder = "ProfilePics/";
    $this -> photo = $this -> folder . basename($_FILES["getFile"]["name"]); //obtains the
photo to be updated and store them in the ProfilePics folder
    $this -> imageFileType = strtolower(pathinfo($this -> photo,
PATHINFO_EXTENSION)); //obtain the path of the type of file (.jpg, .png, etc...)
    $this -> check = getimagesize($_FILES["getFile"]["tmp_name"]); //checks the size of
the image

```

```

//check if the file is a real image or not
if($this -> check !== false)
{
    //if the image file is not jpg, jpeg, png and gif files
    if($this -> imageFileType != "jpg" && $this -> imageFileType != "png" && $this ->
imageFileType != "jpeg" && $this -> imageFileType != "gif")
    {
        echo "<p id = 'errorMessage'> Sorry, only JPG, JPEG, PNG, and GIF files are allowed!
</p>";
    }
    else //else if the image file is a jpg, jpeg, png, and/or gif
    {
        //if the file is too large
        if($_FILES["getFile"]["size"] > 500000)
        {
            echo "<p id = 'errorMessage'> Sorry, your file is too big to upload! </p>";
        }
        else //else if file is not too large
        {
            $pic = $this -> photo; //this variable is just a placeholder to hold the photo variable

            //upload the image
            if(move_uploaded_file($_FILES["getFile"]["tmp_name"], $this -> photo))
            {
                /* update the profile_pic columns in both the accounts and replies table */
                $this -> sql1 = "UPDATE accounts SET profile_pic='$pic' WHERE
user_id='$id'";
                $this -> sql1 .= "UPDATE replies SET profile_pic='$pic' WHERE
user_id='$id'";

                if(!$this -> db -> multi_query($this -> sql1))
                {
                    echo "<script> alert('Multi Query failed: " . $this -> db -> error . "');
</script>";
                }
                else
                {
                    echo "<p id = 'succMessage'> Your picture was successfully uploaded!
Please wait 2 seconds for new picture to show up! </p>";
                    $_SESSION['pic'] = $pic; //update the profile picture session variable to
have it change right away after uploading new pic
                    header("refresh: 2"); //refresh the current page for 2 seconds
                }
            }
            //if the image could not be uploaded
            else

```

```

        {
            echo "<p id = 'errMessage'> Sorry, there was a problem uploading your image
file </p>";
        }
    }
}
else
{
    echo "<p id = 'errMessage'> The image you selected is not a file! </p>";
}
}
...

```

Next, you want to do the same with the `updateInfo()` method (note: you do not need to do the same for the user password, because you only want the information that is seen to be changed and the changes to be seen immediately, not the password):

#### **ProfileHandler.php**

```

...
/* This method updates user information such as name, email, password, etc */
public function updateInfo($userid, $newUsername, $newPassword, $newEmail,
    $setStatus)
{
    //if the username field is not empty, but the others are, change only the username
    if(empty($newPassword) && empty($newEmail) && empty($setStatus))
    {
        $this->newUsername = $newUsername; //private $newUsername =
        updateInfo($newUsername)

        $id = intval($userid);

        $this->sql2 = "UPDATE accounts SET user_username='$this->newUsername'
WHERE user_id='$id'";
        $this->sql2 .= "UPDATE replies SET user_username='$this->newUsername'
WHERE user_id='$id'";
        $this->sql2 .= "UPDATE threads SET thread_author='$this->newUsername'
WHERE user_id='$id'";

        if(!$this->db->multi_query($this->sql2))
        {
            echo "<script> alert('Multi Query failed! : " . $this->db->error . "'); </script>";
        }
        else
        {

```

```

    $_SESSION['username'] = $this -> newUsername; //change the username
session variable immediately after updating
    echo "<p id = 'succMessage'> Username has been changed! Please wait 2
seconds for new information to be updated! </p>";
    header("refresh: 2"); //refresh the current page for 2 seconds
}
}
//if the password field is not empty, but the others are, change only the password
else if(empty($newUsername) && empty($newEmail) && empty($setStatus))
{
    $this -> newPassword = $newPassword; //private $newPassword =
updateInfo($newPassword)
    $hashedPwd = password_hash($this -> newPassword, PASSWORD_DEFAULT);
//must hash the password to prevent hackers from seeing user password

    $id = intval($userid);

    $this -> sql2 = "UPDATE accounts SET user_password='$hashedPwd' WHERE
user_id='$id'";

    if(!$this -> db -> query($this -> sql2))
    {
        echo "<script> alert('SQL ERROR: " . $this -> db -> error . "'); </script>";
    }
    else
    {
        echo "<p id = 'succMessage'> Password has been changed! </p>";
    }
}
//if the email field is not empty, but the others are, change only the email
else if(empty($newUsername) && empty($newPassword) && empty($setStatus))
{
    $this -> newEmail = $newEmail; //private $newEmail = updateInfo($newEmail)

    $id = intval($userid);

    $this -> sql2 = "UPDATE accounts SET user_email='$this->newEmail' WHERE
user_id='$id'";

    if(!$this -> db -> query($this -> sql2))
    {
        echo "<script> alert('SQL ERROR: " . $this -> db -> error . "'); </script>";
    }
    else
    {

```

```

        $_SESSION['useremail'] = $this -> newEmail; //change the user email session
variable immediately after updating
        echo "<p id = 'succMessage'> Email has been changed! Please wait 2 seconds
for information to be uploaded! </p>";
        header("refresh: 2"); //refresh the current page for 2 seconds
    }
}
//if the status is not empty, but the others are, change only the status
else if(empty($newUsername) && empty($newPassword) && empty($newEmail))
{
    $this -> setStatus = $setStatus; //private $setStatus = updateInfo($setStatus)

    $id = intval($userid);

    $this -> sql2 = "UPDATE accounts SET user_status='$this->setStatus' WHERE
user_id='$id'";
    $this -> sql2 .= "UPDATE replies SET user_status='$this->setStatus' WHERE
user_id='$id'";

    if(!$this -> db -> multi_query($this -> sql2))
    {
        echo "<script> alert('Multi Query failed!: " . $this -> db -> error . "'); </script>";
    }
    else
    {
        $_SESSION['status'] = $this -> setStatus; //change the user status session
variable immediately after updating
        echo "<p id = 'succMessage'> Status has been updated! Please wait 2 seconds
for information to be uploaded! </p>";
        header("refresh: 2"); //refresh the current page for 2 seconds
    }
}
//if only the username field is blank, but the rest is not, change the password, email and
status
else if(empty($newUsername))
{
    $this -> newPassword = $newPassword;
    $this -> newEmail = $newEmail;
    $this -> setStatus = $setStatus;

    $hashedPwd = password_hash($this -> newPassword, PASSWORD_DEFAULT);

    $id = intval($userid);

    $this -> sql2 = "UPDATE accounts SET user_password='$hashedPwd',
user_email='$this->newEmail', user_status='$this->setStatus' WHERE user_id='$id'";

```

```

    $this -> sql2 .= "UPDATE replies SET user_status='$this->setStatus' WHERE
user_id='$id'";

    if(!$this -> db -> multi_query($this -> sql2))
    {
        echo "<script> alert('Multi Query failed!: " . $this -> db -> error . "'); </script>";
    }
    else
    {
        $_SESSION['useremail'] = $this -> newEmail; //change the user email session
variable immediately after updating
        $_SESSION['status'] = $this -> setStatus; //change the user status session
variable immediately after updating
        echo "<p id = 'succMessage'> The password and email have been changed and
the status has been updated! Please wait 2 seconds for information to be uploaded!
</p>";
        header("refresh: 2"); //refresh the current page for 2 seconds
    }
}
//if only the password field is blank, but the rest is not, change the username, email and
status
else if(empty($newPassword))
{
    $this -> newUsername = $newUsername;
    $this -> newEmail = $newEmail;
    $this -> setStatus = $setStatus;

    $id = intval($userid);

    $this -> sql2 = "UPDATE accounts SET user_username='$this->newUsername',
user_email='$this->newEmail', user_status='$this->setStatus' WHERE user_id='$id'";
    $this -> sql2 .= "UPDATE replies SET user_username='$this->newUsername',
user_status='$this->setStatus' WHERE user_id='$id'";
    $this -> sql2 .= "UPDATE threads SET thread_author='$this->newUsername'
WHERE user_id='$id'";

    if(!$this -> db -> multi_query($this -> sql2))
    {
        echo "<script> alert('Multi Query failed!: " . $this -> db -> error . "'); </script>";
    }
    else
    {
        $_SESSION['username'] = $this -> newUsername; //change the username
session variable immediately after updating
        $_SESSION['useremail'] = $this -> newEmail; //change the user email session
variable immediately after updating

```

```

        $_SESSION['status'] = $this -> setStatus; //change the user status session
variable immediately after updating
        echo "<p id = 'succMessage'> The username and email have been changed and
the status has been updated! Please wait 2 seconds for information to be uploaded!
</p>";
        header("refresh: 2"); //refresh the current page for 2 seconds
    }
}
//if only the email field is blank, but the rest is not, change the username, password and
status
else if(empty($newEmail))
{
    $this -> newUsername = $newUsername;
    $this -> newPassword = $newPassword;
    $this -> setStatus = $setStatus;

    $hashedPwd = password_hash($this -> newPassword, PASSWORD_DEFAULT);

    $id = intval($userid);

    $this -> sql2 = "UPDATE accounts SET user_username='$this->newUsername',
user_password='$hashedPwd', user_status='$this->setStatus' WHERE user_id='$id'";
    $this -> sql2 .= "UPDATE replies SET user_username='$this->newUsername',
user_status='$this->setStatus' WHERE user_id='$id'";
    $this -> sql2 .= "UPDATE threads SET thread_author='$this->newUsername'
WHERE user_id='$id'";

    if(!$this -> db -> multi_query($this -> sql2))
    {
        echo "<script> alert('Multi Query failed!: " . $this -> db -> error . "'); </script>";
    }
    else
    {
        $_SESSION['username'] = $this -> newUsername; //change the username
session variable immediately after updating
        $_SESSION['status'] = $this -> setStatus; //change the user status session
variable immediately after updating
        echo "<p id = 'succMessage'> The username and password have been changed
and the status has been updated! Please wait 2 seconds for information to be uploaded!
</p>";
        header("refresh: 2"); //refresh the current page for 2 seconds
    }
}
//if only the status field is blank, but the rest is not, change the username, password and
email
else if(empty($setStatus))

```

```

{
    $this -> newUsername = $newUsername;
    $this -> newEmail = $newEmail;
    $this -> newPassword = $newPassword;

    $hashedPwd = password_hash($this -> newPassword, PASSWORD_DEFAULT);

    $id = intval($userid);

    $this -> sql2 = "UPDATE accounts SET user_username='$this->newUsername',
user_email='$this->newEmail', user_password='$hashedPwd' WHERE user_id='$id'";
    $this -> sql2 .= "UPDATE replies SET user_username='$this->newUsername'
WHERE user_id='$id'";
    $this -> sql2 .= "UPDATE threads SET thread_author='$this->newUsername'
WHERE user_id='$id'";

    if(!$this -> db -> multi_query($this -> sql2))
    {
        echo "<script> alert('Multi Query failed!: " . $this -> db -> error . "'); </script>";
    }
    else
    {
        $_SESSION['username'] = $this -> newUsername; //change the username
session variable immediately after updating
        $_SESSION['useremail'] = $this -> newEmail; //change the user email session
variable immediately after updating
        echo "<p id = 'succMessage'> The username, password and email have been
changed! Please wait 2 seconds for information to be uploaded! </p>";
        header("refresh: 2"); //refresh the current page for 2 seconds
    }
}

//change only the username and password if the email and status fields are blank
else if(empty($newEmail) && empty($setStatus))
{
    $this -> newUsername = $newUsername;
    $this -> newPassword = $newPassword;

    $hashedPwd = password_hash($this -> newPassword, PASSWORD_DEFAULT);

    $id = intval($userid);

    $this -> sql2 = "UPDATE accounts SET user_username='$this->newUsername',
user_password='$hashedPwd' WHERE user_id='$id'";
    $this -> sql2 .= "UPDATE replies SET user_username='$this->newUsername'
WHERE user_id='$id'";

```



```

    $this -> sql2 .= "UPDATE threads SET thread_author='$this->newUsername'
WHERE user_id='$id'";

    if(!$this -> db -> multi_query($this -> sql2))
    {
        echo "<script> alert('Multi Query failed!: " . $this -> db -> error . "'); </script>";
    }
    else
    {
        $_SESSION['username'] = $this -> newUsername; //change the username
session variable immediately after updating
        echo "<p id = 'succMessage'> The username and password has been changed!
Please wait 2 seconds for information to be uploaded! </p>";
        header("refresh: 2"); //refresh the current page for 2 seconds
    }
}
//change only the username and email if the password and status fields are empty
else if(empty($newPassword) && empty($setStatus))
{
    $this -> newUsername = $newUsername;
    $this -> newEmail = $newEmail;

    $id = intval($userid);

    $this -> sql2 = "UPDATE accounts SET user_username='$this->newUsername',
user_email='$this->newEmail' WHERE user_id='$id'";
    $this -> sql2 .= "UPDATE replies SET user_username='$this->newUsername'
WHERE user_id='$id'";
    $this -> sql2 .= "UPDATE threads SET thread_author='$this->newUsername'
WHERE user_id='$id'";

    if(!$this -> db -> multi_query($this -> sql2))
    {
        echo "<script> alert('Multi Query failed!: " . $this -> db -> error . "'); </script>";
    }
    else
    {
        $_SESSION['username'] = $this -> newUsername; //change the username
session variable immediately after updating
        $_SESSION['useremail'] = $this -> newEmail; //change the user email session
variable immediately after updating
        echo "<p id = 'succMessage'> The username and email has been changed!
Please wait 2 seconds for information to be uploaded! </p>";
        header("refresh: 2"); //refresh the current page for 2 seconds
    }
}
}

```

```

//change only the username and status if the password and email fields are empty
else if(empty($newPassword) && empty($newEmail))
{
    $this->newUsername = $newUsername;
    $this->setStatus = $setStatus;

    $id = intval($userid);

    $this->sql2 = "UPDATE accounts SET user_username='$this->newUsername',
user_status='$this->setStatus' WHERE user_id='$id'";
    $this->sql2 .= "UPDATE replies SET user_username='$this->newUsername',
user_status='$this->setStatus' WHERE user_id='$id'";
    $this->sql2 .= "UPDATE threads SET thread_author='$this->newUsername'
WHERE user_id='$id'";

    if(!$this->db->multi_query($this->sql2))
    {
        echo "<script> alert('Multi Query failed!: " . $this->db->error . "'); </script>";
    }
    else
    {
        $_SESSION['username'] = $this->newUsername; //change the username
session variable immediately after updating
        $_SESSION['status'] = $this->setStatus; //change the user status session
variable immediately after updating
        echo "<p id = 'succMessage'> The username and status has been changed!
Please wait 2 seconds for information to be uploaded! </p>";
        header("refresh: 2"); //refresh the current page for 2 seconds
    }
}

//change only the password and email if the username and status fields are empty
else if(empty($newUsername) && empty($newEmail))
{
    $this->newPassword = $newPassword;
    $this->newEmail = $newEmail;

    $hashedPwd = password_hash($this->newPassword, PASSWORD_DEFAULT);

    $id = intval($userid);

    $this->sql2 = "UPDATE accounts SET user_password='$hashedPwd',
user_email='$this->newEmail' WHERE user_id='$id'";

    if(!$this->db->query($this->sql2))
    {
        echo "<script> alert('SQL ERROR: " . $this->db->error . "'); </script>";
    }
}

```

```

    }
    else
    {
        $_SESSION['useremail'] = $this -> newEmail; //change the user email session variable immediately after updating
        echo "<p id = 'succMessage'> The password and email has been changed! Please wait 2 seconds for information to be uploaded! </p>";
        header("refresh: 2"); //refresh the current page for 2 seconds
    }
}
//change only the password and status if the username and email fields are empty
else if(empty($newUsername) && empty($newEmail))
{
    $this -> newPassword = $newPassword;
    $this -> setStatus = $setStatus;

    $hashedPwd = password_hash($this -> newPassword, PASSWORD_DEFAULT);

    $id = intval($userid);

    $this -> sql2 = "UPDATE accounts SET user_password='$hashedPwd', user_status='$this->setStatus' WHERE user_id='$id'";
    $this -> sql2 .= "UPDATE replies SET user_status='$this->setStatus' WHERE user_id='$id'";

    if(!$this -> db -> multi_query($this -> sql2))
    {
        echo "<script> alert('Multi Query failed!: " . $this -> db -> error . "'); </script>";
    }
    else
    {
        $_SESSION['status'] = $this -> setStatus; //change the user status session variable immediately after updating
        echo "<p id = 'succMessage'> The password and status has been changed! Please wait 2 seconds for information to be uploaded! </p>";
        header("refresh: 2"); //refresh the current page for 2 seconds
    }
}
//change only the email and status if the username and password fields are empty
else if(empty($newUsername) && empty($newPassword))
{
    $this -> newEmail = $newEmail;
    $this -> setStatus = $setStatus;

    $id = intval($userid);

```

```

        $this->sql2 = "UPDATE accounts SET user_email='$this->newEmail',
user_status='$this->setStatus' WHERE user_id='$id'";
        $this->sql2 .= "UPDATE replies SET user_status='$this->setStatus' WHERE
user_id='$id'";

        if(!$this->db->multi_query($this->sql2))
        {
            echo "<script> alert('Multi Query failed!: " . $this->db->error . "'); </script>";
        }
        else
        {
            $_SESSION['useremail'] = $this->newEmail; //change the user email session
variable immediately after updating
            $_SESSION['status'] = $this->setStatus; //change the user status session
variable immediately after updating
            echo "<p id = 'succMessage'> The email and status has been changed! Please
wait 2 seconds for information to be uploaded! </p>";
            header("refresh: 2"); //refresh the current page for 2 seconds
        }
    }
}
...

```

## 6/29/19

Next thing you want to do is similar to the above, but for the thread pages. When a user replies to a thread, their reply won't show up only until the user leaves the page, then comes back to it. You want to fix that bug, make the reply appear right away. In the createReply() method of the ThreadHandler.php, add the following:

```

ThreadHandler.php
...
    /* This method creates the comment a user made on a particular topic thread */
    public function createReply($id, $username, $pic, $title, $directory, $dateCreated,
    $comment)
    {
        /* Store the comment in the database with the respective thread file */
        if(empty($comment))
        {
            echo "<p id = 'errormessage'> You must enter a comment! </p>";
        }
        else
        {
            $this->sql3 = "INSERT INTO replies(user_id, user_username, profile_pic,
thread_title, thread_file, replies_creation, replies_content)
                VALUES('$id', '$username', '$pic', '$title' , '$directory', '$dateCreated',
'$comment')";

```

```

$stmt3 = $this->db->stmt_init();

if(!$stmt3->prepare($this->sql3))
{
    echo "<script> alert('SQL ERROR: " . $stmt3->error . "'); </script>";
}
else
{
    $stmt3->execute();
    echo "<script> location.assign(location.href); </script>"; //refreshes current
    thread page to show replies as they are created
}
}
}
...

```

- Now you want to add the code that shows how many replies there are for a particular topic thread. It's just displaying a simple number of how many there are, nothing special. You should also take this opportunity to take out a few unnecessary stuff from your page such as 'latest post' and 'views' (it's very hard finding how much people visited a particular topic thread). So for each topic page, you want to have the following inside the thread window:

```

...
<div id = "threadwindow">
    <table class = "table">
        <thead>
            <tr>
                <th> Title/Threads </th>
                <th> Replies </th>
                <th> Author </th>
            </tr>
        </thead>

        <tbody>
            <tr>
                <?php
                    include '../includes/ThreadHandler.php';

                    $Subject = "(subject)";
                    $Topic = "(topic)";

                    $displayThreads = new ThreadHandler();

```

```

                $displayThreads -> displayThread($Subject, $Topic);
            ?>
        </tr>
    </tbody>
</table>
</div>
...

```

To show the number of replies on the topic thread, edit the entire displayThread() method in the ThreadHandler.php file:

### **ThreadHandler.php**

```

...
    /* This method is responsible for displaying the topic thread on the forumhome.php page
    and the subject pages */
    public function displayThread($subject, $topic)
    {
        /* This query is for displaying each topic thread of a particular subject */
        $this -> sql2 = "SELECT thread_author, thread_subject, thread_topic, thread_title,
thread_file
                        FROM threads
                        WHERE thread_subject='$subject' AND thread_topic='$topic'";

        $stmt2 = $this -> db -> stmt_init();

        if(!$stmt2 -> prepare($this -> sql2))
        {
            echo "<script> alert('SQL ERROR: " . $stmt2 -> error . "'); </script>";
        }
        else
        {
            $stmt2 -> execute();
            $result1 = $stmt2 -> get_result();

            if($result1 -> num_rows > 0)
            {
                while($row1 = $result1 -> fetch_assoc()) //stores the results of the thread table
query in an associative array and stores it in a variable called '$row1'
                {
                    $dir = $row1['thread_file']; //obtains the path of the topic thread at hand

                    /* This query is for obtaining the number of replies made to a particular topic
thread */
                    $this -> sql5 = "SELECT COUNT(replies_id) AS 'count' FROM replies WHERE
thread_file='$dir'"; //counts how many replies were made to a topic thread

```

```

$stmt5 = $this -> db -> stmt_init();

if(!$stmt5 -> prepare($this -> sql5))
{
    echo "<script> alert('SQL ERROR: " . $stmt5 -> error . "'); </script>";
}
else
{
    $stmt5 -> execute();
    $result2 = $stmt5 -> get_result();

    while($row2 = $result2 -> fetch_assoc()) //stores the results of the replies table
query in an associative array and stores it in a variable called '$row2'
    {
        echo "<tr>
            <td><a href = '" . $row1['thread_file'] . "'>" . $row1['thread_title'] .
" </a></td>
            <td>" . $row2['count'] . "</td>
            <td> By: <em>" . $row1['thread_author'] . "</em> </td>
            </tr>";
    }
}
}
}
else
{
    echo "<b id = 'warningMessage'> There are no current threads posted. </b>";
}
}
}
...

```

Now this will display (in the topic pages) the number of replies made to a topic thread on a particular subject.

**6/30/19**

Don't forget to add the following on all topic pages of all subjects:

```
...
...
<div id = "threadwindow">
    <table class = "table">
        <thead>
            <tr>
                <th> Title/Threads </th>
                <th> Replies </th>
                <th> Author </th>
            </tr>
        </thead>

        <tbody>
            <tr>
                <?php
                    include '../includes/ThreadHandler.php';

                    $Subject = "(subject)";
                    $Topic = "(topic)";

                    $displayThreads = new ThreadHandler();
                    $displayThreads -> displayThread($Subject, $Topic);
                ?>
            </tr>
        </tbody>
    </table>
</div>
...
...
```



4. In the forumhome.php page, see the latest post on a particular subject. You want to add the code that will display the last thread that was made on a particular subject. For the SQL code, you will only need to obtain the thread title, thread author and thread\_subject where the thread id is the most recent one (the most recent thread id is usually a higher number, thus the thread title is “higher” as well). In the ThreadHandler.php file, add a method called “obtainTopicThread”, and this method will do everything explained above:

#### **ThreadHandler.php**

```
...
/* This method obtains the last topic thread that was made on a particular subject */
public function obtainTopicThread($subject, $topic)
{
    /* This SQL query selects the most recent topic thread of a particular subject.
       The most recent one has a high thread id number, thus the thread title for that
       thread must also be high as well */
    $this->sql6 = "SELECT MAX(thread_id), thread_author, thread_subject,
thread_topic, MAX(thread_title), thread_file, MAX(thread_creation)
                FROM threads WHERE thread_subject='$subject' AND
thread_topic='$topic'";

    $stmt6 = $this->db->stmt_init();

    if(!$stmt6->prepare($this->sql6))
    {
        echo "<script> alert('SQL ERROR: ". $stmt6->error . "'); </script>";
    }
    else
    {
        $stmt6->execute();

        $result = $stmt6->get_result();

        while($row = $result->fetch_assoc())
        {
            if($row['MAX(thread_id)'] == null && $row['thread_author'] == null &&
$row['thread_subject'] == null
            && $row['thread_topic'] == null && $row['MAX(thread_title)'] == null &&
$row['thread_file'] == null
            && $row['MAX(thread_creation)'] == null)
            {
                echo "No threads for this topic have been posted";
            }
            else
            {
                echo "<p>
```

```

                <a href = 'Subjects/$subject/'. $row['thread_file'] ." "> ".
$row['MAX(thread_title)] ." </a> </br>
                by <em> ". $row['thread_author'] ." </em> <br/>
                Created on: ". $row['MAX(thread_creation)] ."
            </p>";
        }
    }
}
}
...

```

Now in the forumhome.php page, you want to add the following:

### **forumhome.php**

```

...
<?php
    session_start();

    include "includes/ThreadHandler.php";
    $(topic)Threads = new ThreadHandler(); //for displaying the most recent (topic) thread
    ...(keep adding for all subject topics)
?>
...

...
<table class = "table">
    <thead>
        <tr>
            <th class = "leftside"> (Subject) </th>
            <th class = "rightside"> Latest Posts </th>
        </tr>
    </thead>

    <tbody>
        <tr>
            <td class = "leftside"> <a href = "Subjects/(Subject)/(Topic).php">(Topic)</a>
        </td>
            <td class = "rightside"> <?php $(topic)Threads ->
obtainTopicThread("(Subject)", "(Topic)"); ?> </td>
        </tr>
        ... (keep adding for all topics)
    </tbody>
</table>
...

```

7/1/19

5. What you want to do now is re-decorate some of the topic threads and other pages such as making the scrollbar fixed while scrolling down, fixing the background of the subject threads, and deleting some unnecessary stuff throughout the pages.

You want to create a variable in the ThreadHandler.php file called “\$subjectStyling”, and this is primarily for the DiscussionWindow.css and is for styling the background of each topic thread. In the ThreadHandler.php file, add the following code segment to the createThread() method:

#### **ThreadHandler.php**

```
...
/* This method is responsible for obtaining the thread information and storing that
information in the threads table */
    public function createThread($directory, $id, $author, $subject, $topic, $title,
$content, $topicPage, $subjectStyling)
    {
        /* Before adding the thread information into the database, first append that
information into a file called 'DiscussionWindow_(topic)_(title).php' */
        $threadfile = fopen("$directory/DiscussionWindow_$topic" . "_" . " $title.php",
"w");

        $threadPath = "$directory/DiscussionWindow_$topic" . "_" . " $title.php"; //full
thread file path
        $regularPath = "DiscussionWindow_$topic" . "_" . " $title.php"; //half thread file
path
        $threadCreated = date("m/d/Y");

        //set the thread page layout for the 'DiscussionWindow_(topic)_(title).php' page
        $threadlayout = "<?php
            session_start();
            include '../..../includes/ThreadHandler.php';
        ?>
        <html>
            <head>
                <title> $title </title>
                <meta charset = 'utf-8'/>

                <link rel = 'stylesheet' href = '../..../ThreadsCssFiles/Navbar.css'/>
                <link rel = 'stylesheet' href =
'../..../ThreadsCssFiles/DiscussionWindow.css'/>
            </head>

            <body id = '$subject' class = '$subjectStyling'>
                <!-- Navbar -->
                <ul class = 'navbar'>
```

```

</li>
    <li class = 'left-tabs'> <a href = '../..../forumhome.php'> Home </a>

    <li class = 'left-tabs'> <a href = '#'> Board </a> </li>
    <li class = 'left-tabs'> <a href = '#'> Community </a> </li>

    <?php ".$$.upic = ".$$_SESSION['pic']; ?>

    <li class = 'right-tabs'><a href = '../..../includes/logout.php'> Logout
</a></li>

    <li class = 'dropdown right-tabs'>
        <a href = '#' class = 'dropdownButton'> Hello: <?php echo
        ".$$_SESSION['username']; ?> </a>
        <div class = 'dropdown-content'>
            <a href = '#'> <?php echo \"<img src = '../..../".$$.upic'
            height='70' width='70'/>\"; ?> </a>
            <a href = '#'> Full Name: <?php echo
            ".$$_SESSION['firstname'].".\" \" ".$$_SESSION['lastname']; ?> </a>
            <a href = '#'> Email: <?php echo
            ".$$_SESSION['useremail']; ?> </a>
            <a href = '#'> Current Status: <?php echo
            ".$$_SESSION['status']; ?> </a>
            <a href = '#'> Join Date: <?php echo
            ".$$_SESSION['regdate']; ?> </a>
            <a href = '../..../userprofile.php'> Go to account settings </a>
        </div>
    </li>
</ul>

...

```

The in each of the Add\_(Subject)\_thread.php page, you want to add the following the following “\$Styler” variable and the add it to the parameters of the createThread() method:

```

Add_(Subject)_thread.php
...
<?php
    include_once '../..../includes/ThreadHandler.php';

    if(isset($_POST['submit-info']))
    {
        $Directory = "(Subject) Threads";
        $Suid = $_SESSION['userid'];
        $Author = $_SESSION['username'];
        $Subject = "(Subject)";
        $TopicSelection = "";
        $Title = $_POST['title'];
        $Content = $_POST['content'];
    }

```

```

$TopicPage = "";
$Styler = "(Class Selector)"

/* For the Topic selections */
if(isset($_POST['topics-(Subject)']))
{
    $Topic = $_POST['topics-(Subject)'];

    switch($Topic)
    {
        case '(topic1)': //if the user chooses '(topic1)' in the Topics selection
            $TopicSelection = "(topic1)";
            $TopicPage = "(topic1).php";
            break;
        case '(topic2)': //else if the user chooses '(topic2)' in the Topics
selection
            $TopicSelection = "(topic2)";
            $TopicPage = "(topic2).php";
            break;
        case '(topic3)': //else if the user chooses '(topic3)' in the Topics
selection
            $TopicSelection = "(topic3)";
            $TopicPage = "(topic3).php";
            break;
        ...(continue with more cases if there are more topic selections)
        default: //else, display an error message if the user did not choose
anything
            echo "<p style = 'color: red;'>You must choose one of the following
topic selections!</p>";
            break;
    }
}

$createFile = new ThreadHandler(); //object of the ThreadHandler class,
named 'createFile'
$createFile -> createThread($Directory, $uid, $Author, $Subject,
$TopicSelection, $Title, $Content, $TopicPage, $Styler);
}
?>
...

```

The following class selector values are listed below for the \$Styler variable:

<b>Add_(Subject)_thread.php</b>	<b>\$Styler = “(Class Selector)”</b>
Add_(Culture)_thread.php	\$Styler = "CultureThread";
Add_ScienceMath_thread.php	\$Styler = "SciMathThread";
Add_SocialScience_thread.php	\$Styler = "SoScienceThread";
Add_Tech_thread.php	\$Styler = "TechThread";
Add_News_thread.php	\$Styler = "WorldNewsThread";

Now, in the DiscussionWindow.css file, you should have the following that will set the background images of each topic thread page:

<p><b><u>DiscussionWindow.css</u></b></p> <pre> ... /*Sets the backgrounds of the Discussion Windows*/ .CultureThread {     background-image: url("http://blogs.elon.edu/gst336/files/2014/05/culture-pic.jpg");     background-repeat: no-repeat;     background-size: cover;     background-attachment: fixed; } .SciMathThread {     background-image: url("https://previews.123rf.com/images/ekaart/ekaart1610/ekaart161000549/67943436- the-science-and-mathematics-abstract-background-with-circles-cube-triangles-and-a- lot-of-lines-sacre.jpg");     background-repeat: no-repeat;     background-size: cover;     background-attachment: fixed; } .SoScienceThread {     background-image: url("https://www.yesmagazine.org/issues/america-the- remix/images-for-the-spring-2010-issue/conversation-by-robert-woodstock/image");     background-repeat: no-repeat;     background-size: cover;     background-attachment: fixed; } .TechThread {     background-image: url("https://ak0.picdn.net/shutterstock/videos/9597380/thumb/1.jpg");     background-repeat: no-repeat;     background-size: cover; </pre>
---

```

    background-attachment: fixed;
}
.WorldNewsThread
{
    background-image:
url("https://ak7.picdn.net/shutterstock/videos/8558317/thumb/1.jpg");
    background-repeat: no-repeat;
    background-size: cover;
    background-attachment: fixed;
}
...

```

6. Now you want to fix the comment box of the thread pages. When user changes the size of the browser, for some reason, the comment box does not stay in the parent div it is contained in. You want that comment box to stay inside the parent div whenever the user changes the size of the browser. In the DiscussionWindow.css, add the following:

#### **DiscussionWindow.css**

```

...
    /* Style the comment window */
    .commentWindow
    {
        display: flex; /* This will keep the divs in the commentWindow the same size
and in a single row */
    }
    .commentWindow .styleButtons
    {
        margin-left: 25%;
    }
    .commentWindow .styleButtons button
    {
        border: none;
        border-radius: 3px;
        border-style: inset;
        padding: 8px;

        cursor: pointer;
    }
    .commentWindow .commentField
    {
        text-align: center;
    }
    .commentWindow .commentField textarea
    {

```

```
border-radius: 5px;
border-style: inset;

resize: none;

font-family: 'Noticia Text', serif;
}
.commentWindow .commentField input[type=submit]
{
    font-family: 'Noticia Text', serif;

padding: 10px;

cursor: pointer;

border: none;
border-radius: 5px;
border-style: inset;
}
.commentWindow .styleButtons, .commentWindow .commentField
{
    background-color: lightgrey;

border-radius: 5px;
border-style: inset;

padding: 15px;
}
```

...



**7/31/19**

Again, once again, after a long break from this project, you have to now continue and finish up customizing the page and fixing some bugs. You cannot spend so much time on this project as you've already spent almost 8 months on it. You need to put something in your portfolio.

7. Next you want to add the code that shows the user the all threads they have created in the userprofile.php page. In the ThreadHandler.php and userprofile.php files, add the following:

**ThreadHandler.php**

```
<?php
include_once 'database.php';

/* This class will handle the processes of both creating the topic thread of a
particular subject
and displaying it on the forumhome.php and the subject pages */
class ThreadHandler
{
    private $db;

    //SQL variables
    private $sql1; //for the createThread() method. For storing the thread
information in the thread table.
    private $sql2; //for the displayThread() method. For obtaining the thread
information from the thread table and displaying it in their respective topic
thread file
    private $sql3; //for the createReply() method. For obtaining the user
comment and storing it in the replies table
    private $sql4; //for the displayReply() method. For obtaining the user
comment from the replies table and displaying it in the respective topic thread
    private $sql5; //for the displayThread() method. For counting how many
replies were made to a particular topic tread
    private $sql6; //for the obtainTopicThread() method. For obtaining the last
topic thread made to a particular subject
    private $sql7; //for the showThreads() method. Shows the threads that
a particular user created

    public function __construct()
    {
        $this -> db = new Connection();
        $this -> db = $this -> db -> connect();
    }

    /* This method is responsible for obtaining the thread information and
storing that information in the threads table */
```

```

        public function createThread($directory, $id, $author, $subject, $topic,
        $title, $content, $topicPage, $subjectStyling)
        {
            /* Before adding the thread information into the database, first append
            that information into a file called 'DiscussionWindow_(topic)_(title).php' */
            $threadfile = fopen("$directory/DiscussionWindow_$topic" . "_" . "
            $title.php", "w");

            $threadPath = "$directory/DiscussionWindow_$topic" . "_" . "
            $title.php"; //full thread file path
            $regularPath = "DiscussionWindow_$topic" . "_" . " $title.php"; //half
            thread file path
            $threadCreated = date("m/d/Y");

            //set the thread page layout for the
            'DiscussionWindow_(topic)_(title).php' page
            $threadlayout = "<?php
            session_start();
            include '../..../includes/ThreadHandler.php';
            ?>
            <html>
            <head>
            <title> $title </title>
            <meta charset = 'utf-8'/>

            <link rel = 'stylesheet' href =
            '../..../ThreadsCssFiles/Navbar.css'/>
            <link rel = 'stylesheet' href =
            '../..../ThreadsCssFiles/DiscussionWindow.css'/>
            </head>

            <body id = '$subject' class = '$subjectStyling'>
            <!-- Navbar -->
            <ul class = 'navbar'>
            <li class = 'left-tabs'> <a href = '../..../forumhome.php'>
            Home </a> </li>
            <li class = 'left-tabs'> <a href = '#'> Board </a> </li>
            <li class = 'left-tabs'> <a href = '#'> Community </a> </li>

            <?php ".$$_SESSION['pic']; ?>

            <li class = 'right-tabs'><a href = '../..../includes/logout.php'>
            Logout </a></li>
            <li class = 'dropdown right-tabs'>
            <a href = '#' class = 'dropdownButton'> Hello: <?php echo
            ".$$_SESSION['username']; ?> </a>

```

```

        <div class = 'dropdown-content'>
            <a href = '#'> <?php echo \"<img src =
'../..../\".$\".\"upic' height='70' width='70'/>\"; ?> </a>
            <a href = '#'> Full Name: <?php echo
\".$\"._SESSION['firstname'].\".\".\".\".$\"._SESSION['lastname']; ?> </a>
            <a href = '#'> Email: <?php echo
\".$\"._SESSION['useremail']; ?> </a>
            <a href = '#'> Current Status: <?php echo
\".$\"._SESSION['status']; ?> </a>
            <a href = '#'> Join Date: <?php echo
\".$\"._SESSION['regdate']; ?> </a>
            <a href = '../..../userprofile.php'> Go to account
settings </a>
        </div>
    </li>
</ul>

<!-- Thread window -->
<div class = 'threadWindow'>
    <div class = 'userInfo'>
        <div id = 'userPicture'> <?php echo \"<img src =
'../..../\".$\".\"upic' height='70' width='70'/>\"; ?> </div>
        <h2> <?php echo \".$\"._SESSION['username']; ?>
    </h2>
        <h4> Join date: <?php echo \".$\"._SESSION['regdate'];
?> </h4>
        <p> Current Status: <?php echo
\".$\"._SESSION['status']; ?> </p>
    </div>
    <div class = 'threadContent'>
        <h1> $title </h1>
        <h4> Thread created: $threadCreated </h4>
        <h5> Topic: $topic </h5>
        <div class = 'underline' style = 'border-bottom: 1px solid
black;'></div>
        <p>
            $content
        </p>
    </div>
</div>

<br/><br/><br/>

<!-- Replies window -->
<?php
\".$\".replies = new ThreadHandler();

```

```

        ".$".Directory = \"$threadPath\";

        ".$" .replies -> displayReply( ".$".Directory);
    ?>

<br/><br/><br/>

<!-- Comment window -->
<div class = 'commentWindow'>
    <div class = 'styleButtons'>
        <button id = 'bold' onclick = 'styleText(\"<b>\",
        \"</b>\")'> <abbr title = 'Bold'> <b>B</b> </abbr> </button> <br/> <br/>
        <button id = 'italic' onclick = 'styleText(\"<em>\",
        \"</em>\")'> <abbr title = 'Italic'> <em>i</em> </abbr> </button> <br/> <br/>
        <button id = 'underline' onclick = 'styleText(\"<u>\",
        \"</u>\")'> <abbr title = 'Underline'> <u>U</u> </abbr> </button> <br/> <br/>
        <button id = 'subscript' onclick = 'styleText(\"<sub>\",
        \"</sub>\")'> <abbr title = 'Subscript'> A<sub>2</sub> </abbr> </button>
        <br/> <br/>
        <button id = 'superscript' onclick = 'styleText(\"<sup>\",
        \"</sup>\")'> <abbr title = 'Superscript'> A<sup>2</sup> </abbr> </button>
        <br/> <br/>
        <button id = 'quote' onclick =
        'styleText(\"<blockquote>\", \"</blockquote>\")'> <abbr title = 'Quote'> \"Q\"
        </abbr> </button>
    </div>
    <div class = 'commentField'>
        <form method = 'POST' action = '$regularPath'>
            <textarea name = 'commentbox' id = 'commentbox'
            rows = '15' cols = '90'></textarea> <br/>
            <input type = 'submit' name = 'submit-comment'
            value = 'Submit comment'>
        </form>
    <?php
        if(isset( ".$"._POST['submit-comment']))
        {
            ".$".Replies = new ThreadHandler();

            ".$".uid = ".$"._SESSION['userid']; //id of the
            user who made the comment/reply
            ".$".uname = ".$"._SESSION['username'];
            //username of the user who made the comment/reply
            ".$".upic = ".$"._SESSION['pic']; //profile pic of
            user who made the comment/reply

```

user is replying to

the thread file

was created

reply

```

"."$.Title = \"$.title\"; //title of the thread who the
"."$.Directory = \"$.threadPath\"; //full file path of
"."$.replyCreated = date('m/d/Y'); //date the reply
"."$.Reply = "."$. _POST['commentbox']; //the

```

```

"."$.Replies -> createReply("."$.uid,
"."$.uname, "."$.upic, "."$.Title, "."$.Directory, "."$.replyCreated,
"."$.Reply);

```

```

}
?>
</div>
</div>

```

```

<script src =
'../../ThreadsJSFiles/DiscussionWindow.js'></script>
</body>
</html>";

```

```

fwrite($threadfile, $threadlayout);

```

```

/* Now add the thread information to the database */
$this -> sql1 = "INSERT INTO threads(user_id, thread_author,
thread_subject, thread_topic, thread_title, thread_content, thread_file,
thread_creation)
VALUES('$.id', '$author', '$subject', '$topic', '$title', '$content',
'$.threadPath', '$threadCreated');"

```

```

$stmt1 = $this -> db -> stmt_init();

```

```

if(!$stmt1 -> prepare($this -> sql1))
{
    echo "<script> alert('SQL ERROR: ". $stmt1 -> error . "'); </script>";
}
else
{
    $stmt1 -> execute();
    header("Location: $topicPage?threadcreation=success");
}
}

```

```

/* This method is responsible for displaying the topic thread on the
forumhome.php page and the subject pages */
public function displayThread($subject, $topic)
{
    /* This query is for displaying each topic thread of a particular subject */
    $this -> sql2 = "SELECT thread_author, thread_subject, thread_topic,
thread_title, thread_file
                    FROM threads
                    WHERE thread_subject='$subject' AND
thread_topic='$topic'";

    $stmt2 = $this -> db -> stmt_init();

    if(!$stmt2 -> prepare($this -> sql2))
    {
        echo "<script> alert('SQL ERROR: " . $stmt2 -> error . "'); </script>";
    }
    else
    {
        $stmt2 -> execute();
        $result1 = $stmt2 -> get_result();

        if($result1 -> num_rows > 0)
        {
            while($row1 = $result1 -> fetch_assoc()) //stores the results of the
thread table query in an associative array and stores it in a variable called
'$row1'
            {
                $dir = $row1['thread_file']; //obtains the path of the topic thread at
hand

                /* This query is for obtaining the number of replies made to a
particular topic thread */
                $this -> sql5 = "SELECT COUNT(replies_id) AS 'count' FROM
replies WHERE thread_file='$dir'"; //counts how many replies were made to a
topic thread

                $stmt5 = $this -> db -> stmt_init();

                if(!$stmt5 -> prepare($this -> sql5))
                {
                    echo "<script> alert('SQL ERROR: " . $stmt5 -> error . "');
</script>";
                }
                else
                {

```

```

$stmt5 -> execute();
$result2 = $stmt5 -> get_result();

while($row2 = $result2 -> fetch_assoc()) //stores the results of the
replies table query in an associative array and stores it in a variable called
'$row2'
    {
        echo "<tr>
            <td><a href = '". $row1['thread_file'] ."'>" .
$row1['thread_title'] . "</a></td>
            <td>" . $row2['count'] . "</td>
            <td> By: <em>" . $row1['thread_author'] . "</em> </td>
            </tr>";
    }
}
}
}
else
{
    echo "<b id = 'warningMessage'> There are no current threads posted.
</b>";
}
}
}

/* This method creates the comment a user made on a particular topic
thread */
public function createReply($id, $username, $pic, $title, $directory,
$dateCreated, $comment)
{
    /* Store the comment in the database with the respective thread file */
    if(empty($comment))
    {
        echo "<p id = 'errorMessage'> You must enter a comment! </p>";
    }
    else
    {
        $this -> sql3 = "INSERT INTO replies(user_id, user_username,
profile_pic, thread_title, thread_file, replies_creation, replies_content)
VALUES('$id', '$username', '$pic', '$title' , '$directory',
'$dateCreated', '$comment')";

        $stmt3 = $this -> db -> stmt_init();

        if(!$stmt3 -> prepare($this -> sql3))
        {

```

```

        echo "<script> alert('SQL ERROR: ". $stmt3 -> error . "'); </script>";
    }
    else
    {
        $stmt3 -> execute();
        echo "<script> location.assign(location.href); </script>"; //refreshes
current thread page to show replies as they are created
    }
}
}

/* This method displays the comments that users made on a particular topic
thread */
public function displayReply($directory)
{
    /* Get only the replies of that particular topic thread */
    $this -> sql4 = "SELECT user_username, user_status, profile_pic,
thread_file, replies_creation, replies_content
FROM replies
WHERE thread_file = '$directory'";

    $stmt4 = $this -> db -> stmt_init();

    if(!$stmt4 -> prepare($this -> sql4))
    {
        echo "<script> alert('SQL ERROR: ". $stmt4 -> error . "'); </script>";
    }
    else
    {
        $stmt4 -> execute();

        $result = $stmt4 -> get_result();

        if($result -> num_rows > 0)
        {
            while($row = $result -> fetch_assoc())
            {
                echo "<div class = 'repliesWindow'>
                    <div class = 'userInfo'>
                        <img src = \" . \"../..\" . $row['profile_pic'] . \" height='70'
width='70'> <br/>
                        <h2>\" . $row['user_username'] . \"</h2>
                        <p> Current Status: \" . $row['user_status'] . \" </p>
                    </div>
                    <div class = 'replyContent'>
                        <p>\" . $row['replies_content'] . \"</p>

```



```

        </div>
    </div> <br/><br/>";
    }
}
else
{
    echo "<b id = 'warningMessage'> There are no current replies posted.
</b>";
}
}
}
}

```

/\* This method obtains the last topic thread that was made on a particular subject \*/

```
public function obtainTopicThread($subject, $topic)
```

```
{
```

/\* This SQL query selects the most recent topic thread of a particular subject.

The most recent one has a high thread id number, thus the thread title for that

thread must also be high as well \*/

```
$this -> sql6 = "SELECT MAX(thread_id), thread_author,
thread_subject, thread_topic, MAX(thread_title), thread_file,
MAX(thread_creation)
```

```
FROM threads WHERE thread_subject='$subject' AND
thread_topic='$topic';";
```

```
$stmt6 = $this -> db -> stmt_init();
```

```
if(!$stmt6 -> prepare($this -> sql6))
```

```
{
```

```
    echo "<script> alert('SQL ERROR: ". $stmt6 -> error . "'); </script>";
```

```
}
```

```
else
```

```
{
```

```
    $stmt6 -> execute();
```

```
$result = $stmt6 -> get_result();
```

```
while($row = $result -> fetch_assoc())
```

```
{
```

```
    if($row['MAX(thread_id)'] == null && $row['thread_author'] == null
&& $row['thread_subject'] == null
```

```
        && $row['thread_topic'] == null && $row['MAX(thread_title)'] ==
null && $row['thread_file'] == null
```

```
        && $row['MAX(thread_creation)'] == null)
```

```

        {
            echo "<p> There are no threads posted </p>";
        }
        else
        {
            echo "<p>
                <a href = 'Subjects/$subject/'. $row['thread_file'] .'"> ".
$row["MAX(thread_title)"] ." </a> <br>
                by <em> ". $row['thread_author'] ." </em> <br>
                Created on: ". $row["MAX(thread_creation)"] ."
                </p>";
        }
    }
}

/* This method shows the threads that a user created */
public function showThreads($id)
{
    $this -> sql7 = "SELECT thread_title, thread_subject, thread_topic,
thread_creation, thread_file FROM threads WHERE user_id='$id'";

    $stmt7 = $this -> db -> stmt_init();

    if(!$stmt7 -> prepare($this -> sql7))
    {
        echo "<script> alert('SQL ERROR: ". $stmt7 -> error ."'");
    </script>";
    }
    else
    {
        $stmt7 -> execute();

        $result = $stmt7 -> get_result();

        if($result -> num_rows == 0)
        {
            echo "<script>
                var threadwindow =
document.getElementById('threadsCreated');
                threadwindow.style.display = 'none';

                var Notice = document.createElement('p');
                var textnode = document.createTextNode('No threads for
this topic have been posted');

```

```

        var notice = Notice.appendChild(textnode);

        document.getElementById('user-
posts').appendChild(notice);
    </script>";
    }
    else
    {
        while($row = $result -> fetch_assoc())
        {
            echo "<tr>
                <td><a href =
'Subjects/' . $row['thread_subject'] . '/' . $row['thread_file'] . "'>". $row['thre
ad_title'] . "</a></td>
                <td>". $row['thread_subject'] . "</td>
                <td>". $row['thread_topic'] . "</td>
                <td>". $row['thread_creation'] . "</td>
            </tr>";
        }
    }
}
}
}
?>

```

### userprofile.php

```

...
    <!-- List of threads you last made -->
    <div id = "user-posts">
        <table>
            <tr>
                <th> Threads you have created: </th>
                <th> Subject of your threads: </th>
                <th> Topic of your threads </th>
                <th> Date of threads posted: </th>
            </tr>
            <?php
                include 'includes/ThreadHandler.php';

                $showThreads = new ThreadHandler();
                $showThreads -> showThreads($_SESSION['userid']);
            ?>
        </table>
    </div>
...

```

8. Next, you want to handle the private messaging system, where users can send each other private messages. Create the table called “MessageInbox”, and this will contain all the private messages, and well as the usernames of members who sent the messages:

```
CREATE TABLE MessageInbox
(
    user_id int NOT NULL, /*id of user who is sending the message*/
    user_id2 int NOT NULL, /*id of user who will be receiving the message*/
    messageTitle varchar(100), /*title of the private message*/
    messageContent text NOT NULL, /*content of the private message*/
    messageFrom varchar(100) NOT NULL, /*username of user who sent a private message*/
    messageTo varchar(100) NOT NULL, /*username of the user who the private message was sent to*/
    date_sent varchar(100) NOT NULL /*the date the private message was created*/
);
```

Now, in both userProfile.php and ProfileHandler.php pages, add the following:

**ProfileHandler.php**

```
<?php
    include_once 'database.php';

    /* This class updates user information such as name, email and profile
    picture */
    class ProfileHandler
    {
        private $db; //to connect to the discussionforum database

        /* These variables are reserved for the uploadPics() method */
        private $sql1; //for uploading profile pictures
        private $photo = ""; //obtains the photo to be uploaded (initialize to nothing
        until user selects a picture to upload)
        private $imageFileType = ""; //obtain the type of file (".jpg", ".png", etc...)
        (initialize to nothing until user selects a picture to upload)
        private $folder; //the folder to store the uploaded pictures
        private $check; //checks the size of the image

        /* These variables are reserved for the updateStatus() method */
        private $sql2; //for updating user information
        private $newUsername = ""; //sets new user name
        private $newPassword = ""; //sets new user password
        private $newEmail = ""; //sets new user email
        private $setStatus = ""; //sets new user status

        /* These variables are reserved for the PrivateMessage() method */
        private $sql3; //for checking if a username the user sending the PM
        actually exists in the accounts table
```

```
private $sql4; //if username exists, add the information into the  
MessageBox table
```

```
/* These variables are reserved for the DisplayMessages1() method */  
private $sql5; //for displaying the messages that the current logged in  
user sent to other members
```

```
/* These variables are reserved for the DisplayMessage2() method */  
private $sql6; //for displaying the messages that other members has  
sent to the current logged in user
```

```
/* Initialize connection to the discussionforum database upon creating the  
object of this class */
```

```
public function __construct()  
{  
    $this -> db = new Connection();  
    $this -> db = $this -> db -> connect();  
}
```

```
/* This method uploads new profile pictures */
```

```
/* This will update the profile_pic column in both the accounts and replies  
table */
```

```
public function uploadPics($id)  
{  
    $this -> folder = "ProfilePics/";  
    $this -> photo = $this -> folder . basename($_FILES["getFile"]["name"]);  
    //obtains the photo to be updated and store them in the ProfilePics folder  
    $this -> imageFileType = strtolower(pathinfo($this -> photo,  
    PATHINFO_EXTENSION)); //obtain the path of the type of file (".jpg",  
    ".png", etc...)  
    $this -> check = getimagesize($_FILES["getFile"]["tmp_name"]);  
    //checks the size of the image
```

```
    //check if the file is a real image or not
```

```
    if($this -> check !== false)
```

```
    {
```

```
        //if the image file is not jpg, jpeg, png and gif files
```

```
        if($this -> imageFileType != "jpg" && $this -> imageFileType !=  
        "png" && $this -> imageFileType != "jpeg" && $this -> imageFileType !=  
        "gif")
```

```
        {
```

```
            echo "<p id = 'errMessage'> Sorry, only JPG, JPEG, PNG, and GIF  
files are allowed! </p>";
```

```
        }
```

```
        else //else if the image file is a jpg, jpeg, png, and/or gif
```

```
        {
```

```

        //if the file is too large
        if($_FILES["getFile"]["size"] > 500000)
        {
            echo "<p id = 'errMessage'> Sorry, your file is too big to upload!
</p>";
        }
        else //else if file is not too large
        {
            $pic = $this -> photo; //this variable is just a placeholder to hold the
photo variable

            //upload the image
            if(move_uploaded_file($_FILES["getFile"]["tmp_name"], $this ->
photo))
            {
                /* update the profile_pic columns in both the accounts and replies
table */
                $this -> sql1 = "UPDATE accounts SET profile_pic='$pic'
WHERE user_id='$id'";
                $this -> sql1 .= "UPDATE replies SET profile_pic='$pic' WHERE
user_id='$id'";

                if(!$this -> db -> multi_query($this -> sql1))
                {
                    echo "<script> alert('Multi Query failed: " . $this -> db -> error .
"); </script>";
                }
                else
                {
                    echo "<p id = 'succMessage'> Your picture was successfully
uploaded! Please wait 2 seconds for new picture to show up! </p>";
                    $_SESSION['pic'] = $pic; //update the profile picture session
variable to have it change right away after uploading new pic
                    header("refresh: 2"); //refresh the current page for 2 seconds
                }
            }
            //if the image could not be uploaded
            else
            {
                echo "<p id = 'errMessage'> Sorry, there was a problem uploading
your image file </p>";
            }
        }
    }
}
else

```

```

        {
            echo "<p id = 'errMessage'> The image you selected is not a file!
        </p>";
        }
    }

    /* This method updates user information such as name, email, password,
    etc */
    public function updateInfo($userid, $newUsername, $newPassword,
    $newEmail, $setStatus)
    {
        //if the username field is not empty, but the others are, change only the
        username
        if(empty($newPassword) && empty($newEmail) && empty($setStatus))
        {
            $this->newUsername = $newUsername; //private $newUsername =
            updateInfo($newUsername)

            $id = intval($userid);

            $this->sql2 = "UPDATE accounts SET user_username='$this-
            >newUsername' WHERE user_id='$id'";
            $this->sql2 .= "UPDATE replies SET user_username='$this-
            >newUsername' WHERE user_id='$id'";
            $this->sql2 .= "UPDATE threads SET thread_author='$this-
            >newUsername' WHERE user_id='$id'";

            if(!$this->db->multi_query($this->sql2))
            {
                echo "<script> alert('Multi Query failed! : " . $this->db->error . "');
            </script>";
            }
            else
            {
                $_SESSION['username'] = $this->newUsername; //change the
                username session variable immediately after updating
                echo "<p id = 'succMessage'> Username has been changed! Please
                wait 2 seconds for new information to be updated! </p>";
                header("refresh: 2"); //refresh the current page for 2 seconds
            }
        }
        //if the password field is not empty, but the others are, change only the
        password
        else if(empty($newUsername) && empty($newEmail) &&
        empty($setStatus))

```

```

    {
        $this -> newPassword = $newPassword; //private $newPassword =
updateInfo($newPassword)
        $hashedPwd = password_hash($this -> newPassword,
PASSWORD_DEFAULT); //must hash the password to prevent hackers from
seeing user password

        $id = intval($userid);

        $this -> sql2 = "UPDATE accounts SET user_password='$hashedPwd'
WHERE user_id='$id'";

        if(!$this -> db -> query($this -> sql2))
        {
            echo "<script> alert('SQL ERROR: " . $this -> db -> error . "');
</script>";
        }
        else
        {
            echo "<p id = 'succMessage'> Password has been changed! </p>";
        }
    }
    //if the email field is not empty, but the others are, change only the email
    else if(empty($newUsername) && empty($newPassword) &&
empty($setStatus))
    {
        $this -> newEmail = $newEmail; //private $newEmail =
updateInfo($newEmail)

        $id = intval($userid);

        $this -> sql2 = "UPDATE accounts SET user_email='$this->newEmail'
WHERE user_id='$id'";

        if(!$this -> db -> query($this -> sql2))
        {
            echo "<script> alert('SQL ERROR: " . $this -> db -> error . "');
</script>";
        }
        else
        {
            $_SESSION['useremail'] = $this -> newEmail; //change the user
email session variable immediately after updating
            echo "<p id = 'succMessage'> Email has been changed! Please wait 2
seconds for information to be uploaded! </p>";
            header("refresh: 2"); //refresh the current page for 2 seconds

```



```

    }
  }
  //if the status is not empty, but the others are, change only the status
  else if(empty($newUsername) && empty($newPassword) &&
empty($newEmail))
  {
    $this -> setStatus = $setStatus; //private $setStatus =
updateInfo($setStatus)

    $id = intval($userid);

    $this -> sql2 = "UPDATE accounts SET user_status='$this->setStatus'
WHERE user_id='$id'";
    $this -> sql2 .= "UPDATE replies SET user_status='$this->setStatus'
WHERE user_id='$id'";

    if(!$this -> db -> multi_query($this -> sql2))
    {
      echo "<script> alert('Multi Query failed!: " . $this -> db -> error . "');
</script>";
    }
    else
    {
      $_SESSION['status'] = $this -> setStatus; //change the user status
session variable immediately after updating
      echo "<p id = 'succMessage'> Status has been updated! Please wait 2
seconds for information to be uploaded! </p>";
      header("refresh: 2"); //refresh the current page for 2 seconds
    }
  }
  //if only the username field is blank, but the rest is not, change the
password, email and status
  else if(empty($newUsername))
  {
    $this -> newPassword = $newPassword;
    $this -> newEmail = $newEmail;
    $this -> setStatus = $setStatus;

    $hashedPwd = password_hash($this -> newPassword,
PASSWORD_DEFAULT);

    $id = intval($userid);

    $this -> sql2 = "UPDATE accounts SET user_password='$hashedPwd',
user_email='$this->newEmail', user_status='$this->setStatus' WHERE
user_id='$id'";

```

```

        $this -> sql2 .= "UPDATE replies SET user_status='$this->setStatus'
WHERE user_id='$id';";

        if(!$this -> db -> multi_query($this -> sql2))
        {
            echo "<script> alert('Multi Query failed!: " . $this -> db -> error . "');
</script>";
        }
        else
        {
            $_SESSION['useremail'] = $this -> newEmail; //change the user
email session variable immediately after updating
            $_SESSION['status'] = $this -> setStatus; //change the user status
session variable immediately after updating
            echo "<p id = 'succMessage'> The password and email have been
changed and the status has been updated! Please wait 2 seconds for information
to be uploaded! </p>";
            header("refresh: 2"); //refresh the current page for 2 seconds
        }
    }
    //if only the password field is blank, but the rest is not, change the
username, email and status
    else if(empty($newPassword))
    {
        $this -> newUsername = $newUsername;
        $this -> newEmail = $newEmail;
        $this -> setStatus = $setStatus;

        $id = intval($userid);

        $this -> sql2 = "UPDATE accounts SET user_username='$this-
>newUsername', user_email='$this->newEmail', user_status='$this->setStatus'
WHERE user_id='$id';";
        $this -> sql2 .= "UPDATE replies SET user_username='$this-
>newUsername', user_status='$this->setStatus' WHERE user_id='$id';";
        $this -> sql2 .= "UPDATE threads SET thread_author='$this-
>newUsername' WHERE user_id='$id';";

        if(!$this -> db -> multi_query($this -> sql2))
        {
            echo "<script> alert('Multi Query failed!: " . $this -> db -> error . "');
</script>";
        }
        else
        {

```

```

        $_SESSION['username'] = $this -> newUsername; //change the
username session variable immediately after updating
        $_SESSION['useremail'] = $this -> newEmail; //change the user
email session variable immediately after updating
        $_SESSION['status'] = $this -> setStatus; //change the user status
session variable immediately after updating
        echo "<p id = 'succMessage'> The username and email have been
changed and the status has been updated! Please wait 2 seconds for information
to be uploaded! </p>";
        header("refresh: 2"); //refresh the current page for 2 seconds
    }
}
//if only the email field is blank, but the rest is not, change the username,
password and status
else if(empty($newEmail))
{
    $this -> newUsername = $newUsername;
    $this -> newPassword = $newPassword;
    $this -> setStatus = $setStatus;

    $hashedPwd = password_hash($this -> newPassword,
PASSWORD_DEFAULT);

    $id = intval($userid);

    $this -> sql2 = "UPDATE accounts SET user_username='$this-
>newUsername', user_password='$hashedPwd', user_status='$this->setStatus'
WHERE user_id='$id'";
    $this -> sql2 .= "UPDATE replies SET user_username='$this-
>newUsername', user_status='$this->setStatus' WHERE user_id='$id'";
    $this -> sql2 .= "UPDATE threads SET thread_author='$this-
>newUsername' WHERE user_id='$id'";

    if(!$this -> db -> multi_query($this -> sql2))
    {
        echo "<script> alert('Multi Query failed!: " . $this -> db -> error . "');
</script>";
    }
    else
    {
        $_SESSION['username'] = $this -> newUsername; //change the
username session variable immediately after updating
        $_SESSION['status'] = $this -> setStatus; //change the user status
session variable immediately after updating

```

```

        echo "<p id = 'succMessage'> The username and password have been
changed and the status has been updated! Please wait 2 seconds for information
to be uploaded! </p>";
        header("refresh: 2"); //refresh the current page for 2 seconds
    }
}
//if only the status field is blank, but the rest is not, change the username,
password and email
else if(empty($setStatus))
{
    $this -> newUsername = $newUsername;
    $this -> newEmail = $newEmail;
    $this -> newPassword = $newPassword;

    $hashedPwd = password_hash($this -> newPassword,
PASSWORD_DEFAULT);

    $id = intval($userid);

    $this -> sql2 = "UPDATE accounts SET user_username='$this-
>newUsername', user_email='$this->newEmail', user_password='$hashedPwd'
WHERE user_id='$id'";
    $this -> sql2 .= "UPDATE replies SET user_username='$this-
>newUsername' WHERE user_id='$id'";
    $this -> sql2 .= "UPDATE threads SET thread_author='$this-
>newUsername' WHERE user_id='$id'";

    if(!$this -> db -> multi_query($this -> sql2))
    {
        echo "<script> alert('Multi Query failed!: " . $this -> db -> error . "');
</script>";
    }
    else
    {
        $_SESSION['username'] = $this -> newUsername; //change the
username session variable immediately after updating
        $_SESSION['useremail'] = $this -> newEmail; //change the user
email session variable immediately after updating
        echo "<p id = 'succMessage'> The username, password and email
have been changed! Please wait 2 seconds for information to be uploaded!
</p>";
        header("refresh: 2"); //refresh the current page for 2 seconds
    }
}
//change only the username and password if the email and status fields
are blank

```

```

else if(empty($newEmail) && empty($setStatus))
{
    $this->newUsername = $newUsername;
    $this->newPassword = $newPassword;

    $hashedPwd = password_hash($this->newPassword,
PASSWORD_DEFAULT);

    $id = intval($userid);

    $this->sql2 = "UPDATE accounts SET user_username='$this-
>newUsername', user_password='$hashedPwd' WHERE user_id='$id'";
    $this->sql2 .= "UPDATE replies SET user_username='$this-
>newUsername' WHERE user_id='$id'";
    $this->sql2 .= "UPDATE threads SET thread_author='$this-
>newUsername' WHERE user_id='$id'";

    if(!$this->db->multi_query($this->sql2))
    {
        echo "<script> alert('Multi Query failed!: " . $this->db->error . "');
</script>";
    }
    else
    {
        $_SESSION['username'] = $this->newUsername; //change the
username session variable immediately after updating
        echo "<p id = 'succMessage'> The username and password has been
changed! Please wait 2 seconds for information to be uploaded! </p>";
        header("refresh: 2"); //refresh the current page for 2 seconds
    }
}
//change only the username and email if the password and status fields
are empty
else if(empty($newPassword) && empty($setStatus))
{
    $this->newUsername = $newUsername;
    $this->newEmail = $newEmail;

    $id = intval($userid);

    $this->sql2 = "UPDATE accounts SET user_username='$this-
>newUsername', user_email='$this->newEmail' WHERE user_id='$id'";
    $this->sql2 .= "UPDATE replies SET user_username='$this-
>newUsername' WHERE user_id='$id'";
    $this->sql2 .= "UPDATE threads SET thread_author='$this-
>newUsername' WHERE user_id='$id'";

```

```

        if(!$this -> db -> multi_query($this -> sql2))
        {
            echo "<script> alert('Multi Query failed!: " . $this -> db -> error . "');
</script>";
        }
        else
        {
            $_SESSION['username'] = $this -> newUsername; //change the
username session variable immediately after updating
            $_SESSION['useremail'] = $this -> newEmail; //change the user
email session variable immediately after updating
            echo "<p id = 'succMessage'> The username and email has been
changed! Please wait 2 seconds for information to be uploaded! </p>";
            header("refresh: 2"); //refresh the current page for 2 seconds
        }
    }
    //change only the username and status if the password and email fields
are empty
    else if(empty($newPassword) && empty($newEmail))
    {
        $this -> newUsername = $newUsername;
        $this -> setStatus = $setStatus;

        $id = intval($userid);

        $this -> sql2 = "UPDATE accounts SET user_username='$this-
>newUsername', user_status='$this->setStatus' WHERE user_id='$id'";
        $this -> sql2 .= "UPDATE replies SET user_username='$this-
>newUsername', user_status='$this->setStatus' WHERE user_id='$id'";
        $this -> sql2 .= "UPDATE threads SET thread_author='$this-
>newUsername' WHERE user_id='$id'";

        if(!$this -> db -> multi_query($this -> sql2))
        {
            echo "<script> alert('Multi Query failed!: " . $this -> db -> error . "');
</script>";
        }
        else
        {
            $_SESSION['username'] = $this -> newUsername; //change the
username session variable immediately after updating
            $_SESSION['status'] = $this -> setStatus; //change the user status
session variable immediately after updating
            echo "<p id = 'succMessage'> The username and status has been
changed! Please wait 2 seconds for information to be uploaded! </p>";

```

```

        header("refresh: 2"); //refresh the current page for 2 seconds
    }
}
//change only the password and email if the username and status fields
are empty
else if(empty($newUsername) && empty($newEmail))
{
    $this -> newPassword = $newPassword;
    $this -> newEmail = $newEmail;

    $hashedPwd = password_hash($this -> newPassword,
PASSWORD_DEFAULT);

    $id = intval($userid);

    $this -> sql2 = "UPDATE accounts SET user_password='$hashedPwd',
user_email='$this->newEmail' WHERE user_id='$id'";

    if(!$this -> db -> query($this -> sql2))
    {
        echo "<script> alert('SQL ERROR: " . $this -> db -> error . "');
</script>";
    }
    else
    {
        $_SESSION['useremail'] = $this -> newEmail; //change the user
email session variable immediately after updating
        echo "<p id = 'succMessage'> The password and email has been
changed! Please wait 2 seconds for information to be uploaded! </p>";
        header("refresh: 2"); //refresh the current page for 2 seconds
    }
}
//change only the password and status if the username and email fields
are empty
else if(empty($newUsername) && empty($newEmail))
{
    $this -> newPassword = $newPassword;
    $this -> setStatus = $setStatus;

    $hashedPwd = password_hash($this -> newPassword,
PASSWORD_DEFAULT);

    $id = intval($userid);

    $this -> sql2 = "UPDATE accounts SET user_password='$hashedPwd',
user_status='$this->setStatus' WHERE user_id='$id'";

```

```

        $this -> sql2 .= "UPDATE replies SET user_status='$this->setStatus'
WHERE user_id='$id'";

        if(!$this -> db -> multi_query($this -> sql2))
        {
            echo "<script> alert('Multi Query failed!: " . $this -> db -> error . "');
</script>";
        }
        else
        {
            $_SESSION['status'] = $this -> setStatus; //change the user status
session variable immediately after updating
            echo "<p id = 'succMessage'> The password and status has been
changed! Please wait 2 seconds for information to be uploaded! </p>";
            header("refresh: 2"); //refresh the current page for 2 seconds
        }
    }
    //change only the email and status if the username and password fields
are empty
    else if(empty($newUsername) && empty($newPassword))
    {
        $this -> newEmail = $newEmail;
        $this -> setStatus = $setStatus;

        $id = intval($userid);

        $this -> sql2 = "UPDATE accounts SET user_email='$this-
>newEmail', user_status='$this->setStatus' WHERE user_id='$id'";
        $this -> sql2 .= "UPDATE replies SET user_status='$this->setStatus'
WHERE user_id='$id'";

        if(!$this -> db -> multi_query($this -> sql2))
        {
            echo "<script> alert('Multi Query failed!: " . $this -> db -> error . "');
</script>";
        }
        else
        {
            $_SESSION['useremail'] = $this -> newEmail; //change the user
email session variable immediately after updating
            $_SESSION['status'] = $this -> setStatus; //change the user status
session variable immediately after updating
            echo "<p id = 'succMessage'> The email and status has been changed!
Please wait 2 seconds for information to be uploaded! </p>";
            header("refresh: 2"); //refresh the current page for 2 seconds
        }
    }

```



```

    }
}

/* This method will add private messages to the MessageInbox table */
public function PrivateMessage($id, $messageTitle, $messageContent,
$messageTo, $messageFrom, $dateCreated)
{
    if(empty($messageTitle) || empty($messageContent) ||
empty($messageTo))
    {
        echo "<p style = 'color: red; font-family: serif; text-align: center;*>
Do not leave anything blank! </p>";
    }
    else
    {
        //check if the username the user sending the message has entered
exists in the accounts table
        $this -> sql3 = "SELECT * FROM accounts WHERE
user_username='$messageTo'";
        $result = $this -> db -> query($this -> sql3);

        if($result -> num_rows == 0)
        {
            echo "<p style = 'color: red; font-family: serif; text-align:
center;*> There is no such member called \"'\". $messageTo.\"'\". Make sure
to enter member username as exactly how it looks. </p>";
        }
        else //else if username does exist, obtain the user_id of that member
of who the message will be sent to
        {
            $row1 = $result -> fetch_assoc();
            $userid = $row1['user_id']; //obtain the id of the user who the
message will be sent to

            $this -> sql4 = "INSERT INTO messageinbox(user_id, user_id2,
messageTitle, messageContent, messageFrom, messageTo, date_sent)
VALUES(?, ?, ?, ?, ?, ?, ?)";

            $stmt = $this -> db -> stmt_init();

            if(!$stmt -> prepare($this -> sql4))
            {
                echo "<script> SQL ERROR: alert('\". $stmt -> error.\"');
</script>";
            }
            else

```

```

        {
            $stmt -> bind_param("ssssss", $id, $userid, $messageTitle,
                $messageContent, $messageFrom, $messageTo, $dateCreated);
            $stmt -> execute();

            echo "<p style = 'color: green; font-family: serif; text-align:
center;'> Message has been sent! </p>";
        }
    }
}

/* This method displays the messages the current user logged in has
sent to other members */
public function DisplayMessages1($id)
{
    $this -> sql5 = "SELECT * FROM messageinbox WHERE
user_id='$id'";

    $stmt = $this -> db -> stmt_init();

    if(!$stmt -> prepare($this -> sql5))
    {
        echo "<script> alert('".$stmt -> error."'); </script>";
    }
    else
    {
        $stmt -> execute();
        $result = $stmt -> get_result();

        if($result -> num_rows == 0)
        {
            echo "<script>
                var messagewindow =
document.getElementById('messagessend');
                messagewindow.style.display = 'none';

                var Notice = document.createElement('p');
                var textnode = document.createTextNode('You have not
sent any private messages');

                var notice = Notice.appendChild(textnode);

document.getElementById('sentmessages').appendChild(notice);
            </script>";
        }
    }
}

```

```

    }
    else
    {
        while($row = $result -> fetch_assoc())
        {
            echo "<tr>
                <td>".$row['messageTo']. "</td>
                <td>".$row['messageTitle']. "</td>
                <td>".$row['date_sent']. "</td>
                <td>".$row['messageContent']. "</td>
            </tr>";
        }
    }
}

```

**/\* This method displays the messages that other members sent to the current user logged in \*/**

```

public function DisplayMessage2($id)
{
    $this -> sql6 = "SELECT * FROM messageinbox WHERE
user_id2='$id'";

    $stmt = $this -> db -> stmt_init();

    if(!$stmt -> prepare($this -> sql6))
    {
        echo "<script> alert('".$stmt -> error()."'); </script>";
    }
    else
    {
        $stmt -> execute();
        $result = $stmt -> get_result();

        if($result -> num_rows == 0)
        {
            echo "<script>
                var messagewindow =
document.getElementById('usermessageinbox');
                messagewindow.style.display = 'none';

                var Notice = document.createElement('p');
                var textnode = document.createTextNode('No one has sent
you any private messages');

                var notice = Notice.appendChild(textnode);

```

```

document.getElementById('messageinbox').appendChild(notice);
</script>";
}
else
{
while($row = $result -> fetch_assoc())
{
echo "<tr>
    <td>".$row['messageFrom']. "</td>
    <td>".$row['messageTitle']. "</td>
    <td>".$row['date_sent']. "</td>
    <td>".$row['messageContent']. "</td>
</tr>";
}
}
}
}
}
?>

```

#### userprofile.php

```

...
<!-- Displays messages from other users and messages sent -->
<div id = "message-window">
    <!-- Messages from other users -->
    <div id = "messageinbox">
        <h4> Your private message inbox (messages other forum members
have sent to you): </h4>
        <br/>
        <table id = "usermessageinbox">
            <tr>
                <th> From </th>
                <th> Title </th>
                <th> Date sent </th>
                <th> Message </th>
            </tr>
            <?php
                $update -> DisplayMessage2($_SESSION['userid']);
            ?>
        </table>
    </div>
    <br/>
    <!-- Messages you sent -->
    <div id = "sentmessages">

```

```

</h4> Private messages you have sent to other forum members:
<br/>
<table id = "messagessend">
  <tr>
    <th> To </th>
    <th> Title </th>
    <th> Date sent </th>
    <th> Message </th>
  </tr>
  <?php
    $update -> DisplayMessages1($_SESSION['userid']);
  ?>
</table>
</div>
</div>

<!-- Option for sending private message -->
<div id = "privatemessage-window">
  <h3> Send Private Message </h3>

  <form method = "POST" action = "userprofile.php" id =
"sendpm">
    <label> Enter username of forum member: <input type =
"text" name = "memberName"/> </label>
    <br/>
    <label> Enter title of private message: <input type = "text"
name = "messageTitle"/> </label>
    <br/><br/><br/>
    <label> Enter Message: </label>
    <br/>
    <textarea name = "pm-message"></textarea>
    <br/><br/>
    <input type = "submit" name = "submitPM" value = "Send
PM"/>
  </form>
  <?php
    if(isset($_POST['submitPM']))
    {
      $datecreated = date("m/d/Y"); //when the message was
created
      $username = $_SESSION['username']; //username of the
member sending the message
      $membername = $_POST['memberName']; //username of the
member who the message is being sent to

```

```

        $messagetitle = $_POST['messageTitle']; //title of the private
message
        $messagecontent = $_POST['pm-message']; //content of the
private message

        $update -> PrivateMessage($_SESSION['userid'],
        $messagetitle, $messagecontent, $membername, $username, $datecreated);
    }
    ?>
</div>
...

```

**8/1/19**

9. What you now want to work on the search field in the forumhome.php page, which will allow users to search for a topic thread in a particular subject. How this will work is that the code will query through the thread table, see if that topic thread exists and display the link to that topic thread (making sure the file for that thread topic exists and had not been deleted) in the form of a table displaying the topic thread and the subject of that topic thread. If the thread does not exist, simply output something like: “Thread in (Subject) does not exist”. In the ThreadHandler.php, as well as the forumhome.php and main.css pages, input the following:

**ThreadHandler.php**

```

<?php
    include_once 'database.php';

    /* This class will handle the processes of both creating the topic thread of a
particular subject
and displaying it on the forumhome.php and the subject pages */
    class ThreadHandler
    {
        private $db;

        //SQL variables
        private $sql1; //for the createThread() method. For storing the thread
information in the thread table.
        private $sql2; //for the displayThread() method. For obtaining the thread
information from the thread table and displaying it in their respective topic
thread file
        private $sql3; //for the createReply() method. For obtaining the user
comment and storing it in the replies table
        private $sql4; //for the displayReply() method. For obtaining the user
comment from the replies table and displaying it in the respective topic thread
        private $sql5; //for the displayThread() method. For counting how many
replies were made to a particular topic tread
    }

```

private \$sql6; //for the obtainTopicThread() method. For obtaining the last topic thread made to a particular subject

private \$sql7; //for the showThreads() method. Shows the threads that a particular user created

**private \$sql8; //for the searchThreads() method. For matching a topic thread input to one in the thread table**

```
public function __construct()
{
    $this->db = new Connection();
    $this->db = $this->db->connect();
}
```

/\* This method is responsible for obtaining the thread information and storing that information in the threads table \*/

```
public function createThread($directory, $id, $author, $regdate,
$profilepic, $subject, $topic, $title, $content, $topicPage, $subjectStyling)
```

```
{
    /* Before adding the thread information into the database, first append
that information into a file called 'DiscussionWindow_(topic)_(title).php' */
    $threadfile = fopen("$directory/DiscussionWindow_$topic" . "_" . "
$title.php", "w");
```

```
    $threadPath = "$directory/DiscussionWindow_$topic" . "_" . "
$title.php"; //full thread file path
    $regularPath = "DiscussionWindow_$topic" . "_" . " $title.php"; //half
thread file path
    $threadCreated = date("m/d/Y");
```

```
    //set the thread page layout for the
'DiscussionWindow_(topic)_(title).php' page
    $threadlayout = "<?php
        session_start();
        include '../..../includes/ThreadHandler.php';
    ?>
    <html>
    <head>
        <title> $title </title>
        <meta charset = 'utf-8'/>

        <link rel = 'stylesheet' href =
'../..../ThreadsCssFiles/Navbar.css'/>
        <link rel = 'stylesheet' href =
'../..../ThreadsCssFiles/DiscussionWindow.css'/>
    </head>
```

```

<body id = '$subject' class = '$subjectStyling'>
  <!-- Navbar -->
  <ul class = 'navbar'>
    <li class = 'left-tabs'> <a href = '../././forumhome.php'>
Home </a> </li>
    <li class = 'left-tabs'> <a href = '#'> Board </a> </li>
    <li class = 'left-tabs'> <a href = '#'> Community </a> </li>

    <?php ".$$.upic = ".$$_SESSION['pic']; ?>

    <li class = 'right-tabs'><a href = '../././includes/logout.php'>
Logout </a></li>
    <li class = 'dropdown right-tabs'>
      <a href = '#' class = 'dropdownButton'> Hello: <?php echo
".$$_SESSION['username']; ?> </a>
      <div class = 'dropdown-content'>
        <a href = '#'> <?php echo "\"<img src =
'../././".$$.upic' height='70' width='70'/>\""; ?> </a>
        <a href = '#'> Full Name: <?php echo
".$$_SESSION['firstname'].".\" \". ".$$_SESSION['lastname']; ?> </a>
        <a href = '#'> Email: <?php echo
".$$_SESSION['useremail']; ?> </a>
        <a href = '#'> Current Status: <?php echo
".$$_SESSION['status']; ?> </a>
        <a href = '#'> Join Date: <?php echo
".$$_SESSION['regdate']; ?> </a>
        <a href = '../././userprofile.php'> Go to account
settings </a>
      </div>
    </li>
  </ul>

  <!-- Thread window -->
  <div class = 'threadWindow'>
    <div class = 'userInfo'>
      <div id = 'userPicture'> <?php echo "\"<img src =
'../././".$$.profilepic.'" height='70' width='70'/>\""; ?> </div>
      <h2> $author </h2>
      <h4> Join date: '$regdate' </h4>
    </div>
    <div class = 'threadContent'>
      <h1> $title </h1>
      <h4> Thread created: $threadCreated </h4>
      <h5> Topic: $topic </h5>
      <div class = 'underline' style = 'border-bottom: 1px solid
black;'></div>

```



```

        <p>
            $content
        </p>
    </div>
</div>

<br/><br/><br/>

<!-- Replies window -->
<?php
    ".$$. "replies = new ThreadHandler();

    ".$$. "Directory = \"$threadPath\";

    ".$$. "replies -> displayReply(".$$. "Directory);
?>

<br/><br/><br/>

<!-- Comment window -->
<div class = 'commentWindow'>
    <div class = 'styleButtons'>
        <button id = 'bold' onclick = 'styleText(\"<b>\",
\"</b>\")'> <abbr title = 'Bold'> <b>B</b> </abbr> </button> <br/> <br/>
        <button id = 'italic' onclick = 'styleText(\"<em>\",
\"</em>\")'> <abbr title = 'Italic'> <em>i</em> </abbr> </button> <br/> <br/>
        <button id = 'underline' onclick = 'styleText(\"<u>\",
\"</u>\")'> <abbr title = 'Underline'> <u>U</u> </abbr> </button> <br/> <br/>
        <button id = 'subscript' onclick = 'styleText(\"<sub>\",
\"</sub>\")'> <abbr title = 'Subscript'> A<sub>2</sub> </abbr> </button>
<br/> <br/>
        <button id = 'superscript' onclick = 'styleText(\"<sup>\",
\"</sup>\")'> <abbr title = 'Superscript'> A<sup>2</sup> </abbr> </button>
<br/> <br/>
        <button id = 'quote' onclick =
'styleText(\"<blockquote>\", \"</blockquote>\")'> <abbr title = 'Quote'> \"Q\"
</abbr> </button>
    </div>
    <div class = 'commentField'>
        <form method = 'POST' action = '$regularPath'>
            <textarea name = 'commentbox' id = 'commentbox'
rows = '15' cols = '90'></textarea> <br/>
            <input type = 'submit' name = 'submit-comment'
value = 'Submit comment' />
        </form>
    </div>
<?php

```

```

        if(isset("$.$_POST['submit-comment']"))
        {
            $(".$").Replies = new ThreadHandler();

            $(".$").uid = $(".$")._SESSION['userid']; //id of the
user who made the comment/reply
            $(".$").uname = $(".$")._SESSION['username'];
//username of the user who made the comment/reply
            $(".$").upic = $(".$")._SESSION['pic']; //profile pic of
user who made the comment/reply
            $(".$").Title = "$title"; //title of the thread who the
user is replying to
            $(".$").Directory = "$threadPath"; //full file path of
the thread file
            $(".$").replyCreated = date('m/d/Y'); //date the reply
was created
            $(".$").Reply = $(".$")._POST['commentbox']; //the
reply

            $(".$").Replies -> createReply($(".$").uid,
$(".$").uname, $(".$").upic, $(".$").Title, $(".$").Directory, $(".$").replyCreated,
$(".$").Reply);
        }
    ?>
</div>
</div>

<script src =
'../../ThreadsJSFiles/DiscussionWindow.js'></script>
</body>
</html>";

fwrite($threadfile, $threadlayout);

/* Now add the thread information to the database */
$this -> sql1 = "INSERT INTO threads(user_id, thread_author,
thread_subject, thread_topic, thread_title, thread_content, thread_file,
thread_creation)
VALUES('$id', '$author', '$subject', '$topic', '$title', '$content',
'$threadPath', '$threadCreated');"

$stmt1 = $this -> db -> stmt_init();

if(!$stmt1 -> prepare($this -> sql1))
{
    echo "<script> alert('SQL ERROR: ". $stmt1 -> error . "'); </script>";
}

```

```

    }
    else
    {
        $stmt1 -> execute();
        header("Location: $topicPage?threadcreation=success");
    }
}

/* This method is responsible for displaying the topic thread on the
forumhome.php page and the subject pages */
public function displayThread($subject, $topic)
{
    /* This query is for displaying each topic thread of a particular subject */
    $this -> sql2 = "SELECT thread_author, thread_subject, thread_topic,
thread_title, thread_file
                    FROM threads
                    WHERE thread_subject='$subject' AND
thread_topic='$topic'";

    $stmt2 = $this -> db -> stmt_init();

    if(!$stmt2 -> prepare($this -> sql2))
    {
        echo "<script> alert('SQL ERROR: " . $stmt2 -> error . "'); </script>";
    }
    else
    {
        $stmt2 -> execute();
        $result1 = $stmt2 -> get_result();

        if($result1 -> num_rows > 0)
        {
            while($row1 = $result1 -> fetch_assoc()) //stores the results of the
thread table query in an associative array and stores it in a variable called
'$row1'
            {
                $dir = $row1['thread_file']; //obtains the path of the topic thread at
hand

                /* This query is for obtaining the number of replies made to a
particular topic thread */
                $this -> sql5 = "SELECT COUNT(replies_id) AS 'count' FROM
replies WHERE thread_file='$dir'"; //counts how many replies were made to a
topic thread

```

```

$stmt5 = $this -> db -> stmt_init();

if(!$stmt5 -> prepare($this -> sql5))
{
    echo "<script> alert('SQL ERROR: " . $stmt5 -> error . "');";
</script>";
}
else
{
    $stmt5 -> execute();
    $result2 = $stmt5 -> get_result();

    while($row2 = $result2 -> fetch_assoc()) //stores the results of the
    replies table query in an associative array and stores it in a variable called
    '$row2'
    {
        echo "<tr>
            <td><a href = \"" . $row1['thread_file'] . "\">\" .
$row1['thread_title'] . "</a></td>
            <td>\" . $row2['count'] . "</td>
            <td> By: <em>\" . $row1['thread_author'] . "</em> </td>
            </tr>";
    }
}
}
}
else
{
    echo "<b id = 'warningMessage'> There are no current threads posted.
</b>";
}
}
}

```

```

/* This method creates the comment a user made on a particular topic
thread */
public function createReply($id, $username, $pic, $title, $directory,
$dateCreated, $comment)
{
    /* Store the comment in the database with the respective thread file */
    if(empty($comment))
    {
        echo "<p id = 'errormessage'> You must enter a comment! </p>";
    }
    else
    {

```

```

        $this -> sql3 = "INSERT INTO replies(user_id, user_username,
        profile_pic, thread_title, thread_file, replies_creation, replies_content)
        VALUES('$id', '$username', '$pic', '$title', '$directory',
        '$dateCreated', '$comment')";

        $stmt3 = $this -> db -> stmt_init();

        if(!$stmt3 -> prepare($this -> sql3))
        {
            echo "<script> alert('SQL ERROR: ". $stmt3 -> error . "'); </script>";
        }
        else
        {
            $stmt3 -> execute();
            echo "<script> location.assign(location.href); </script>"; //refreshes
            current thread page to show replies as they are created
        }
    }
}

/* This method displays the comments that users made on a particular topic
thread */
public function displayReply($directory)
{
    /* Get only the replies of that particular topic thread */
    $this -> sql4 = "SELECT user_username, user_status, profile_pic,
    thread_file, replies_creation, replies_content
    FROM replies
    WHERE thread_file = '$directory'";

    $stmt4 = $this -> db -> stmt_init();

    if(!$stmt4 -> prepare($this -> sql4))
    {
        echo "<script> alert('SQL ERROR: ". $stmt4 -> error . "'); </script>";
    }
    else
    {
        $stmt4 -> execute();

        $result = $stmt4 -> get_result();

        if($result -> num_rows > 0)
        {
            while($row = $result -> fetch_assoc())
            {

```

```

        echo "<div class = 'repliesWindow'>
            <div class = 'userInfo'>
                <img src = \" . \"../..\" . $row['profile_pic'] . \"\" height='70'
width='70'> <br/>
                <h2>\" . $row['user_username'] . \"</h2>
                <p> Current Status: \" . $row['user_status'] . \" </p>
            </div>
            <div class = 'replyContent'>
                <p>\" . $row['replies_content'] . \"</p>
            </div>
        </div> <br/><br/>";
    }
}
else
{
    echo "<b id = 'warningMessage'> There are no current replies posted.
</b>";
}
}
}

```

/\* This method obtains the last topic thread that was made on a particular subject \*/

```

public function obtainTopicThread($subject, $topic)
{
    /* This SQL query selects the most recent topic thread of a particular
subject.
The most recent one has a high thread id number, thus the thread title
for that
thread must also be high as well */
    $this -> sql6 = "SELECT MAX(thread_id), thread_author,
thread_subject, thread_topic, MAX(thread_title), thread_file,
MAX(thread_creation)
FROM threads WHERE thread_subject='$subject' AND
thread_topic='$topic'";

```

```

    $stmt6 = $this -> db -> stmt_init();

```

```

    if(!$stmt6 -> prepare($this -> sql6))
    {
        echo "<script> alert('SQL ERROR: \" . $stmt6 -> error . \"'); </script>";
    }
    else
    {
        $stmt6 -> execute();
    }
}

```

```

$result = $stmt6 -> get_result();

while($row = $result -> fetch_assoc())
{
    if($row['MAX(thread_id)'] == null && $row['thread_author'] == null
    && $row['thread_subject'] == null
    && $row['thread_topic'] == null && $row['MAX(thread_title)'] ==
    null && $row['thread_file'] == null
    && $row['MAX(thread_creation)'] == null)
    {
        echo "<p> There are no threads posted </p>";
    }
    else
    {
        echo "<p>
        <a href = 'Subjects/$subject/'. $row['thread_file'] ." "> ".
        $row['MAX(thread_title)'] ." </a> <br>
        by <em> ". $row['thread_author'] ." </em> <br>
        Created on: ". $row['MAX(thread_creation)'] ."
        </p>";
    }
}
}

/* This method shows the threads that a user created */
public function showThreads($id)
{
    $this -> sql7 = "SELECT thread_title, thread_subject, thread_topic,
    thread_creation, thread_file FROM threads WHERE user_id='$id'";

    $stmt7 = $this -> db -> stmt_init();

    if(!$stmt7 -> prepare($this -> sql7))
    {
        echo "<script> alert('SQL ERROR: ". $stmt7 -> error . "'); </script>";
    }
    else
    {
        $stmt7 -> execute();

        $result = $stmt7 -> get_result();

        if($result -> num_rows == 0)
        {
            echo "<script>

```

```

        var threadwindow =
document.getElementById('threadsCreated');
        threadwindow.style.display = 'none';

        var Notice = document.createElement('p');
        var textnode = document.createTextNode('No threads for this
topic have been posted');

        var notice = Notice.appendChild(textnode);

        document.getElementById('user-posts').appendChild(notice);
    </script>";
    }
    else
    {
        while($row = $result -> fetch_assoc())
        {
            echo "<tr>
                <td><a href =
'Subjects/' . $row['thread_subject'] . '/' . $row['thread_file'] . ">" . $row['thread_title'
]. "</a></td>
                <td>" . $row['thread_subject'] . "</td>
                <td>" . $row['thread_topic'] . "</td>
                <td>" . $row['thread_creation'] . "</td>
            </tr>";
        }
    }
}

/* This method is for searching for a topic thread on a particular
subject in the threads table */
public function searchThreads($threadname, $subjectname)
{
    if(empty($threadname) && empty($subjectname))
    {
        echo "<p style = 'color: red; font-family: serif; text-align: center;'>
Do not just leave anything blank and submit it! </p>";
    }
    else
    {
        $this -> sql8 = "SELECT * FROM threads WHERE
thread_title='$threadname' AND thread_subject='$subjectname'";

        $stmt = $this -> db -> stmt_init();
    }
}

```



```

if(!$stmt -> prepare($this -> sql8))
{
    echo "<script> alert('".$stmt -> error."'); </script>";
}
else
{
    $stmt -> execute();
    $result = $stmt -> get_result();

    if($result -> num_rows == 0)
    {
        echo "<p style = 'color: red; font-family: serif; text-align:
center;'>
            No such thread exists! Make sure to enter the thread name
correctly as well as entering the subject as shown below!
        </p>";
    }
    else
    {
        while($row = $result -> fetch_assoc())
        {
            echo "<p style = 'color: white; font-family: serif; text-align:
center;'> Your search results: </p>
                <table id = 'searchResults'>
                    <thead>
                        <tr>
                            <th> Thread name </th>
                            <th> Subject </th>
                            <th> Author </th>
                            <th> Thread created </th>
                        </tr>
                    </thead>
                    <tbody>
                        <tr>
                            <td><a href =
'Subjects/' . $row['thread_subject'] . '/' . $row['thread_file'] . "'>". $row['thre
ad_title'] . "</a></td>
                            <td>". $row['thread_subject'] . "</td>
                            <td>". $row['thread_author'] . "</td>
                            <td>". $row['thread_creation'] . "</td>
                        </tr>
                    </tbody>
                </table>";
        }
    }
}

```

```
}  
}  
}  
?>
```

### **forumhome.php**

```
...  
    <!-- Search for forums -->  
    <div id = "searchwindow">  
        <p> Search for your favorite topic thread </p>  
  
        <form method = "POST" action = "forumhome.php">  
            <input type = "text" name = "threadname" placeholder = "Enter  
topic thread"/>  
            <br/>  
            <input type = "text" name = "subjectname" placeholder = "Enter  
subject"/>  
            <br/>  
            <input type = "submit" name = "submit-search" value = "Search"/>  
        </form>  
        <?php  
            if(isset($_POST['submit-search']))  
            {  
                $ThreadName = $_POST['threadname'];  
                $SubjectName = $_POST['subjectname'];  
  
                $searchThread = new ThreadHandler();  
                $searchThread -> searchThreads($ThreadName, $SubjectName);  
            }  
        ?>  
    </div>  
...
```

### **main.css**

```
...  
  
#searchwindow #searchResults, #searchwindow #searchResults th,  
#searchwindow #searchResults td, #searchwindow #searchResults td a  
{  
    font-family: Times New Roman;  
    border-radius: 5px;  
    padding: 7px;  
  
    color: #D3D3D3;  
    text-decoration: none;  
}  
  
#searchwindow #searchResults thead  
{
```

```

        background-color: purple;
    }

    #searchwindow #searchResults tbody
    {
        background-color: #2F4F4F;
    }
    ...

```

**8/7/19**

10. For some reason, when you log in, if you enter a correct username, but enter a wrong password, or vice versa, the user will still get logged in. You might want to fix the SQL code.

It turns out in the following SQL code “SELECT \* FROM accounts WHERE user\_username=? OR user\_password=?;”, when user does enter the password correctly, there is an error because the password in the database is hashed, thus it will only take the hashed password, not the password the user putted in. To fix this, you want to match the password the user putted in with the hashed password in the database to see if the hashed password is actually the password the user inputted in. In the login method of the Accounts class, input the following:

**AccountHandler.php**

```

/* This method is for logging in new users into the forum */
public function login($username, $password)
{
    if(empty($username) && empty($password))
    {
        echo "<p id = 'errorMessage'> Do not submit blank information! </p>";
    }
    else if(empty($username))
    {
        echo "<p id = 'errorMessage'> You must enter your username! </p>";
    }
    else if(empty($password))
    {
        echo "<p id = 'errorMessage'> You must enter your password! </p>";
    }
    else
    {
        $this->sql1 = "SELECT * FROM accounts WHERE user_username=?
OR user_password=?;";

        $stmt = $this->dbConn->stmt_init();

```

```

if(!$stmt -> prepare($this -> sql1))
{
    echo "<script> alert('SQL ERROR: " . $stmt -> error . "'); </script>";
}
else
{
    $stmt -> bind_param("ss", $username, $password);
    $stmt -> execute();

    $result = $stmt -> get_result();

    if($row = $result -> fetch_assoc()) //store the results in an associative
array
    {
        session_start();

        /*obtain the id, username, email, firstname, lastname as well as
profile picture
and status and registration date, and stores them in session
variables*/
        $_SESSION['userid'] = $row['user_id'];
        $_SESSION['firstname'] = $row['user_firstname'];
        $_SESSION['lastname'] = $row['user_lastname'];
        $_SESSION['useremail'] = $row['user_email'];
        $_SESSION['password'] = $row['user_password'];
        $_SESSION['username'] = $row['user_username'];
        $_SESSION['pic'] = $row['profile_pic'];
        $_SESSION['status'] = $row['user_status'];
        $_SESSION['regdate'] = $row['reg_date'];

        //checks to see if the password the user typed in matches the
hashed password in the accounts table
        if(password_verify($password, $_SESSION['password']))
        {
            //sends user to the forum homepage upon successful login
            header("Location: forumhome.php?login=successful");
            exit();
        }
        else
        {
            echo "<p id = 'errorMessage'> Username exists but password
is incorrect! </p>";
        }
    }
    else
    {

```

```

        echo "<p id = 'errormessage'> There is no such account! </p>";
    }
}
}
}

```

This will check the password the user inputted and match it with the hashed password stored in the session variable. If they both match, the user will be directed to the forum home page.

11. What you want to do now is, in the tab, there are three links: “Home”, “Members” and “Activity”.

- 1) Home just links back to the forum homepage (forumhome.php).
- 2) Members checks both new users who just registered an account for your forum website and who is currently logged into the forum website.
- 3) Activity checks recent user activity, including new topic threads created by users, replies to a particular topic thread made by other users, who just created a forum account.

You don’t have to do anything for the Home link since it does one small thing.

Starting with the Members link, you want to first create a php file called “members.php”, and this page will show all the members who are logged into your website. You can start off with a basic layout because you want to start implementing the code that will check new users who have just signed up on your site, as well as checking the users who are currently logged on to the forum.

#### **members.php**

```

<?php
    session_start();
?>
<html>
    <head>
        <title> Check to see who's logged in and who just signed up to this
        website </title>

        <link href="https://fonts.googleapis.com/css?family=Lobster"
        rel="stylesheet">
        <link rel = "stylesheet" href = "css/main.css"/>
        <link rel = "stylesheet" href = "css/Navbar.css"/>
    </head>

    <body>
        <!-- navbar -->
        <ul class = "navbar">
            <li class = "left-tabs"><a href = "forumhome.php"> Home </a></li>

```

```

<li class = "left-tabs"><a href = "members.php"> Members </a></li>
<li class = "left-tabs"><a href = "#"> Activity </a></li>

<li class = "right-tabs"><a href = "includes/logout.php"> Logout
</a></li>
<li class = "dropdown right-tabs">
  <a href = "#" class = "dropdownButton"> Hello: <?php echo
$_SESSION['username']; ?> </a>
  <div class = "dropdown-content">
    <a href = "#"> <?php echo "<img src = '" . $_SESSION['pic'] . '"
height='70' width='70'/>"; ?> </a>
    <a href = "#"> Full Name: <?php echo $_SESSION['firstname'] .
" " . $_SESSION['lastname']; ?> </a>
    <a href = "#"> Email: <?php echo $_SESSION['useremail']; ?>
</a>
    <a href = "#"> Current Status: <?php echo $_SESSION['status'];
?> </a>
    <a href = "#"> Join Date: <?php echo $_SESSION['regdate']; ?>
</a>
    <a href = "userprofile.php"> Go to account settings </a>
  </div>
</li>
</ul>

<!-- Members window (displays both the number of users who just
signed up an account and the number of users who are currently logged in) -->
<div id = "memberswindow">
  <!-- This table displays the number of user who just registered for a
forum account -->
  <h4 style = "color: white; font-family: 'Lobster', cursive; font-size:
25px;"> See who just registered </h4>
  <table id = "registeredUsers">
    <thead>
      <th> Username </th>
      <th> Date of registration </th>
    </thead>
    <tbody>
      <td> username </td>
      <td> registration </td>
    </tbody>
  </table>

  </br></br></br></br><br/>

  <!-- This table displays the number of users currently logged on to the
forum -->

```

```
        <h4 style = "color: white; font-family: 'Lobster', cursive; font-size:
25px;"> See who's currently logged in </h4>
        <table id = "loggedUsers">
            <thead>
                <th> Username </th>
                <th> Logged in at </th>
            </thead>
            <tbody>
                <td> username </td>
                <td> logged in </td>
            </tbody>
        </table>
    </div>
</body>
```

```
</html>
```

#### main.css

```
body
{
    background-image:
url("https://png.pngtree.com/thumb_back/fh260/back_pic/04/00/23/4757fc9e77
387db.jpg");
    background-repeat: no-repeat;
    background-size: cover;
}

/* Styles the forumhome.php page */
#searchwindow
{
    padding-left: 25%;
    padding-top: 10%;
}
#searchwindow input[type=text]
{
    width: 60%;
    height: 30px;

    background-color: #F5F5DC;
    font-family: Times New Roman;
}
#searchwindow input[type=submit]
{
    width: 60%;
    height: 30px;

    background-color: #F5FFFA;
    font-family: Times New Roman;
```

```

}
#searchwindow p
{
    padding-right: 30%;
    font-family: "Lobster", cursive;
    font-size: 200%;
}
#searchwindow #searchResults, #searchwindow #searchResults th,
#searchwindow #searchResults td, #searchwindow #searchResults td a
{
    font-family: Times New Roman;
    border-radius: 5px;
    padding: 7px;

    color: #D3D3D3;
    text-decoration: none;
}
#searchwindow #searchResults thead
{
    background-color: purple;
}
#searchwindow #searchResults tbody
{
    background-color: #2F4F4F;
}
#forumwindow
{
    padding-top: 70px;
    padding-left: 5%;
}
#searchwindow p, #forumwindow .table th, #forumwindow .table td,
#forumwindow .table td a
{
    color: #D3D3D3;
    text-decoration: none;
}
#forumwindow .table th, #forumwindow .table td
{
    font-family: Times New Roman;
    border-radius: 5px;
    padding: 7px;
}
#forumwindow .table .leftside
{
    width: 8%;
}

```



```
#forumwindow .table .rightside
{
    width: 20%;
}
#forumwindow .table thead
{
    background-color: purple;
}
#forumwindow .table tbody
{
    background-color: #2F4F4F;
}

/* Styles the members.php page */
#memberswindow
{
    padding-left: 40%;
    padding-top: 10%;
}
#memberwindow .welcomeTitles
{
    color: white;
    font-family: "Lobster", cursive;
}
#memberswindow #registeredUsers, #memberswindow #loggedUsers
{
    color: white;

    text-align: center;
}
#memberswindow #registeredUsers thead th, #memberswindow
#registeredUsers tbody td, #memberswindow #loggedUsers thead th,
#memberswindow #loggedUsers tbody td
{
    border: 1px solid white;

    padding: 10px;
}
```

### 8/7/19 - 8/9/19

Now, in the AccountHandler.php file, inside the Accounts class, create two methods:

- 1) RegisteredUsers: This method shows new users who just registered a forum account. It will display up to 10 new user accounts and after a week, the list will refresh and a new list of 10 new users will be shown.
- 2) setLoggedUsers: This method sets the time the user logged in. If the user is not logged in, just output "Currently offline".
- 3) displayLoggedUsers: This method displays all the users and shows if they are logged in or not.

For the setLoggedUsers and displayLoggedUsers methods, you want to create a table called "activeaccounts", and this table will check who's logged in and who's not.

Input the following SQL code:

```
CREATE TABLE activeaccounts
(
    user_id int NOT NULL AUTO_INCREMENT,
    user_username varchar(100) NOT NULL,
    log_status varchar(100), /* Stores whether the user is logged in or offline */
    login_time varchar(100), /* Stores when the user last logged in */
    PRIMARY KEY(user_id)
);
```

Now, in the AccountHandler.php and members.php pages, add the following:

#### AccountHandler.php

```
<?php
    include_once 'database.php';

    /* This class will handle the login, logout and signup processes */
    class Accounts
    {
        private $dbConn;
        private $sql1; //For the login(). For logging in new users to the
        forumhome.php

        //Both are for the signup() method. For creating new user accounts
        private $sql2; //For the signup() method. For adding new users to the
        accounts table
        private $sql3; //For the signup() method. For adding new users to the
        activeaccounts table
        private $sql4; //For the RegisteredUsers() method. For showing new users
        who have just registered a new forum account

        //Both are for the setLoggedUsers() method. Sets the login time, that is,
        when the time the user first logged in
```

```
private $sql5; //for showing that the user is logged on  
private $sql6; //for showing that users are logged off
```

```
private $sql7; //For the displayLoggedUsers() method. For displaying  
how many users are logged in and how many aren't
```

```
/* initialize connection to the discussionforum database upon calling the  
object of the class */
```

```
public function __construct()  
{  
    $this -> dbConn = new Connection();  
    $this -> dbConn = $this -> dbConn -> connect();  
}
```

```
/* This method is for logging in new users into the forum */
```

```
public function login($username, $password)  
{  
    if(empty($username) && empty($password))  
    {  
        echo "<p id = 'errorMessage'> Do not submit blank information! </p>";  
    }  
    else if(empty($username))  
    {  
        echo "<p id = 'errorMessage'> You must enter your username! </p>";  
    }  
    else if(empty($password))  
    {  
        echo "<p id = 'errorMessage'> You must enter your password! </p>";  
    }  
    else  
    {  
        $this -> sql1 = "SELECT * FROM accounts WHERE user_username=?  
OR user_password=?";
```

```
$stmt = $this -> dbConn -> stmt_init();
```

```
if(!$stmt -> prepare($this -> sql1))  
{  
    echo "<script> alert('SQL ERROR: " . $stmt -> error . "'); </script>";  
}  
else  
{  
    $stmt -> bind_param("ss", $username, $password);  
    $stmt -> execute();
```

```
$result = $stmt -> get_result();
```

```

        if($row = $result -> fetch_assoc()) //store the results in an associative
array
        {
            session_start();

            /*obtain the id, username, email, firstname, lastname as well as
profile picture
            and status and registration date, and stores them in session
variables*/
            $_SESSION['userid'] = $row['user_id'];
            $_SESSION['firstname'] = $row['user_firstname'];
            $_SESSION['lastname'] = $row['user_lastname'];
            $_SESSION['useremail'] = $row['user_email'];
            $_SESSION['password'] = $row['user_password'];
            $_SESSION['username'] = $row['user_username'];
            $_SESSION['pic'] = $row['profile_pic'];
            $_SESSION['status'] = $row['user_status'];
            $_SESSION['regdate'] = $row['reg_date'];

            //checks to see if the password the user typed in matches the hashed
password in the accounts table
            if(password_verify($password, $_SESSION['password']))
            {
                //sends user to the forum homepage upon successful login
                header("Location: forumhome.php?login=successful");
                exit();
            }
            else
            {
                echo "<p id = 'errormessage'> Username exists but password is
incorrect! </p>";
            }
        }
        else
        {
            echo "<p id = 'errormessage'> There is no such account! </p>";
        }
    }
}

/* This method is for registering new users and adding their information to
the accounts table */
public function signup($accountCreated, $firstname, $lastname, $email,
$username, $password)

```

```

        {
            if(empty($firstname) && empty($lastname) && empty($email) &&
empty($username) && empty($password))
            {
                echo "<p id = 'errormessage'> Do not submit empty information!
</p>";
            }
            else if(empty($firstname) || empty($lastname) || empty($email) ||
empty($username) || empty($password))
            {
                echo "<p id = 'errormessage'> You must fill out everything! </p>";
            }
            else
            {
                $this -> sql2 = "INSERT INTO accounts(reg_date, user_firstname,
user_lastname, user_email, user_username, user_password)
                VALUES(?, ?, ?, ?, ?, ?);";

                $stmt1 = $this -> dbConn -> stmt_init();

                if(!$stmt1 -> prepare($this -> sql2))
                {
                    echo "<script> alert('SQL ERROR: " . $stmt1 -> error . "');
</script>";
                }
                else
                {
                    $hashedPwd = password_hash($password,
PASSWORD_DEFAULT);

                    $stmt1 -> bind_param("sssss", $accountCreated, $firstname,
$lastname, $email, $username, $hashedPwd);
                    $stmt1 -> execute();

                    //now insert the username in the activeaccounts table
                    $this -> sql3 = "INSERT INTO activeaccounts(user_username)
                    VALUES('$username');";

                    $stmt2 = $this -> dbConn -> stmt_init();

                    if(!$stmt2 -> prepare($this -> sql3))
                    {
                        echo "<script> alert('SQL ERROR: " . $stmt2 -> error . "');
</script>";
                    }
                    else

```

```

        {
            $stmt2 -> execute();

            echo "<p id = 'successmessage'> You have successfully signed up!
Sign in now to enter forum discussion! </p>";
        }
    }
}

/* This method checks the new users who just signed up a new forum
account */
public function RegisteredUsers()
{
    $this -> sql4 = "SELECT reg_date, user_username FROM accounts;";

    $stmt = $this -> dbConn -> stmt_init();

    if(!$stmt -> prepare($this -> sql4))
    {
        echo "<script> alert('SQL ERROR: " . $stmt -> error . "'); </script>";
    }
    else
    {
        $stmt -> execute();
        $result = $stmt -> get_result();

        if($result -> num_rows > 0)
        {
            while($row = $result -> fetch_assoc())
            {
                echo "<tbody>
                    <td>".$row['user_username']."</td>
                    <td>".$row['reg_date']."</td>
                </tbody>";
            }
        }
        else
        {
            echo "<p style = 'color: red;'> There are no new users at this time
</p>";
        }
    }
}

```

```

        /* Checks users who are currently logged into the forum. If logged in,
        set the time they logged in.
        Else, output 'Offline' */
        public function setLoggedUsers($id, $username)
        {
            $loggeddate = date("m/d/Y h:i A"); //when the user first logged in

            //if the id and username matches the session variables that are set
            if($id == $_SESSION['userid'] && $username ==
            $_SESSION['username'])
            {
                //indicates that the user is logged on
                $this -> sql5 = "UPDATE activeaccounts SET user_id='$id',
log_status='Active', login_time='$loggeddate' WHERE
user_username='$username'";
                $stmt1 = $this -> dbConn -> stmt_init();
                $stmt1 -> prepare($this -> sql5);
                $stmt1 -> execute();

                //indicates that the user is logged off
                $this -> sql6 = "UPDATE activeaccounts SET log_status='Offline',
login_time='Not Logged On' WHERE user_username!='$username'";
                $stmt2 = $this -> dbConn -> stmt_init();
                $stmt2 -> prepare($this -> sql6);
                $stmt2 -> execute();
            }
        }

        /* Display all the number of users who are logged in and those who are
        not */
        public function displayLoggedUsers()
        {
            $this -> sql7 = "SELECT * FROM activeaccounts;";

            $stmt = $this -> dbConn -> stmt_init();
            $stmt -> prepare($this -> sql7);
            $stmt -> execute();

            $result = $stmt -> get_result();
            if($result -> num_rows > 0)
            {
                while($row = $result -> fetch_assoc())
                {
                    echo "<tbody>
                        <td>".$row['user_username']. "</td>
                        <td>".$row['log_status']. "</td>

```

```

        <td>".$row['login_time']."</td>
    </tbody>";
    }
}
else
{
    echo "<p id = 'errormessage'> There are no users currently logged
on </p>";
}
}
}

?>

```

### members.php

```

<?php
    session_start();

    include "includes/AccountHandler.php";
    $registeredAccounts = new Accounts();
?>
<html>
    <head>
        <title> Check to see who's logged in and who just signed up to this
website </title>

        <link href="https://fonts.googleapis.com/css?family=Lobster"
rel="stylesheet">
        <link rel = "stylesheet" href = "css/main.css"/>
        <link rel = "stylesheet" href = "css/Navbar.css"/>
    </head>

    <body>
        <!-- navbar -->
        <ul class = "navbar">
            <li class = "left-tabs"><a href = "forumhome.php"> Home </a></li>
            <li class = "left-tabs"><a href = "members.php"> Members </a></li>
            <li class = "left-tabs"><a href = "#"> Activity </a></li>

            <li class = "right-tabs"><a href = "includes/logout.php"> Logout
</a></li>
            <li class = "dropdown right-tabs">
                <a href = "#" class = "dropdownButton"> Hello: <?php echo
$_SESSION['username']; ?> </a>
                <div class = "dropdown-content">
                    <a href = "#"> <?php echo "<img src = '' . $_SESSION['pic'] . ''
height='70' width='70'/>"; ?> </a>

```



```

        <a href = "#"> Full Name: <?php echo $_SESSION['firstname'] .
" " . $_SESSION['lastname']; ?> </a>
        <a href = "#"> Email: <?php echo $_SESSION['useremail']; ?>
</a>
        <a href = "#"> Current Status: <?php echo $_SESSION['status'];
?> </a>
        <a href = "#"> Join Date: <?php echo $_SESSION['regdate']; ?>
</a>
        <a href = "userprofile.php"> Go to account settings </a>
    </div>
</li>
</ul>

<!-- Members window (displays both the number of users who just
signed up an account and the number of users who are currently logged in) -->
<div id = "memberswindow">
    <!-- This table displays the number of user who just registered for a
forum account -->
    <h4 style = "color: white; font-family: 'Lobster', cursive; font-size:
25px;"> See who just registered </h4>
    <table id = "registeredUsers">
        <thead>
            <th> Username </th>
            <th> Date of registration </th>
        </thead>
        <?php
            $registeredAccounts -> RegisteredUsers();
        ?>
    </table>

    <br><br><br><br><br>

    <!-- This table displays the number of users currently logged on to the
forum -->
    <h4 style = "color: white; font-family: 'Lobster', cursive; font-size:
25px;"> See who's currently logged in </h4>
    <table id = "loggedUsers">
        <thead>
            <th> Username </th>
            <th> Status </th>
            <th> Logged in at </th>
        </thead>
        <?php
            //if user is logged in, update the user_logged column with
the time the user logged in
            $userid = $_SESSION['userid'];

```

```

                $uname = $_SESSION['username'];
                $registeredAccounts -> setLoggedUsers($userid, $uname);
            ?>
        <?php
            $registeredAccounts -> displayLoggedUsers();
        ?>
    </table>
</div>
</body>
</html>

```

There is one bug you'll have to search up what to do. In browser one launching the website and going to the Members page, it will show one user who is logged in as 'Active', but in browser two as another user, the user who logged in browser one will be displayed as 'Offline' in browser two, despite still being logged on in browser one. You'll have to figure that out later on after completing the rest of the website.

12. Next, you want to work on the Activity tab, which shows recent activity in the database, including new topic threads created by users, replies to a particular topic thread made by other users, who just created an forum account. This will involve multiple SQL queries, but you're only going to display the last row of each table needed.

First you'll want to create a file called "activities.php", and this display all the recent forum member activities. First design the page:

```

activities.php
<?php
    session_start();

    include 'includes/AccountHandler.php';
    $recentactivity = new Accounts();
?>
<html>
<head>
    <title> See recent activity of forum members! </title>

    <meta charset = "utf-8"/>

    <link rel = "stylesheet" href = "css/main.css"/>
    <link rel = "stylesheet" href = "css/Navbar.css"/>
</head>

<body>
    <!-- navbar -->
    <ul class = "navbar">
        <li class = "left-tabs"><a href = "forumhome.php"> Home </a></li>
        <li class = "left-tabs"><a href = "members.php"> Members </a></li>
    </ul>

```

```

        <li class = "left-tabs"><a href = "activities.php"> Activity </a></li>

        <li class = "right-tabs"><a href = "includes/logout.php"> Logout
</a></li>
        <li class = "dropdown right-tabs">
            <a href = "#" class = "dropdownButton"> Hello: <?php echo
$_SESSION['username']; ?> </a>
            <div class = "dropdown-content">
                <a href = "#"> <?php echo "<img src = '' . $_SESSION['pic'] . ''
height='70' width='70' />"; ?> </a>
                <a href = "#"> Full Name: <?php echo $_SESSION['firstname'] .
" " . $_SESSION['lastname']; ?> </a>
                <a href = "#"> Email: <?php echo $_SESSION['useremail']; ?>
</a>
                <a href = "#"> Current Status: <?php echo $_SESSION['status'];
?> </a>
                <a href = "#"> Join Date: <?php echo $_SESSION['regdate']; ?>
</a>
                <a href = "userprofile.php"> Go to account settings </a>
            </div>
        </li>
    </ul>

    <!-- Forum activity table -->
    <div id = "forumactivity">
        <h3>Recent Activity</h3> <p>Activity stream for all registered
members</p> </th>
        <div style = "border-bottom: 1px solid white;"></div>
        <?php
            $recentactivity -> RecentActivity();
        ?>
    </div>
</body>
</html>

```

#### main.css

```

...
    /* Styles the activites.php page */
    #forumactivity
    {
        color: white;

        text-align: center;
    }
...

```

Now, in the AccountHandler.php, you want to create an Accounts method called “RecentActivity()”, and this will show recent forum member activity. In the AccountHandler.php and activities.php, add the following code:

**AccountHandler.php**

```
<?php
    include_once 'database.php';

    /* This class will handle the login, logout and signup processes */
    class Accounts
    {
        private $dbConn;
        private $sql1; //For the login(). For logging in new users to the
forumhome.php

        //Both are for the signup() method. For creating new user accounts
        private $sql2; //For the signup() method. For adding new users to the
accounts table
        private $sql3; //For the signup() method. For adding new users to the
activeaccounts table

        private $sql4; //For the RegisteredUsers() method. For showing new users
who have just registered a new forum account

        //Both are for the setLoggedUsers() method. Sets the login time, that is,
when the time the user first logged in
        private $sql5; //For showing that the user is logged on
        private $sql6; //For showing that users are logged off

        private $sql7; //For the displayLoggedUsers() method. For displaying how
many users are logged in and how many aren't

        //For the RecentActivity() method. For displaying recent user forum
activity
        private $sql8; //For looping through the accounts table. To see who
just created an account
        private $sql9; //For looping through the messageinbox table. To see
who just sent a message
        private $sql10; //For looping through the replies table. To see who just
replied to a thread
        private $sql11; //For looping through the threads table. To see who
just created a thread

        /* initialize connection to the discussionforum database upon calling the
object of the class */
        public function __construct()
        {
```

```

$this->dbConn = new Connection();
$this->dbConn = $this->dbConn->connect();
}

/* This method is for logging in new users into the forum */
public function login($username, $password)
{
    if(empty($username) && empty($password))
    {
        echo "<p id = 'errormessage'> Do not submit blank information! </p>";
    }
    else if(empty($username))
    {
        echo "<p id = 'errormessage'> You must enter your username! </p>";
    }
    else if(empty($password))
    {
        echo "<p id = 'errormessage'> You must enter your password! </p>";
    }
    else
    {
        $this->sql1 = "SELECT * FROM accounts WHERE user_username=?
OR user_password=?;";

        $stmt = $this->dbConn->stmt_init();

        if(!$stmt->prepare($this->sql1))
        {
            echo "<script> alert('SQL ERROR: " . $stmt->error . "'); </script>";
        }
        else
        {
            $stmt->bind_param("ss", $username, $password);
            $stmt->execute();

            $result = $stmt->get_result();

            if($row = $result->fetch_assoc()) //store the results in an associative
array
            {
                session_start();

                /*obtain the id, username, email, firstname, lastname as well as
profile picture
and status and registration date, and stores them in session
variables*/

```

```

$_SESSION['userid'] = $row['user_id'];
$_SESSION['firstname'] = $row['user_firstname'];
$_SESSION['lastname'] = $row['user_lastname'];
$_SESSION['useremail'] = $row['user_email'];
$_SESSION['password'] = $row['user_password'];
$_SESSION['username'] = $row['user_username'];
$_SESSION['pic'] = $row['profile_pic'];
$_SESSION['status'] = $row['user_status'];
$_SESSION['regdate'] = $row['reg_date'];

//checks to see if the password the user typed in matches the hashed
password in the accounts table
if(password_verify($password, $_SESSION['password']))
{
    //sends user to the forum homepage upon successful login
    header("Location: forumhome.php?login=successful");
    exit();
}
else
{
    echo "<p id = 'errormessage'> Username exists but password is
incorrect! </p>";
}
}
else
{
    echo "<p id = 'errormessage'> There is no such account! </p>";
}
}
}

/* This method is for registering new users and adding their information to
the accounts table */
public function signup($accountCreated, $firstname, $lastname, $email,
$username, $password)
{
    if(empty($firstname) && empty($lastname) && empty($email) &&
empty($username) && empty($password))
    {
        echo "<p id = 'errormessage'> Do not submit empty information!
</p>";
    }
    else if(empty($firstname) || empty($lastname) || empty($email) ||
empty($username) || empty($password))
    {

```

```

        echo "<p id = 'errormessage'> You must fill out everything! </p>";
    }
    else
    {
        $this -> sql2 = "INSERT INTO accounts(reg_date, user_firstname,
user_lastname, user_email, user_username, user_password)
        VALUES(?, ?, ?, ?, ?, ?);";

        $stmt1 = $this -> dbConn -> stmt_init();

        if(!$stmt1 -> prepare($this -> sql2))
        {
            echo "<script> alert('SQL ERROR: " . $stmt1 -> error . "');
</script>";
        }
        else
        {
            $hashedPwd = password_hash($password,
PASSWORD_DEFAULT);

            $stmt1 -> bind_param("sssss", $accountCreated, $firstname,
$lastname, $email, $username, $hashedPwd);
            $stmt1 -> execute();

            //now insert the username in the activeaccounts table
            $this -> sql3 = "INSERT INTO activeaccounts(user_username)
            VALUES('$username');";

            $stmt2 = $this -> dbConn -> stmt_init();

            if(!$stmt2 -> prepare($this -> sql3))
            {
                echo "<script> alert('SQL ERROR: " . $stmt2 -> error . "');
</script>";
            }
            else
            {
                $stmt2 -> execute();

                echo "<p id = 'successmessage'> You have successfully signed up!
Sign in now to enter forum discussion! </p>";
            }
        }
    }
}
}
}
}

```

```

/* This method checks the new users who just signed up a new forum
account */
public function RegisteredUsers()
{
    $this -> sql4 = "SELECT reg_date, user_username FROM accounts;";

    $stmt = $this -> dbConn -> stmt_init();

    if(!$stmt -> prepare($this -> sql4))
    {
        echo "<script> alert('SQL ERROR: " . $stmt -> error . "'); </script>";
    }
    else
    {
        $stmt -> execute();
        $result = $stmt -> get_result();

        if($result -> num_rows > 0)
        {
            while($row = $result -> fetch_assoc())
            {
                echo "<tbody>
                    <td>".$row['user_username']."</td>
                    <td>".$row['reg_date']."</td>
                </tbody>";
            }
        }
        else
        {
            echo "<p style = 'color: red;'> There are no new users at this time
</p>";
        }
    }
}

```

/\* Checks users who are currently logged into the forum. If logged in, set the time they logged in.

Else, output 'Offline' \*/

```
public function setLoggedUsers($id, $username)
```

```
{
    $loggeddate = date("m/d/Y h:i A"); //when the user first logged in
```

//if the id and username matches the session variables that are set

```
if($id == $_SESSION['userid'] && $username ==
$_SESSION['username'])
```

```
{
```



```

        //indicates that the user is logged on
        $this -> sql5 = "UPDATE activeaccounts SET user_id='$id',
log_status='Active', login_time='$loggeddate' WHERE
user_username='$username'";
        $stmt1 = $this -> dbConn -> stmt_init();
        $stmt1 -> prepare($this -> sql5);
        $stmt1 -> execute();

        //indicates that the user is logged off
        $this -> sql6 = "UPDATE activeaccounts SET log_status='Offline',
login_time='Not Logged On' WHERE user_username!='$username'";
        $stmt2 = $this -> dbConn -> stmt_init();
        $stmt2 -> prepare($this -> sql6);
        $stmt2 -> execute();
    }
}

/* Display all the number of users who are logged in and those who are not
*/
public function displayLoggedUsers()
{
    $this -> sql7 = "SELECT * FROM activeaccounts;";

    $stmt = $this -> dbConn -> stmt_init();
    $stmt -> prepare($this -> sql7);
    $stmt -> execute();

    $result = $stmt -> get_result();
    if($result -> num_rows > 0)
    {
        while($row = $result -> fetch_assoc())
        {
            echo "<tbody>
                <td>".$row['user_username']."</td>
                <td>".$row['log_status']."</td>
                <td>".$row['login_time']."</td>
            </tbody>";
        }
    }
    else
    {
        echo "<p id = 'errorMessage'> There are no users currently logged on
    </p>";
    }
}

```

```

/* Displays all the recent activities of registered forum members */
public function RecentActivity()
{
    //All SQL queries will only obtain the very last row of each table

    /*This query checks who just created an account*/
    $this -> sql8 = "SELECT * FROM accounts ORDER BY user_id
DESC LIMIT 1;";
    $stmt1 = $this -> dbConn -> stmt_init();

    if(!$stmt1 -> prepare($this -> sql8))
    {
        echo "<script> alert('SQL ERROR: " . $stmt1 -> error . "');
</script>";
    }
    else
    {
        //first, execute sql8
        $stmt1 -> execute();
        $result1 = $stmt1 -> get_result();

        while($row1 = $result1 -> fetch_assoc())
        {
            echo "<br/>".$row1['user_username']." just registered a forum
account!<br/>";
        }
    }

    /*This query checks who just sent a message to another user*/
    $this -> sql9 = "SELECT * FROM messageinbox ORDER BY
user_id DESC LIMIT 1;";
    $stmt2 = $this -> dbConn -> stmt_init();

    if(!$stmt2 -> prepare($this -> sql9))
    {
        echo "<script> alert('SQL ERROR: " . $stmt2 -> error . "');
</script>";
    }
    else
    {
        //first, execute sql8
        $stmt2 -> execute();
        $result2 = $stmt2 -> get_result();

        while($row2 = $result2 -> fetch_assoc())
        {

```

```

        echo "<br/>".$row2['user_username']." just registered a forum
account!<br/>";
    }
}

/*This query checks who replied to a certain thread*/
$this -> sql10 = "SELECT * FROM replies ORDER BY user_id
DESC LIMIT 1;";
$stmt3 = $this -> dbConn -> stmt_init();

if(!$stmt3 -> prepare($this -> sql10))
{
    echo "<script> alert('SQL ERROR: " . $stmt3 -> error . "');
</script>";
}
else
{
    $stmt3 -> execute();
    $result3 = $stmt3 -> get_result();

    while($row3 = $result3 -> fetch_assoc())
    {
        echo "<br/> ".$row3['user_username']." replied to the thread:
\"\"".$row3['thread_title']."\" \"<br/>";
    }
}

/*This query checks who just created a thread*/
$this -> sql11 = "SELECT * FROM threads ORDER BY user_id
DESC LIMIT 1;";
$stmt4 = $this -> dbConn -> stmt_init();

if(!$stmt4 -> prepare($this -> sql11))
{
    echo "<script> alert('SQL ERROR: " . $stmt4 -> error . "');
</script>";
}
else
{
    //first, execute sql8
    $stmt4 -> execute();
    $result4 = $stmt4 -> get_result();

    while($row4 = $result4 -> fetch_assoc())
    {

```

```

        echo "<br/>".$row4['thread_author']."' just created a new thread
        called \"'".$row4['thread_title']."'\"!<br/>";
    }
}
}
}
?>

```

### activities.php

```

<?php
    session_start();

    include 'includes/AccountHandler.php';
    $recentactivity = new Accounts();
?>
<html>
    <head>
        <title> See recent activity of forum members! </title>

        <meta charset = "utf-8"/>

        <link rel = "stylesheet" href = "css/main.css"/>
        <link rel = "stylesheet" href = "css/Navbar.css"/>
    </head>

    <body>
        <!-- navbar -->
        <ul class = "navbar">
            <li class = "left-tabs"><a href = "forumhome.php"> Home </a></li>
            <li class = "left-tabs"><a href = "members.php"> Members </a></li>
            <li class = "left-tabs"><a href = "activities.php"> Activity </a></li>

            <li class = "right-tabs"><a href = "includes/logout.php"> Logout
        </a></li>
            <li class = "dropdown right-tabs">
                <a href = "#" class = "dropdownButton"> Hello: <?php echo
                $_SESSION['username']; ?> </a>
                <div class = "dropdown-content">
                    <a href = "#"> <?php echo "<img src = '" . $_SESSION['pic'] . '"
                    height='70' width='70'/>"; ?> </a>
                    <a href = "#"> Full Name: <?php echo $_SESSION['firstname'] .
                    " " . $_SESSION['lastname']; ?> </a>
                    <a href = "#"> Email: <?php echo $_SESSION['useremail']; ?>
                </a>
                    <a href = "#"> Current Status: <?php echo $_SESSION['status'];
                ?> </a>

```

```

        <a href = "#"> Join Date: <?php echo $_SESSION['regdate']; ?>
    </a>
        <a href = "userprofile.php"> Go to account settings </a>
    </div>
</li>
</ul>

<!-- Forum activity table -->
<div id = "forumactivity">
    <h3>Recent Activity</h3> <p>Activity stream for all registered
members</p> </th>
    <div style = "border-bottom: 1px solid white;"></div>
    <?php
        $recentactivity -> RecentActivity();
    ?>
</div>
</body>
</html>

```

This will show all the recent forum member activities, though only displays the very last result of each table.

## **8/10/19**

- Next you want to check if a username is already taken (that is, username is already in the database). If it is, output an error saying that username already exists.

In the AccountHandler.php page, add the following for the signup() method:

```

AccountHandler.php
<?php
    include_once 'database.php';

    /* This class will handle the login, logout and signup processes */
    class Accounts
    {
        private $dbConn;
        private $sql1; //For the login(). For logging in new users to the
forumhome.php

        //Both are for the signup() method. For creating new user accounts
        private $sql2; //For the signup() method. For adding new users to the
accounts table
        private $sql3; //For the signup() method. For adding new users to the
activeaccounts table

        private $sql4; //For the RegisteredUsers() method. For showing new users
who have just registered a new forum account

```

//Both are for the setLoggedUsers() method. Sets the login time, that is, when the time the user first logged in

private \$sql5; //For showing that the user is logged on

private \$sql6; //For showing that users are logged off

private \$sql7; //For the displayLoggedUsers() method. For displaying how many users are logged in and how many aren't

//All are for the RecentActivity() method. For displaying recent user forum activity

private \$sql8; //For looping through the accounts table. To see who just created an account

private \$sql9; //For looping through the messageinbox table. To see who just sent a message

private \$sql10; //For looping through the replies table. To see who just replied to a thread

private \$sql11; //For looping through the threads table. To see who just created a thread

**private \$sql12; //For the signup() method. To check if a username already exists in the accounts database**

/\* initialize connection to the discussionforum database upon calling the object of the class \*/

public function \_\_construct()

{

    \$this->dbConn = new Connection();

    \$this->dbConn = \$this->dbConn->connect();

}

/\* This method is for logging in new users into the forum \*/

public function login(\$username, \$password)

{

    if(empty(\$username) && empty(\$password))

    {

        echo "<p id = 'errormessage'> Do not submit blank information! </p>";

    }

    else if(empty(\$username))

    {

        echo "<p id = 'errormessage'> You must enter your username! </p>";

    }

    else if(empty(\$password))

    {

        echo "<p id = 'errormessage'> You must enter your password! </p>";

    }

```

else
{
    $this -> sql1 = "SELECT * FROM accounts WHERE user_username=?
OR user_password=?";

    $stmt = $this -> dbConn -> stmt_init();

    if(!$stmt -> prepare($this -> sql1))
    {
        echo "<script> alert('SQL ERROR: " . $stmt -> error . "'); </script>";
    }
    else
    {
        $stmt -> bind_param("ss", $username, $password);
        $stmt -> execute();

        $result = $stmt -> get_result();

        if($row = $result -> fetch_assoc()) //store the results in an associative
array
        {
            session_start();

            /*obtain the id, username, email, firstname, lastname as well as
profile picture
and status and registration date, and stores them in session
variables*/
            $_SESSION['userid'] = $row['user_id'];
            $_SESSION['firstname'] = $row['user_firstname'];
            $_SESSION['lastname'] = $row['user_lastname'];
            $_SESSION['useremail'] = $row['user_email'];
            $_SESSION['password'] = $row['user_password'];
            $_SESSION['username'] = $row['user_username'];
            $_SESSION['pic'] = $row['profile_pic'];
            $_SESSION['status'] = $row['user_status'];
            $_SESSION['regdate'] = $row['reg_date'];

            //checks to see if the password the user typed in matches the hashed
password in the accounts table
            if(password_verify($password, $_SESSION['password']))
            {
                //sends user to the forum homepage upon successful login
                header("Location: forumhome.php?login=successful");
                exit();
            }
            else

```

```

        {
            echo "<p id = 'errormessage'> Username exists but password is
incorrect! </p>";
        }
    }
    else
    {
        echo "<p id = 'errormessage'> There is no such account! </p>";
    }
}
}
}

```

/\* This method is for registering new users and adding their information to the accounts table \*/

```

public function signup($accountCreated, $firstname, $lastname, $email,
$username, $password)
{
    //This is to check if the username a user entered already exists in the
accounts table
    $this -> sql12 = "SELECT user_username FROM accounts where
user_username='$username'";
    $stmt3 = $this -> dbConn -> stmt_init();
    if(!$stmt3 -> prepare($this -> sql12))
    {
        echo "<script> alert('SQL ERROR: " . $stmt3 -> error . "')";
</script>";
    }
    else
    {
        $stmt3 -> execute();
        $result = $stmt3 -> get_result();

        if(empty($firstname) && empty($lastname) && empty($email) &&
empty($username) && empty($password))
        {
            echo "<p id = 'errormessage'> Do not submit empty information!
</p>";
        }
        else if(empty($firstname) || empty($lastname) || empty($email) ||
empty($username) || empty($password))
        {
            echo "<p id = 'errormessage'> You must fill out everything! </p>";
        }
        else
        {

```



```

//if a username is already in the accounts table
if($result -> num_rows == 1)
{
    echo "<p id = 'errormessage'> Username already exists! </p>";
}
//if username does not exist in the accounts table
else
{
    $this -> sql2 = "INSERT INTO accounts(reg_date, user_firstname,
user_lastname, user_email, user_username, user_password)
VALUES(?, ?, ?, ?, ?, ?)";

    $stmt1 = $this -> dbConn -> stmt_init();

    if(!$stmt1 -> prepare($this -> sql2))
    {
        echo "<script> alert('SQL ERROR: " . $stmt1 -> error . "');
</script>";
    }
    else
    {
        $hashedPwd = password_hash($password,
PASSWORD_DEFAULT);

        $stmt1 -> bind_param("ssssss", $accountCreated, $firstname,
$lastname, $email, $username, $hashedPwd);
        $stmt1 -> execute();

        //now insert the username in the activeaccounts table
        $this -> sql3 = "INSERT INTO activeaccounts(user_username)
VALUES('$username')";

        $stmt2 = $this -> dbConn -> stmt_init();

        if(!$stmt2 -> prepare($this -> sql3))
        {
            echo "<script> alert('SQL ERROR: " . $stmt2 -> error . "');
</script>";
        }
        else
        {
            $stmt2 -> execute();

            echo "<p id = 'successmessage'> You have successfully signed
up! Sign in now to enter forum discussion! </p>";
        }
    }
}

```

```

    }
    }
    }
    }
}

/* This method checks the new users who just signed up a new forum
account */
public function RegisteredUsers()
{
    $this -> sql4 = "SELECT reg_date, user_username FROM accounts;";

    $stmt = $this -> dbConn -> stmt_init();

    if(!$stmt -> prepare($this -> sql4))
    {
        echo "<script> alert('SQL ERROR: " . $stmt -> error . "'); </script>";
    }
    else
    {
        $stmt -> execute();
        $result = $stmt -> get_result();

        if($result -> num_rows > 0)
        {
            while($row = $result -> fetch_assoc())
            {
                echo "<tbody>
                    <td>".$row['user_username']."</td>
                    <td>".$row['reg_date']."</td>
                </tbody>";
            }
        }
        else
        {
            echo "<p style = 'color: red;'> There are no new users at this time
</p>";
        }
    }
}

/* Checks users who are currently logged into the forum. If logged in, set
the time they logged in.
Else, output 'Offline' */
public function setLoggedUsers($id, $username)
{

```

```

        $loggeddate = date("m/d/Y h:i A"); //when the user first logged in

        //if the id and username matches the session variables that are set
        if($id == $_SESSION['userid'] && $username ==
$_SESSION['username'])
        {
            //indicates that the user is logged on
            $this->sql5 = "UPDATE activeaccounts SET user_id='$id',
log_status='Active', login_time='$loggeddate' WHERE
user_username='$username'";
            $stmt1 = $this->dbConn->stmt_init();
            $stmt1->prepare($this->sql5);
            $stmt1->execute();

            //indicates that the user is logged off
            $this->sql6 = "UPDATE activeaccounts SET log_status='Offline',
login_time='Not Logged On' WHERE user_username!='$username'";
            $stmt2 = $this->dbConn->stmt_init();
            $stmt2->prepare($this->sql6);
            $stmt2->execute();
        }
    }

    /* Display all the number of users who are logged in and those who are not
    */

    public function displayLoggedUsers()
    {
        $this->sql7 = "SELECT * FROM activeaccounts";

        $stmt = $this->dbConn->stmt_init();
        $stmt->prepare($this->sql7);
        $stmt->execute();

        $result = $stmt->get_result();
        if($result->num_rows > 0)
        {
            while($row = $result->fetch_assoc())
            {
                echo "<tbody>
                    <td>".$row['user_username']."</td>
                    <td>".$row['log_status']."</td>
                    <td>".$row['login_time']."</td>
                </tbody>";
            }
        }
        else
    }

```

```

    {
        echo "<p id = 'errormessage'> There are no users currently logged on
    </p>";
    }
}

/* Displays all the recent activities of registered forum members */
public function RecentActivity()
{
    //All SQL queries will only obtain the very last row of each table

    /*This query checks who just created an account*/
    $this -> sql8 = "SELECT * FROM accounts ORDER BY user_id DESC
LIMIT 1;";
    $stmt1 = $this -> dbConn -> stmt_init();

    if(!$stmt1 -> prepare($this -> sql8))
    {
        echo "<script> alert('SQL ERROR: " . $stmt1 -> error . "'); </script>";
    }
    else
    {
        //first, execute sql8
        $stmt1 -> execute();
        $result1 = $stmt1 -> get_result();

        while($row1 = $result1 -> fetch_assoc())
        {
            echo "<br/>".$row1['user_username']. " just registered a forum
account!<br/>";
        }
    }

    /*This query checks who just sent a message to another user*/
    $this -> sql9 = "SELECT * FROM messageinbox ORDER BY user_id
DESC LIMIT 1;";
    $stmt2 = $this -> dbConn -> stmt_init();

    if(!$stmt2 -> prepare($this -> sql9))
    {
        echo "<script> alert('SQL ERROR: " . $stmt2 -> error . "'); </script>";
    }
    else
    {
        //first, execute sql8
        $stmt2 -> execute();
    }
}

```

```

$result2 = $stmt2 -> get_result();

while($row2 = $result2 -> fetch_assoc())
{
    echo "<br/>".$row2['user_username']." just registered a forum
account!<br/>";
}
}

/*This query checks who replied to a certain thread*/
$this -> sql10 = "SELECT * FROM replies ORDER BY user_id DESC
LIMIT 1;";
$stmt3 = $this -> dbConn -> stmt_init();

if(!$stmt3 -> prepare($this -> sql10))
{
    echo "<script> alert('SQL ERROR: " . $stmt3 -> error . "'); </script>";
}
else
{
    $stmt3 -> execute();
    $result3 = $stmt3 -> get_result();

    while($row3 = $result3 -> fetch_assoc())
    {
        echo "<br/> ".$row3['user_username']." replied to the thread:
\"".$row3['thread_title']."\" <br/>";
    }
}

/*This query checks who just created a thread*/
$this -> sql11 = "SELECT * FROM threads ORDER BY user_id DESC
LIMIT 1;";
$stmt4 = $this -> dbConn -> stmt_init();

if(!$stmt4 -> prepare($this -> sql11))
{
    echo "<script> alert('SQL ERROR: " . $stmt4 -> error . "'); </script>";
}
else
{
    //first, execute sql8
    $stmt4 -> execute();
    $result4 = $stmt4 -> get_result();

    while($row4 = $result4 -> fetch_assoc())

```

```

        {
            echo "<br/>".$row4['thread_author']." just created a new thread
called \"\"".$row4['thread_title']."\"!<br/>";
        }
    }
}
}
?>

```

14. What you now want to do is re-decorate the forum pages, that is the design of it. In order to make it neater and responsive, use Bootstrap. You can just use Bootstrap via the CDN so you don't have to go through downloading all the Bootstrap files which will only give you a headache.

In the ThreadHandler.php, add the following (note that you have to fix the navbar here for complicated reasons):

#### **ThreadHandler.php**

```

<?php
    include_once 'database.php';

    /* This class will handle the processes of both creating the topic thread of a
    particular subject
    and displaying it on the forumhome.php and the subject pages */
    class ThreadHandler
    {
        private $db;

        //SQL variables
        private $sql1; //for the createThread() method. For storing the thread
        information in the thread table.
        private $sql2; //for the displayThread() method. For obtaining the thread
        information from the thread table and displaying it in their respective topic
        thread file
        private $sql3; //for the createReply() method. For obtaining the user
        comment and storing it in the replies table
        private $sql4; //for the displayReply() method. For obtaining the user
        comment from the replies table and displaying it in the respective topic thread
        private $sql5; //for the displayThread() method. For counting how many
        replies were made to a particular topic tread
        private $sql6; //for the obtainTopicThread() method. For obtaining the last
        topic thread made to a particular subject
        private $sql7; //for the showThreads() method. Shows the threads that a
        particular user created
        private $sql8; //for the searchThreads() method. For matching a topic
        thread input to one in the thread table
    }

```

```

public function __construct()
{
    $this->db = new Connection();
    $this->db = $this->db->connect();
}

/* This method is responsible for obtaining the thread information and
storing that information in the threads table */
public function createThread($directory, $id, $author, $regdate,
$profilepic, $subject, $topic, $title, $content, $topicPage, $subjectStyling)
{
    /* Before adding the thread information into the database, first
append that information into a file called
'DiscussionWindow_(topic)_(title).php' */
    $threadfile = fopen("$directory/DiscussionWindow_$topic" . "_" . "
$title.php", "w");

    $threadPath = "$directory/DiscussionWindow_$topic" . "_" . "
$title.php"; //full thread file path
    $regularPath = "DiscussionWindow_$topic" . "_" . " $title.php";
//half thread file path
    $threadCreated = date("m/d/Y");

    //set the thread page layout for the
'DiscussionWindow_(topic)_(title).php' page
    $threadlayout = "
        <?php
            session_start();
            include '../includes/ThreadHandler.php';
        ?>
        <!Doctype html>
        <html lang = en>
        <head>
            <title> $title </title>

            <meta charset = 'utf-8'/>
            <meta name = 'viewport' content = 'width=device-
width, initial-scale=1.0'/>

            <!-- Just to style the navbar -->
            <style>
                .navbar
                {
                    border-radius: 5px;

```





```

                                <a class = 'dropdown-item' href =
'#'> <?php echo \"<img src = '../..../'."$"."upic' height='70'
width='70'/>\"; ?> </a>

```

```

                                <a class = 'dropdown-item' href =
'#'> Full Name: <?php echo "\".$"."_SESSION['firstname']".".\" \"".
\".$"."_SESSION['lastname']; ?> </a>

```

```

                                <a class = 'dropdown-item' href =
'#'> Email: <?php echo "\".$"."_SESSION['useremail']; ?> </a>

```

```

                                <a class = 'dropdown-item' href =
'#'> Current Status: <?php echo "\".$"."_SESSION['status']; ?> </a>

```

```

                                <a class = 'dropdown-item' href =
'#'> Join Date: <?php echo "\".$"."_SESSION['regdate']; ?> </a>

```

```

                                <a class = 'dropdown-item' href =
'../..../userprofile.php'> Go to account settings </a>

```

```

                                </div>

```

```

                                </li>

```

```

                                </ul>

```

```

                                </nav>

```

```

                                <!-- Thread Window -->

```

```

                                <div class = 'container'>

```

```

                                    <h2 style = 'font-family: serif;'> <b>$title</b>

```

```

                                </h2>

```

```

                                    <h5 style = 'font-family: serif;'> Created by:

```

```

                                $author </h5>

```

```

                                    <h5 style = 'font-family: serif;'> Created:

```

```

                                $threadCreated </h5>

```

```

                                    <h5 style = 'font-family: serif;'> Topic: $topic

```

```

                                </h5>

```

```

                                <div class = 'row'>

```

```

                                    <!-- Displays user information -->

```

```

                                    <div class = 'col-lg-6 col-md-6' style = 'border:
1px solid black;'>

```

```

                                            <div id = 'userPicture'> <?php echo \"<img
src = '../..../'."$profilepic.\" class = 'img-thumbnail mx-auto d-block' style =
'width: 30%; height: 65%;'/>\"; ?> </div>

```

```

                                            <p style = 'text-align: center; font-family:
serif;'> $author </p>

```

```

                                            <p style = 'text-align: center; font-family:
serif;'> Join date: '$regdate' </p>

```

```

                                            </div>

```

```

                                    <!-- Displays the thread question -->

```

```

                                    <div class = 'col-lg-6 col-md-6' style = 'border:
1px solid black;'>

```

```

<p style = 'font-family: serif;'> $content
</p>
</div>
</div>
</div>

<br/><br/><br/><br/><br/>

<!-- Replies window -->
<?php
    ".$$. "replies = new ThreadHandler();

    ".$$. "Directory = \"$threadPath\";

    ".$$. "replies -> displayReply(".$$. "Directory);
?>

<br/><br/><br/><br/><br/>

<!-- Comment window -->
<div class = 'container'>
    <div class = 'row'>
        <!-- Styling buttons (for styling buttons) -->
        <div class = 'col-lg-3 col-md-3 col-sm-3'>
            <button id = 'bold' class = 'btn btn-info'
style = 'font-family: serif;' onclick = 'styleText(\"<b>\", \"</b>\")'> <abbr
title = 'Bold'> <b>B</b> </abbr> </button>
            <button id = 'italic' class = 'btn btn-info'
style = 'font-family: serif;' onclick = 'styleText(\"<em>\", \"</em>\")'>
<abbr title = 'Italic'> <em>i</em> </abbr> </button>
            <button id = 'underline' class = 'btn btn-
info' style = 'font-family: serif;' onclick = 'styleText(\"<u>\", \"</u>\")'>
<abbr title = 'Underline'> <u>U</u> </abbr> </button><br/>
            <button id = 'subscript' class = 'btn btn-
info' style = 'font-family: serif;' onclick = 'styleText(\"<sub>\",
\"</sub>\")'> <abbr title = 'Subscript'> A<sub>2</sub> </button>
            <button id = 'superscript' class = 'btn btn-
info' style = 'font-family: serif;' onclick = 'styleText(\"<sup>\",
\"</sup>\")'> <abbr title = 'Superscript'> A<sup>2</sup> </abbr>
</button>
            <button id = 'quote' class = 'btn btn-info'
style = 'font-family: serif;' onclick = 'styleText(\"<blockquote>\",
\"</blockquote>\")'> <abbr title = 'Quote'> \"Q\" </abbr> </button>
        </div>
        <!-- Comment box -->
        <div class = 'col-lg-9 col-md-9 col-sm-9'>

```

```

        <form method = 'POST' action =
'$regularPath'>
            <textarea class = 'form-control' style =
'font-family: serif; resize: none;' name = 'commentbox' id = 'commentbox'
rows = '15' cols = '90'></textarea> <br/>
            <input type = 'submit' class = 'btn btn-
info mx-auto d-block' style = 'font-family: serif;' name = 'submit-comment'
value = 'Submit comment'/>
        </form>
        <?php
            if(isset(".$$. "_POST['submit-
comment']"))
            {
                ".$$. "Replies = new
ThreadHandler();

                ".$$. "uid =
".$$. "_SESSION['userid']; //id of the user who made the comment/reply
                ".$$. "uname =
".$$. "_SESSION['username']; //username of the user who made the
comment/reply
                ".$$. "upic = ".$$. "_SESSION['pic'];
//profile pic of user who made the comment/reply
                ".$$. "Title = \"\$title\"; //title of the
thread who the user is replying to
                ".$$. "Directory = \"\$threadPath\";
//full file path of the thread file
                ".$$. "replyCreated = date('m/d/Y');
//date the reply was created
                ".$$. "Reply =
".$$. "_POST['commentbox']; //the reply

                ".$$. "Replies ->
createReply('".$$. "uid, ".$$. "uname, ".$$. "upic, ".$$. "Title,
".$$. "Directory, ".$$. "replyCreated, ".$$. "Reply);
            }
        ?>
    </div>
</div>
</div>

    <!-- Link the Bootstrap 4 CDN JS and jQuery files
(must be in this order: jQuery, Popper, Bootstrap.js) -->
    <script
src='https://ajax.googleapis.com/ajax/libs/jquery/3.4.1/jquery.min.js'></scr
ipt>

```

```
        <script  
src='https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd/popper.mi  
n.js'></script>
```

```
        <script  
src='https://maxcdn.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js'  
></script>
```

```
        <!-- To style the text -->  
        <script src =  
'../ThreadsJSFiles/DiscussionWindow.js'></script>
```

```
    </body>  
</html>";
```

```
    fwrite($threadfile, $threadlayout);
```

```
    /* Now add the thread information to the database */  
    $this -> sql1 = "INSERT INTO threads(user_id, thread_author,  
thread_subject, thread_topic, thread_title, thread_content, thread_file,  
thread_creation)  
VALUES('$id', '$author', '$subject', '$topic', '$title', '$content',  
'$threadPath', '$threadCreated');";
```

```
    $stmt1 = $this -> db -> stmt_init();
```

```
    if(!$stmt1 -> prepare($this -> sql1))  
    {  
        echo "<script> alert('SQL ERROR: ". $stmt1 -> error . "'); </script>";  
    }  
    else  
    {  
        $stmt1 -> execute();  
        header("Location: $topicPage?threadcreation=success");  
    }  
}
```

```
/* This method is responsible for displaying the topic thread on the  
forumhome.php page and the subject pages */
```

```
public function displayThread($subject, $topic)  
{  
    /* This query is for displaying each topic thread of a particular subject */  
    $this -> sql2 = "SELECT thread_author, thread_subject, thread_topic,  
thread_title, thread_file  
FROM threads  
WHERE thread_subject='$subject' AND  
thread_topic='$topic';";
```

```

$stmt2 = $this->db->stmt_init();

if(!$stmt2->prepare($this->sql2))
{
    echo "<script> alert('SQL ERROR: " . $stmt2->error . "'); </script>";
}
else
{
    $stmt2->execute();
    $result1 = $stmt2->get_result();

    if($result1->num_rows > 0)
    {
        while($row1 = $result1->fetch_assoc()) //stores the results of the
thread table query in an associative array and stores it in a variable called
'$row1'
        {
            $dir = $row1['thread_file']; //obtains the path of the topic thread at
hand

            /* This query is for obtaining the number of replies made to a
particular topic thread */
            $this->sql5 = "SELECT COUNT(replies_id) AS 'count' FROM
replies WHERE thread_file='$dir'"; //counts how many replies were made to a
topic thread

            $stmt5 = $this->db->stmt_init();

            if(!$stmt5->prepare($this->sql5))
            {
                echo "<script> alert('SQL ERROR: " . $stmt5->error . "');
</script>";
            }
            else
            {
                $stmt5->execute();
                $result2 = $stmt5->get_result();

                while($row2 = $result2->fetch_assoc()) //stores the results of the
replies table query in an associative array and stores it in a variable called
'$row2'
                {
                    echo "<tr>
                        <td><a href = '". $row1['thread_file'] ."'>" .
$row1['thread_title'] . "</a></td>

```

```

        <td>" . $row2['count'] . "</td>
        <td> By: <em>" . $row1['thread_author'] . "</em> </td>
    </tr>";
    }
    }
    }
    }
    else
    {
        echo "<b id = 'warningMessage'> There are no current threads posted.
    </b>";
    }
    }
}

/* This method creates the comment a user made on a particular topic
thread */
public function createReply($id, $username, $pic, $title, $directory,
$dateCreated, $comment)
{
    /* Store the comment in the database with the respective thread file */
    if(empty($comment))
    {
        echo "<p id = 'errormessage'> You must enter a comment! </p>";
    }
    else
    {
        $this -> sql3 = "INSERT INTO replies(user_id, user_username,
profile_pic, thread_title, thread_file, replies_creation, replies_content)
VALUES('$id', '$username', '$pic', '$title', '$directory',
'$dateCreated', '$comment')";

        $stmt3 = $this -> db -> stmt_init();

        if(!$stmt3 -> prepare($this -> sql3))
        {
            echo "<script> alert('SQL ERROR: " . $stmt3 -> error . "'); </script>";
        }
        else
        {
            $stmt3 -> execute();
            echo "<script> location.assign(location.href); </script>"; //refreshes
current thread page to show replies as they are created
        }
    }
}
}

```

```

/* This method displays the comments that users made on a particular topic
thread */
public function displayReply($directory)
{
    /* Get only the replies of that particular topic thread */
    $this -> sql4 = "SELECT user_username, user_status, profile_pic,
thread_file, replies_creation, replies_content
FROM replies
WHERE thread_file = '$directory'";

    $stmt4 = $this -> db -> stmt_init();

    if(!$stmt4 -> prepare($this -> sql4))
    {
        echo "<script> alert('SQL ERROR: ". $stmt4 -> error . "'); </script>";
    }
    else
    {
        $stmt4 -> execute();

        $result = $stmt4 -> get_result();

        if($result -> num_rows > 0)
        {
            while($row = $result -> fetch_assoc())
            {
                echo "<div class = 'container'>
                <div class = 'row'>
                <!-- Displays user information -->
                <div class = 'col-lg-6 col-md-6' style = 'border: 1px
solid black;'>
                    <img class = 'img-thumbnail mx-auto d-block' src =
'" . "../.." . $row['profile_pic'] . '" style = 'width: 30%; height: 65%;'>
                <br/>
                    <p style = 'text-align: center; font-family: serif;'> " .
                $row['user_username'] . " </p>
                    <p style = 'text-align: center; font-family: serif;'>
                Current Status: " . $row['user_status'] . " </p>
                </div>
                <!-- Displays the thread question -->
                <div class = 'col-lg-6 col-md-6' style = 'border: 1px
solid black;'>
                    <p style = 'font-family: serif;'> " .
                $row['replies_content'] . " </p>
                </div>
            }
        }
    }
}

```

```

        </div>
    </div>";
    }
}
else
{
    echo "<b id = 'warningMessage'> There are no current replies posted.
</b>";
}
}
}
}

```

/\* This method obtains the last topic thread that was made on a particular subject \*/

```
public function obtainTopicThread($subject, $topic)
```

```
{
```

/\* This SQL query selects the most recent topic thread of a particular subject.

The most recent one has a high thread id number, thus the thread title for that

thread must also be high as well \*/

```
$this -> sql6 = "SELECT MAX(thread_id), thread_author,
thread_subject, thread_topic, MAX(thread_title), thread_file,
MAX(thread_creation)
```

```
FROM threads WHERE thread_subject='$subject' AND
thread_topic='$topic';";
```

```
$stmt6 = $this -> db -> stmt_init();
```

```
if(!$stmt6 -> prepare($this -> sql6))
```

```
{
```

```
    echo "<script> alert('SQL ERROR: ". $stmt6 -> error . "'); </script>";
```

```
}
```

```
else
```

```
{
```

```
    $stmt6 -> execute();
```

```
$result = $stmt6 -> get_result();
```

```
while($row = $result -> fetch_assoc())
```

```
{
```

```
    if($row['MAX(thread_id)'] == null && $row['thread_author'] == null
    && $row['thread_subject'] == null
```

```
        && $row['thread_topic'] == null && $row['MAX(thread_title)'] ==
null && $row['thread_file'] == null
```

```
        && $row['MAX(thread_creation)'] == null)
```



```

        {
            echo "<p> There are no threads posted </p>";
        }
        else
        {
            echo "<p>
                <a href = 'Subjects/$subject/'. $row['thread_file'] .' "> ".
                $row['MAX(thread_title)] .' </a> </br>
                by <em> ". $row['thread_author'] ." </em> <br/>
                Created on: ". $row['MAX(thread_creation)] ."
                </p>";
        }
    }
}

/* This method shows the threads that a user created */
public function showThreads($id)
{
    $this -> sql7 = "SELECT thread_title, thread_subject, thread_topic,
thread_creation, thread_file FROM threads WHERE user_id='$id'";

    $stmt7 = $this -> db -> stmt_init();

    if(!$stmt7 -> prepare($this -> sql7))
    {
        echo "<script> alert('SQL ERROR: ". $stmt7 -> error ."); </script>";
    }
    else
    {
        $stmt7 -> execute();

        $result = $stmt7 -> get_result();

        if($result -> num_rows == 0)
        {
            echo "<script>
                var threadwindow =
document.getElementById('threadsCreated');
                threadwindow.style.display = 'none';

                var Notice = document.createElement('p');
                var textnode = document.createTextNode('No threads for this
topic have been posted');

                var notice = Notice.appendChild(textnode);

```

```

        document.getElementById('user-posts').appendChild(notice);
    </script>";
}
else
{
    while($row = $result -> fetch_assoc())
    {
        echo "<tr>
            <td><a href =
'Subjects/' . $row['thread_subject'] . '/' . $row['thread_file'] . ">" . $row['thread_title'
]. "</a></td>
            <td>" . $row['thread_subject'] . "</td>
            <td>" . $row['thread_topic'] . "</td>
            <td>" . $row['thread_creation'] . "</td>
        </tr>";
    }
}
}

/* This method is for searching for a topic thread on a particular subject in
the threads table */
public function searchThreads($threadname, $subjectname)
{
    if(empty($threadname) && empty($subjectname))
    {
        echo "<p style = 'color: red; font-family: serif; text-align: center;'> Do
not just leave anything blank and submit it! </p>";
    }
    else
    {
        $this -> sql8 = "SELECT * FROM threads WHERE
thread_title='$threadname' AND thread_subject='$subjectname'";

        $stmt = $this -> db -> stmt_init();

        if(!$stmt -> prepare($this -> sql8))
        {
            echo "<script> alert('". $stmt -> error. "'); </script>";
        }
        else
        {
            $stmt -> execute();
            $result = $stmt -> get_result();

```



8/20/19

15. For some reason, when typing a comment in the comment box, when hitting the enter to create a new line, when added to the database, it will store it as a single line. You want to fix this: when user creates a comment like this:

Hello world It's me
------------------------

You don't want it to be displayed like this:

Hello worldIt's me
--------------------

You want it to be displayed as before.

To do this, simply create a JavaScript function that will convert any space and/or new line as space break (&nbsp;) and/or new line (\n) respectively. Upon clicking submit, the space breaks and/or new lines will be stored in the database, and later what you typed in the textarea will be displayed as it was typed.

In the DiscussionWindow.js, add the following function:

<p><b><u>DiscussionWindow.js</u></b></p> <pre>//use the window.getSelection() method to obtain highlighted text in the comment box //Use following link as reference: https://jsfiddle.net/qpw1eemr/1/  function styleText(x, y) {     var commentbox = document.getElementById("commentbox"); //obtains the textarea comment box      var startText = commentbox.selectionStart; //gets the beginning of the selected portion of the comment text in the textarea     var endText = commentbox.selectionEnd; //gets the end of the selected portion of the comment text in the textarea     var allText = commentbox.value; //gets all the comment text input from the comment box textarea      var selectedText = allText.substring(startText, endText); //obtains the selected text from start letter to end letter     var newText; //the new comment input to be styled      /* inside the newText variable translates to: "go from the beginning text all the way to just before the beginning of the selected text,     then add a beginning style tag (x) before the beginning of the selected text,     then add the end style tag (y) after the end of the selected text,     then continue on from the end of the selected text to the end of the text" */     if(x == "&lt;b&gt;" &amp;&amp; y == "&lt;/b&gt;") //bold the selected text</pre>
---

```

    {
        newText = allText.substring(0, startText) + "<b>" + selectedText + "</b>"
+ allText.substring(endText, allText.length);
    }
    else if(x == "<em>" && y == "</em>") //italicize the selected text
    {
        newText = allText.substring(0, startText) + "<em>" + selectedText +
"</em>" + allText.substring(endText, allText.length);
    }
    else if(x == "<u>" && y == "</u>") //underline the selected text
    {
        newText = allText.substring(0, startText) + "<u>" + selectedText + "</u>"
+ allText.substring(endText, allText.length);
    }
    else if(x == "<sub>" && y == "</sub>") //subscript the selected text
    {
        newText = allText.substring(0, startText) + "<sub>" + selectedText +
"</sub>" + allText.substring(endText, allText.length);
    }
    else if(x == "<sup>" && y == "</sup>") //superscript the selected text
    {
        newText = allText.substring(0, startText) + "<sup>" + selectedText +
"</sup>" + allText.substring(endText, allText.length);
    }
    else if(x == "<blockquote>" && y == "</blockquote>") //quote a text from
another user
    {
        newText = allText.substring(0, startText) + "<blockquote style =
'background-color: lightgrey;'" + selectedText + "</blockquote>" +
allText.substring(endText, allText.length);
    }

    //replace the original comment input with the comment input to be styled with
the above options
    commentbox.value = newText;
}

```

**function addNewLines()**

```

{
    var comment = document.getElementById('commentbox');

    var commentvalue = comment.value;
    commentvalue = commentvalue.replace(/ /g, "&nbsp;"); //replace
    spacings with non-breaking spaces
    commentvalue = commentvalue.replace(/\n/g, "<br/>"); //replace new
    lines with breaks
}

```

```

        comment.value = commentvalue; //replace the text in the commentbox
        with the text with the line breaks and spaces
    }

```

Now, in the ThreadHandler.php file, add the following bolded in the comment window code:

### ThreadHandler.php

```

        <!-- Comment window -->
        <div class = 'container'>
            <div class = 'row'>
                <!-- Styling buttons (for styling buttons) -->
                <div class = 'col-lg-3 col-md-3 col-sm-3'>
                    <button id = 'bold' class = 'btn btn-info' style =
'font-family: serif;' onclick = 'styleText(\"<b>\", \"</b>\")'> <abbr title =
'Bold'> <b>B</b> </abbr> </button>
                    <button id = 'italic' class = 'btn btn-info' style
= 'font-family: serif;' onclick = 'styleText(\"<em>\", \"</em>\")'> <abbr title =
'Italic'> <em>i</em> </abbr> </button>
                    <button id = 'underline' class = 'btn btn-info'
style = 'font-family: serif;' onclick = 'styleText(\"<u>\", \"</u>\")'> <abbr title =
'Underline'> <u>U</u> </abbr> </button><br/>
                    <button id = 'subscript' class = 'btn btn-info'
style = 'font-family: serif;' onclick = 'styleText(\"<sub>\", \"</sub>\")'> <abbr
title = 'Subscript'> A<sub>2</sub> </button>
                    <button id = 'superscript' class = 'btn btn-info'
style = 'font-family: serif;' onclick = 'styleText(\"<sup>\", \"</sup>\")'> <abbr
title = 'Superscript'> A<sup>2</sup> </abbr> </button>
                    <button id = 'quote' class = 'btn btn-info' style
= 'font-family: serif;' onclick = 'styleText(\"<blockquote>\",
\"</blockquote>\")'> <abbr title = 'Quote'> \"Q\" </abbr> </button>
                </div>
                <!-- Comment box -->
                <div class = 'col-lg-9 col-md-9 col-sm-9'>
                    <form method = 'POST' action =
'$regularPath'>
                        <textarea class = 'form-control' style =
'font-family: serif; resize: none;' name = 'commentbox' id = 'commentbox' rows
= '15' cols = '90'></textarea> <br/>
                        <input type = 'submit' class = 'btn btn-info
mx-auto d-block' style = 'font-family: serif;' name = 'submit-comment' value =
'Submit comment' onclick = 'addNewLines()'>
                    </form>
                <?php
                    if(isset(\"$_POST['submit-comment']\"))
                    {

```

```

        ".$"."Replies = new ThreadHandler();

        ".$"."uid = ".$"."_SESSION['userid']; //id
of the user who made the comment/reply
        ".$"."uname =
".$"."_SESSION['username']; //username of the user who made the
comment/reply
        ".$"."upic = ".$"."_SESSION['pic'];
//profile pic of user who made the comment/reply
        ".$"."Title = \"$title\"; //title of the thread
who the user is replying to
        ".$"."Directory = \"$threadPath\"; //full
file path of the thread file
        ".$"."replyCreated = date('m/d/Y'); //date
the reply was created
        ".$"."Reply =
".$"."_POST['commentbox']; //the reply

        ".$"."Replies -> createReply($(".$"."uid,
".$"."uname, ".$"."upic, ".$"."Title, ".$"."Directory, ".$"."replyCreated,
".$"."Reply);
    }
    ?>
</div>
</div>
</div>

```

Now, whenever a user types in a reply and adds a new line and space, the comment will be displays as it was typed in.

**8/25/19**

Now that everything is done, what you now want to do is:

1. Re-decorate the entire thread page (including the navbar, the thread window that displays the main question, and the replies window) to make it look more user friendly.

### **ThreadHandler.php**

```
...
/* This method is responsible for obtaining the thread information and storing
that information in the threads table */
public function createThread($directory, $id, $author, $regdate,
$profilepic, $subject, $topic, $title, $content, $topicPage, $subjectStyling)
{
    /* Before adding the thread information into the database, first append
that information into a file called 'DiscussionWindow_(topic)_(title).php' */
    $threadfile = fopen("$directory/DiscussionWindow_$topic" . "_" . "
$title.php", "w");

    $threadPath = "$directory/DiscussionWindow_$topic" . "_" . "
$title.php"; //full thread file path
    $regularPath = "DiscussionWindow_$topic" . "_" . " $title.php"; //half
thread file path
    $threadCreated = date("m/d/Y");

    //set the thread page layout for the
'DiscussionWindow_(topic)_(title).php' page
    $threadlayout = "
        <?php
            session_start();
            include '../includes/ThreadHandler.php';
        ?>
        <!Doctype html>
        <html lang = en>
        <head>
            <title> $title </title>

            <meta charset = 'utf-8'/>
            <meta name = 'viewport' content = 'width=device-
width, initial-scale=1.0'/>

            <!-- Just to style the navbar -->
            <style>
                .nav-item:hover
                {
                    background-color: blue;
                }
                .nav-link, .card-header, .card-text
                {
```



```

        font-family: serif;
    }
</style>

<!-- Link the Bootstrap 4 CDN CSS files -->
<link rel='stylesheet'
href='https://maxcdn.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css'>
</head>

<body id = '$subject' class = '$subjectStyling'>
    <!-- Navbar -->
    <nav class = 'navbar navbar-expand-sm bg-dark navbar-
dark fixed-top' style = 'border-radius: 10px;'>
        <ul class = 'navbar-nav mr-auto'>
            <li class = 'nav-item'>
                <a href = '../..../forumhome.php' class = 'nav-
link text-white'> Home </a>
            </li>
            <li class = 'nav-item'>
                <a href = '../..../members.php' class = 'nav-link
text-white'> Members </a>
            </li>
            <li class = 'nav-item'>
                <a href = '../..../activities.php' class = 'nav-link
text-white'> Activity </a>
            </li>
        </ul>
        <ul class = 'navbar-nav ml-auto'>
            <li class = 'nav-item dropdown'>
                <a href = '#' class = 'nav-link dropdown-toggle
text-white' data-toggle = 'dropdown'> Hello: <?php echo
"."$"."_SESSION['username']; ?> </a>
                <div class = 'dropdown-menu'>
                    <span class = 'dropdown-link d-flex justify-
content-center' style = 'padding: 10px;'> <?php echo "\"<img src =
'../..../".$"."_SESSION['pic'].\"' height='70' width='70'/>\"; ?> </span>
                    <span class = 'dropdown-link d-flex justify-
content-center' style = 'padding: 10px;'> <?php echo
"."$"."_SESSION['firstname'].\".\".\"$"."_SESSION['lastname']; ?>
</span>
                    <span class = 'dropdown-link d-flex justify-
content-center' style = 'padding: 10px;'> Email: <?php echo
"."$"."_SESSION['useremail']; ?> </span>
                    <span class = 'dropdown-link d-flex justify-
content-center' style = 'padding: 10px;'> Current Status: <?php echo
"."$"."_SESSION['status']; ?> </span>
                </div>
            </li>
        </ul>
    </nav>

```

```

                                <span class = 'dropdown-link d-flex justify-
content-center' style = 'padding: 10px;'> Join date: <?php echo
"."$"."_SESSION['regdate']; ?> </span>
                                <a href = '../../userprofile.php' class =
'dropdown-link btn btn-primary justify-content-center'> Go to account settings
</a>

                                </div>
                                </li>
                                <li class = 'nav-item'>
                                <a href = '../../includes/logout.php' class =
'nav-link text-white'> Logout </a>
                                </li>
                                </ul>
                                </nav>

                                <!-- Thread Window -->
                                <div class = 'container' style = 'margin-top: 10%;'>
                                <div class = 'row' style = 'margin-right: 10px;
margin-left: 10px; padding-right: 10%;'>
                                <!-- User information -->
                                <div class = 'col'>
                                <div class = 'card' style = 'width: 150px; float:
right;'>

                                <img class = 'card-img-top' src =
'../../".$profilepic."'/>
                                <div class = 'card-header' style = 'text-
align: center;'> $author </div>
                                <p class = 'card-text' style = 'text-align:
center;'> Join date: $regdate </p>
                                </div>
                                </div>

                                <!-- Topic thread question -->
                                <div class = 'col' style = 'border-left: 1px dotted
black; background-color: #FFF8DC;'>
                                <h2 style = 'font-family: serif; text-align:
center;'> <b>$title</b> </h2>
                                <h6 style = 'font-family: serif; text-align:
center;'> Created by: $author </h6>
                                <h6 style = 'font-family: serif; text-align:
center;'> Created on: $threadCreated </h6>
                                <h6 style = 'font-family: serif; text-align:
center;'> Topic: $topic </h6>
                                <div class = 'dropdown-divider'></div>

                                <p style = 'font-family: serif;'> $content </p>

```

```

        </div>
    </div>
</div>

<br/><br/><br/>

<div class = 'container'>
    <h4 style = 'text-align: center; font-family: serif;'>
Replies </h4>
    <div style = 'border-bottom: 1px groove
black;'></div>
</div>

<!-- Replies window -->
<?php
    ". "$" ."replies = new ThreadHandler();

    ". "$" ."Directory = \"$threadPath\";

    ". "$" ."replies -> displayReply("$. "$" ."Directory);
?>

<br/><br/><br/><br/><br/>

<!-- Comment box -->
<div class = 'container' style = 'margin-top: 15%;
margin-bottom: 10px;'>
    <form method = 'POST' action = '$regularPath'>
        <textarea name = 'commentbox' id =
'commentbox' style = 'font-family: serif; resize: none;' class = 'form-control'
rows = '10' cols = '5'></textarea>
        <div class = 'd-flex justify-content-center' style =
'margin-top: 10px; margin-bottom: 10px;'>
            <button id = 'bold' type = 'button' class = 'btn
btn-info' style = 'font-family: serif; margin-left: 10px;' onclick =
'styleText(\"<b>\", \"</b>\")'> <abbr title = 'Bold'> <b>B</b> </abbr>
</button>

            <button id = 'italic' type = 'button' class = 'btn
btn-info' style = 'font-family: serif; margin-left: 10px;' onclick =
'styleText(\"<em>\", \"</em>\")'> <abbr title = 'Italic'> <em>i</em> </abbr>
</button>

            <button id = 'underline' type = 'button' class =
'btn btn-info' style = 'font-family: serif; margin-left: 10px;' onclick =
'styleText(\"<u>\", \"</u>\")'> <abbr title = 'Underline'> <u>U</u> </abbr>
</button>

```

```
<button id = 'subscript' type = 'button' class =  
'btn btn-info' style = 'font-family: serif; margin-left: 10px;' onclick =  
'styleText(\"<sub>\", \"</sub>\")'> <abbr title = 'Subscript'> A<sub>2</sub>  
</button>
```

```
<button id = 'superscript' type = 'button' class  
= 'btn btn-info' style = 'font-family: serif; margin-left: 10px;' onclick =  
'styleText(\"<sup>\", \"</sup>\")'> <abbr title = 'Superscript'> A<sup>2</sup>  
</abbr> </button>
```

```
<button id = 'quote' type = 'button' class = 'btn  
btn-info' style = 'font-family: serif; margin-left: 10px;' onclick =  
'styleText(\"<blockquote>\", \"</blockquote>\")'> <abbr title = 'Quote'> \"Q\"  
</abbr> </button>
```

```
</div>
```

```
<input type = 'submit' name = 'submit-comment'  
class = 'btn btn-primary form-control' style = 'font-family: serif;' value =  
'Submit reply' onclick = 'addNewLines()'/>
```

```
</form>
```

```
<?php
```

```
if(isset(\".$\".\"_POST['submit-comment']))
```

```
{
```

```
\".$\".\"Replies = new ThreadHandler();
```

```
\".$\".\"uid = \".$\".\"_SESSION['userid']; //id of the  
user who made the comment/reply
```

```
\".$\".\"uname = \".$\".\"_SESSION['username'];
```

```
//username of the user who made the comment/reply
```

```
\".$\".\"upic = \".$\".\"_SESSION['pic']; //profile
```

```
pic of user who made the comment/reply
```

```
\".$\".\"Title = \".$title\"; //title of the thread who
```

```
the user is replying to
```

```
\".$\".\"Directory = \".$threadPath\"; //full file path
```

```
of the thread file
```

```
\".$\".\"replyCreated = date('m/d/Y'); //date the
```

```
reply was created
```

```
\".$\".\"Reply = \".$\".\"_POST['commentbox'];
```

```
//the reply
```

```
\".$\".\"Replies -> createReply(\".$\".\"uid,  
\".$\".\"uname, \".$\".\"upic, \".$\".\"Title, \".$\".\"Directory, \".$\".\"replyCreated,  
\".$\".\"Reply);
```

```
}
```

```
?>
```

```
</div>
```

```
<!-- Link the Bootstrap 4 CDN JS and jQuery files  
(must be in this order: jQuery, Popper, Bootstrap.js) -->
```

```

<script
src='https://ajax.googleapis.com/ajax/libs/jquery/3.4.1/jquery.min.js'></script>
<script
src='https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd/popper.min.js'
></script>

<script
src='https://maxcdn.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js'></scr
ipt>

<!-- To style the text -->
<script src =
'../ThreadsJSFiles/DiscussionWindow.js'></script>
</body>
</html>";

```

```

fwrite($threadfile, $threadlayout);

```

```

/* Now add the thread information to the database */
$this -> sql1 = "INSERT INTO threads(user_id, thread_author,
thread_subject, thread_topic, thread_title, thread_content, thread_file,
thread_creation)
VALUES('$id', '$author', '$subject', '$topic', '$title', '$content',
'$threadPath', '$threadCreated');"

```

```

$stmt1 = $this -> db -> stmt_init();

```

```

if(!$stmt1 -> prepare($this -> sql1))
{
    echo "<script> alert('SQL ERROR: ". $stmt1 -> error . "'); </script>";
}
else
{
    $stmt1 -> execute();
    header("Location: $topicPage?threadcreation=success");
}
}

```

```

...

```

(one problem here is the profile picture won't save in the thread window when the user of the thread changes their profile picture, but this isn't a big deal)

**8/27/19**

2. What you want to do now is revamp the navbar on all pages using Bootstrap, the same as how it looks like in the topic thread page as shown above. Also, make the navbar stick to the top so that when the user scrolls down the page, the navbar “sticks” to the page.

(Keep in mind that you are only using Bootstrap for this specific part of the re-decoration process. There is no need to re-decorate all the parts of the forum, including the tables, backgrounds, etc, because they are already perfect as they are. You just want to revamp the navbar to make it more responsive. Plus, re-decorating the entire layout using Bootstrap is so much work and a real headache to make sure everything is responsive and not “glitchy”)

**8/28/19**

3. Continue with re-decorating the website layout, and check every single page to make sure that there are no PHP code errors and that there are no fatal errors. Also delete any unnecessary parts of your website.

**9/4/19 – 9/7/19**

4. You want to fix the ProfileHandler.php and userprofile.php page, mainly the form parts. Make the user information inputs separate and also make individual methods for each user information change. You also want to re-decorate it with Bootstrap. Remember, you don’t have to re-decorate the entire forum website with Bootstrap, just re-decorate the parts where you also have to fix the PHP code also.  
For the ProfileHandler.php, you want to change the entire code because of problems with the PHP header function. Instead of user the header function, after the user updates their information, you want to update the session variables, then reset the page using JavaScript so that change can be shown immediately. You also want to create individual update inputs because the previous implementation had so many if-else statements whenever a user wanted to change specific information.

**ProfileHandler.php**

```
<?php
    include_once 'database.php';

    /* This class updates user information such as name, email and profile
    picture */
    class ProfileHandler
    {
        private $db; //to connect to the discussionforum database

        /* These variables are reserved for the uploadPics() method */
        private $sql1; //for uploading profile pictures
        private $photo = ""; //obtains the photo to be uploaded (initialize to nothing
        until user selects a picture to upload)
```

```

        private $imageFileType = ""; //obtain the type of file (".jpg", ".png", etc...)
        (initialize to nothing until user selects a picture to upload)
        private $folder; //the folder to store the uploaded pictures
        private $check; //checks the size of the image

        /* These variables are reserved for the updateFname(),
        updateLname(), updateUname(), updatePwd(), updateEmail() and
        updateStatus() methods */
        private $sql2; //for updating user information
        private $newUsername = ""; //sets new user name
        private $newPassword = ""; //sets new user password
        private $newEmail = ""; //sets new user email
        private $setStatus = ""; //sets new user status

        /* These variables are reserved for the PrivateMessage() method */
        private $sql3; //for checking if a username the user sending the PM actually
        exists in the accounts table
        private $sql4; //if username exists, add the information into the
        MessageBox table

        /* These variables are reserved for the DisplayMessages1() method */
        private $sql5; //for displaying the messages that the current logged in user
        sent to other members

        /* These variables are reserved for the DisplayMessage2() method */
        private $sql6; //for displaying the messages that other members has sent to
        the current logged in user

        /* Initialize connection to the discussionforum database upon creating the
        object of this class */
        public function __construct()
        {
            $this -> db = new Connection();
            $this -> db = $this -> db -> connect();
        }

        /* This method uploads new profile pictures */
        /* This will update the profile_pic column in both the accounts and
        replies table */
        public function uploadPics($id)
        {
            $this -> folder = "ProfilePics/";
            $this -> photo = $this -> folder .
            basename($_FILES["getFile"]["name"]); //obtains the photo to be
            updated and store them in the ProfilePics folder

```

```

    $this->imageFileType = strtolower(pathinfo($this->photo,
PATHINFO_EXTENSION)); //obtain the path of the type of file ('.jpg',
'.png', etc...)
    $this->check = getimagesize($_FILES['getFile']['tmp_name']);
//checks the size of the image

    //check if the file is a real image or not
    if($this->check !== false)
    {
        //if the image file is not jpg, jpeg, png and gif files
        if($this->imageFileType != "jpg" && $this->imageFileType !=
"png" && $this->imageFileType != "jpeg" && $this->imageFileType
!= "gif")
        {
            echo "<p id = 'errMessage'> Sorry, only JPG, JPEG, PNG, and
GIF files are allowed! </p>";
        }
        else //else if the image file is a jpg, jpeg, png, and/or gif
        {
            //if the file is too large
            if($_FILES['getFile']['size'] > 500000)
            {
                echo "<p id = 'errMessage'> Sorry, your file is too big to
upload! </p>";
            }
            else //else if file is not too large
            {
                $pic = $this->photo; //this variable is just a placeholder to hold
the photo variable

                //upload the image
                if(move_uploaded_file($_FILES['getFile']['tmp_name'], $this
->photo))
                {
                    /* update the profile_pic columns in both the accounts and
replies table */
                    $this->sql1 = "UPDATE accounts SET profile_pic='$pic'
WHERE user_id='$id'";
                    $this->sql1 .= "UPDATE replies SET profile_pic='$pic'
WHERE user_id='$id'";

                    if(!$this->db->multi_query($this->sql1))
                    {
                        echo "<p id = 'errMessage'> MULTI QUERY ERROR: ".
$this->db->error. " </p>";
                    }
                }
            }
        }
    }

```



```

        else
        {
            echo "<p id = 'succMessage'> Your picture was successfully
uploaded! </p>";
            $_SESSION['pic'] = $pic; //update the profile picture
session variable to have it change right away after uploading new pic
            echo "<script> setTimeout(function(){
location.assign(location.href); }, 1000) </script>"; //refresh the page in 1
second to reveal the changed session variables
        }
    }
    //if the image could not be uploaded
    else
    {
        echo "<p id = 'errMessage'> Sorry, there was a problem
uploading your image file </p>";
    }
}
}
else
{
    echo "<p id = 'errMessage'> The image you selected is not a file!
</p>";
}
}

/* The following methods update the firstname, lastname, username,
password, email and status of users */
/* This method updates the firstname of the user */
public function updateFname($userid, $newFname)
{
    if(empty($newFname))
    {
        echo "<p class = 'text-danger' style = 'font-family: serif; margin-
left: 10px; margin-top: 5px;'> Enter a new firstname, else don't submit
blank info! </p>";
    }
    else
    {
        $this->sql2 = "UPDATE accounts SET
user_firstname='$newFname' WHERE user_id='".$userid."'";

        $stmt = $this->db->stmt_init();
        if(!$stmt->prepare($this->sql2))
        {

```

```

        echo "<p class = 'text-danger' style = 'font-family: serif; margin-
left: 10px; margin-top: 5px;'> SQL ERROR: ".$this->db->error."
</p>";
    }
    else
    {
        $stmt->execute();

        $_SESSION['firstname'] = $newFname; //update the session
variable to have it displayed right away
        echo "<p class = 'text-success' style = 'font-family: serif; margin-
left: 10px; margin-top: 5px;'> Firstname has been updated! </p>";
        echo "<script> setTimeout(function(){
location.assign(location.href); }, 1000) </script>"; //refresh the page in 1
second to reveal the changed session variables
    }
}
}
/* This method updates the lastname of the user */
public function updateLname($userid, $newLname)
{
    if(empty($newLname))
    {
        echo "<p class = 'text-danger' style = 'font-family: serif; margin-
left: 10px; margin-top: 5px;'> Enter a new lastname, else don't submit
blank info! </p>";
    }
    else
    {
        $this->sql2 = "UPDATE accounts SET
user_lastname='".$newLname.'" WHERE user_id='".$userid.'"";

        $stmt = $this->db->stmt_init();
        if(!$stmt->prepare($this->sql2))
        {
            echo "<p class = 'text-danger' style = 'font-family: serif; margin-
left: 10px; margin-top: 5px;'> SQL ERROR: ".$this->db->error."
</p>";
        }
        else
        {
            $stmt->execute();

            $_SESSION['lastname'] = $newLname; //update the session
variable to have it displayed right away

```

```

        echo "<p class = 'text-success' style = 'font-family: serif; margin-
left: 10px; margin-top: 5px;'> Lastname has been updated! </p>";
        echo "<script> setTimeout(function(){
location.assign(location.href); }, 1000) </script>"; //refresh the page in 1
second to reveal the changed session variables
    }
}
}
/* This method updates the username of the user */
public function updateUname($userid, $newUname)
{
    if(empty($newUname))
    {
        echo "<p class = 'text-danger' style = 'font-family: serif; margin-
left: 10px; margin-top: 5px;'> Enter a new username, else don't submit
blank info! </p>";
    }
    else
    {
        $this -> sql2 = "UPDATE accounts SET
user_username='$newUname' WHERE user_id='".$userid."'";

        $stmt = $this -> db -> stmt_init();
        if(!$stmt -> prepare($this -> sql2))
        {
            echo "<p class = 'text-danger' style = 'font-family: serif; margin-
left: 10px; margin-top: 5px;'> SQL ERROR: ".$this -> db -> error."
</p>";
        }
        else
        {
            $stmt -> execute();

            $_SESSION['username'] = $newUname; //update the session
variable to have it displayed right away
            echo "<p class = 'text-success' style = 'font-family: serif; margin-
left: 10px; margin-top: 5px;'> Username has been updated! </p>";
            echo "<script> setTimeout(function(){
location.assign(location.href); }, 1000) </script>"; //refresh the page in 1
second to reveal the changed session variables
        }
    }
}
/* This method updates the password of the user */
public function updatePwd($userid, $newPwd)
{

```

```

        if(empty($newPwd))
        {
            echo "<p class = 'text-danger' style = 'font-family: serif; margin-
left: 10px; margin-top: 5px;'> Enter a new password, else don't submit
blank info! </p>";
        }
        else
        {
            //hash the password before adding it to database
            $newHashedPwd = password_hash($newPwd,
PASSWORD_DEFAULT);

            $this -> sql2 = "UPDATE accounts SET
user_password='$newHashedPwd' WHERE user_id='.$userid.'";

            $stmt = $this -> db -> stmt_init();
            if(!$stmt -> prepare($this -> sql2))
            {
                echo "<p class = 'text-danger' style = 'font-family: serif; margin-
left: 10px; margin-top: 5px;'> SQL ERROR: ".$this -> db -> error."
</p>";
            }
            else
            {
                $stmt -> execute();

                echo "<p class = 'text-success' style = 'font-family: serif; margin-
left: 10px; margin-top: 5px;'> Password has been updated! </p>";
                echo "<script> setTimeout(function(){
location.assign(location.href); }, 1000) </script>"; //refresh the page in 1
second to reveal the changed session variables
            }
        }
    }
    /* This method updates the email of the user */
    public function updateEmail($userid, $newEmail)
    {
        if(empty($newEmail))
        {
            echo "<p class = 'text-danger' style = 'font-family: serif; margin-
left: 10px; margin-top: 5px;'> Enter a new email, else don't submit blank
info! </p>";
        }
        else
        {

```

```

    $this -> sql2 = "UPDATE accounts SET user_email='$newEmail'
WHERE user_id=".$userid."";

    $stmt = $this -> db -> stmt_init();
    if(!$stmt -> prepare($this -> sql2))
    {
        echo "<p class = 'text-danger' style = 'font-family: serif; margin-
left: 10px; margin-top: 5px;'> SQL ERROR: ".$this -> db -> error."
</p>";
    }
    else
    {
        $stmt -> execute();

        $_SESSION['useremail'] = $newEmail; //update the session
variable to have it displayed right away
        echo "<p class = 'text-success' style = 'font-family: serif; margin-
left: 10px; margin-top: 5px;'> Email has been updated! </p>";
        echo "<script> setTimeout(function(){
location.assign(location.href); }, 1000) </script>"; //refresh the page in 1
second to reveal the changed session variables
    }
}
}
/* This method updates the status of the user */
public function updateStatus($userid, $newStatus)
{
    if(empty($newStatus))
    {
        echo "<p class = 'text-danger' style = 'font-family: serif; margin-
left: 10px; margin-top: 5px;'> Set new status, else don't submit blank info!
</p>";
    }
    else
    {
        $this -> sql2 = "UPDATE accounts SET user_status='$newStatus'
WHERE user_id=".$userid."";

        $stmt = $this -> db -> stmt_init();
        if(!$stmt -> prepare($this -> sql2))
        {
            echo "<p class = 'text-danger' style = 'font-family: serif; margin-
left: 10px; margin-top: 5px;'> SQL ERROR: ".$this -> db -> error."
</p>";
        }
        else
    }
}

```

```

        {
            $stmt -> execute();

            $_SESSION['status'] = $newStatus; //update the session variable
            to have it displayed right away
            echo "<p class = 'text-success' style = 'font-family: serif; margin-
            left: 10px; margin-top: 5px;'> Status has been updated! </p>";
            echo "<script> setTimeout(function(){
            location.assign(location.href); }, 1000) </script>"; //refresh the page in 1
            second to reveal the changed session variables
        }
    }
}

/* This method will add private messages to the MessageInbox table */
public function PrivateMessage($id, $messageTitle, $messageContent,
$messageTo, $messageFrom, $dateCreated)
{
    if(empty($messageTitle) || empty($messageContent) ||
    empty($messageTo))
    {
        echo "<p style = 'color: red; font-family: serif; text-align: center;'> Do
        not leave anything blank! </p>";
    }
    else
    {
        //check if the username the user sending the message has entered exists
        in the accounts table
        $this -> sql3 = "SELECT * FROM accounts WHERE
        user_username='$messageTo'";
        $result = $this -> db -> query($this -> sql3);

        if($result -> num_rows == 0)
        {
            echo "<p style = 'color: red; font-family: serif; text-align: center;'>
            There is no such member called \"\".$messageTo.\"\". Make sure to enter
            member username as exactly how it looks. </p>";
        }
        else //else if username does exist, obtain the user_id of that member of
        who the message will be sent to
        {
            $row1 = $result -> fetch_assoc();
            $userid = $row1['user_id']; //obtain the id of the user who the message
            will be sent to

```

```

        $this -> sql4 = "INSERT INTO messageinbox(user_id, user_id2,
messageTitle, messageContent, messageFrom, messageTo, date_sent)
        VALUES(?, ?, ?, ?, ?, ?, ?)";

        $stmt = $this -> db -> stmt_init();

        if(!$stmt -> prepare($this -> sql4))
        {
            echo "<script> SQL ERROR: alert('".$stmt -> error()."'); </script>";
        }
        else
        {
            $stmt -> bind_param("ssssss", $id, $userid, $messageTitle,
$messageContent, $messageFrom, $messageTo, $dateCreated);
            $stmt -> execute();

            echo "<p style = 'color: green; font-family: serif; text-align: center;'>
Message has been sent! </p>";
        }
    }
}

/* This method displays the messages the current user logged in has sent to
other members */
public function DisplayMessages1($id)
{
    $this -> sql5 = "SELECT * FROM messageinbox WHERE
user_id='$id'";

    $stmt = $this -> db -> stmt_init();

    if(!$stmt -> prepare($this -> sql5))
    {
        echo "<script> alert('".$stmt -> error()."'); </script>";
    }
    else
    {
        $stmt -> execute();
        $result = $stmt -> get_result();

        if($result -> num_rows == 0)
        {
            echo "<script>
                var messagewindow =
document.getElementById('messagessend');

```

```

        messagewindow.style.display = 'none';

        var Notice = document.createElement('p');
        var textnode = document.createTextNode('You have not sent
any private messages');

        var notice = Notice.appendChild(textnode);

document.getElementById('sentmessages').appendChild(notice);
    </script>";
    }
    else
    {
        while($row = $result -> fetch_assoc())
        {
            echo "<tr>
                <td>".$row['messageTo']. "</td>
                <td>".$row['messageTitle']. "</td>
                <td>".$row['date_sent']. "</td>
                <td>".$row['messageContent']. "</td>
            </tr>";
        }
    }
}

/* This method displays the messages that other members sent to the
current user logged in */
public function DisplayMessage2($id)
{
    $this -> sql6 = "SELECT * FROM messageinbox WHERE
user_id2='$id'";

    $stmt = $this -> db -> stmt_init();

    if(!$stmt -> prepare($this -> sql6))
    {
        echo "<script> alert('".$stmt -> error()."'); </script>";
    }
    else
    {
        $stmt -> execute();
        $result = $stmt -> get_result();

        if($result -> num_rows == 0)

```



```

        {
            echo "<script>
                var messagewindow =
document.getElementById('usermessageinbox');
                messagewindow.style.display = 'none';

                var Notice = document.createElement('p');
                var textnode = document.createTextNode('No one has sent you
any private messages');

                var notice = Notice.appendChild(textnode);

document.getElementById('messageinbox').appendChild(notice);
            </script>";
        }
        else
        {
            while($row = $result -> fetch_assoc())
            {
                echo "<tr>
                    <td>".$row['messageFrom']. "</td>
                    <td>".$row['messageTitle']. "</td>
                    <td>".$row['date_sent']. "</td>
                    <td>".$row['messageContent']. "</td>
                </tr>";
            }
        }
    }
}
?>

```

### userprofile.php

```

<?php
    session_start(); //needed for starting session

    //for updating user information
    include 'includes/ProfileHandler.php';
    $update = new ProfileHandler();

    $userid = $_SESSION['userid'];
?>
<html>
    <head lang = "en">
        <title> Welcome to your profile </title>
        <meta charset = "utf-8"/>

```

```

        <meta name = "viewport" content = "width=device-width, initial-
scale=1"/>

        <link rel = "stylesheet" href = "css/userprofile.css"/>

        <!-- Link the Bootstrap 4 CDN CSS files (this is mainly for the navbar) --
>
        <link rel='stylesheet'
href='https://maxcdn.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css'>

        <!-- Link the Bootstrap 4 CDN JS and jQuery files (must be in this order:
jQuery, Popper, Bootstrap.js) -->
        <script
src='https://ajax.googleapis.com/ajax/libs/jquery/3.4.1/jquery.min.js'></script>
        <script
src='https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd/popper.min.js'
></script>
        <script
src='https://maxcdn.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js'></scr
ipt>


        <!-- For changing the background color of navbar elements when
hovering cursor on them -->
        <style>
            .nav-item:hover
            {
                background-color: blue;
            }
        </style>
    </head>

    <body>
        <!-- Navbar -->
        <nav class = "navbar navbar-expand-sm bg-dark navbar-dark fixed-top"
style = 'border-radius: 10px; font-family: serif;'>
            <ul class = "navbar-nav mr-auto">
                <li class = "nav-item">
                    <a href = "forumhome.php" class = "nav-link text-white">
Home </a>
                </li>
                <li class = "nav-item">
                    <a href = "members.php" class = "nav-link text-white">
Members </a>
                </li>

```

```

        <li class = "nav-item">
            <a href = "activities.php" class = "nav-link text-white"> Activity
        </a>
        </li>
    </ul>
    <ul class = "navbar-nav ml-auto">
        <li class = "nav-item dropdown">
            <a href = "#" class = "nav-link dropdown-toggle text-white"
data-toggle = "dropdown"> Hello: <?php echo $_SESSION['username']; ?>
        </a>
            <div class = "dropdown-menu" style = "padding: 10px;">
                <span class = "dropdown-link d-flex justify-content-center"
style = "padding: 10px;"> <?php echo "<img src = '' . $_SESSION['pic'] . ''
height='70' width='70'/>"; ?> </span>
                <span class = "dropdown-link d-flex justify-content-center">
<?php echo $_SESSION['firstname'] . " " . $_SESSION['lastname']; ?> </span>
                <span class = "dropdown-link d-flex justify-content-center">
<?php echo $_SESSION['useremail']; ?> </span>
                <span class = "dropdown-link d-flex justify-content-center">
<?php echo $_SESSION['status']; ?> </span>
                <span class = "dropdown-link d-flex justify-content-center">
<?php echo $_SESSION['regdate']; ?> </span>
                <a href = "#" class = "dropdown-link btn btn-primary d-flex
justify-content-center"> Go to account settings </a>
            </div>
        </li>
        <li class = "nav-item">
            <a href = "includes/logout.php" class = "nav-link text-white">
Logout </a>
        </li>
    </ul>
</nav>

```

```

<div style = "margin-top: 7%;"></div>

<!-- Displays profile image and settings for changing picture -->
<div id = "profileimage-window" style = "margin: 5%; padding:
20px; border-radius: 5px; border-style: dotted; border-width: thin; font-
family: serif;">
    <img src = <?php echo $_SESSION['pic']; ?> height = '150' width
= '150' />
    <h3 style = "font-family: serif;"> Select new image to upload:
</h3>
    <form action = 'userprofile.php' method = 'POST' enctype =
'multipart/form-data' style = "margin-top: 5%;" class = "form-inline">
        <div class = "form-control">

```

```

        <input type = 'file' name = 'getFile' class = 'form-control-
file' />
    </div>
    <input type = 'submit' value = 'Upload Photo' name = 'submit-
pic' class = 'btn btn-success form-control' />
</form>
<?php
    if(isset($_POST['submit-pic']))
    {
        $update -> uploadPics($userid);
    }
?>
</div>

<!-- Settings for changing user info -->
<div id = "settings-window" style = "margin: 5%; padding: 10px;
border-radius: 5px; border-width: thin; border-style: dotted;">
    <h3 style = "text-align: center; font-family: serif;"> Change Info
</h3>
    <div class = "container d-flex justify-content-center">
        <form action = 'userprofile.php' method = "POST" class =
"form-inline" style = "font-family: serif;">
            <div class = "form-group">
                <input type = "text" name = "changefname" placeholder
= "Set new firstname" class = "form-control mr-sm-2" />
                <input type = "submit" name = "new-fname" value =
"Change firstname" class = "btn btn-success text-white" />
            </div>
        </form>
        <?php
            if(isset($_POST['new-fname']))
            {
                $newfname = $_POST['changefname'];
                $update -> updateFname($userid, $newfname);
            }
        ?>
    </div>
    <div class = "container d-flex justify-content-center">
        <form action = 'userprofile.php' method = "POST" class =
"form-inline" style = "font-family: serif;">
            <div class = "form-group">
                <input type = "text" name = "changelname" placeholder
= "Set new lastname" class = "form-control mr-sm-2" />
                <input type = "submit" name = "new-lname" value =
"Change lastname" class = "btn btn-success text-white" />
            </div>

```

```

</form>
<?php
    if(isset($_POST['new-lname']))
    {
        $newlname = $_POST['changelname'];
        $update -> updateLname($userid, $newlname);
    }
?>
</div>
<div class = "container d-flex justify-content-center">
    <form action = 'userprofile.php' method = "POST" class =
"form-inline" style = "font-family: serif;">
        <div class = "form-group">
            <input type = "text" name = "changeuname"
placeholder = "Set new username" class = "form-control mr-sm-2"/>
            <input type = "submit" name = "new-uname" value =
"Change username" class = "btn btn-success text-white"/>
        </div>
    </form>
    <?php
        if(isset($_POST['new-uname']))
        {
            $newuname = $_POST['changeuname'];
            $update -> updateUname($userid, $newuname);
        }
    ?>
</div>
<div class = "container d-flex justify-content-center">
    <form action = 'userprofile.php' method = "POST" class =
"form-inline" style = "font-family: serif;">
        <div class = "form-group">
            <input type = "password" name = "changepwd"
placeholder = "Set new password" class = "form-control mr-sm-2"/>
            <input type = "submit" name = "new-pwd" value =
"Change password" class = "btn btn-success text-white"/>
        </div>
    </form>
    <?php
        if(isset($_POST['new-pwd']))
        {
            $newpwd = $_POST['changepwd'];
            $update -> updatePwd($userid, $newpwd);
        }
    ?>
</div>
<div class = "container d-flex justify-content-center">

```

```

        <form action = 'userprofile.php' method = "POST" class =
"form-inline" style = "font-family: serif;">
            <div class = "form-group">
                <input type = "email" name = "changeemail"
placeholder = "Set new email" class = "form-control mr-sm-2"/>
                <input type = "submit" name = "new-email" value =
"Change email" class = "btn btn-success text-white"/>
            </div>
        </form>
        <?php
        if(isset($_POST['new-email']))
        {
            $newemail = $_POST['changeemail'];
            $update -> updateEmail($userid, $newemail);
        }
        ?>
    </div>
    <div class = "container d-flex justify-content-center">
        <form action = 'userprofile.php' method = "POST" class =
"form-inline" style = "font-family: serif;">
            <div class = "form-group">
                <input type = "text" name = "changestatus" placeholder
= "Set new status" class = "form-control mr-sm-2"/>
                <input type = "submit" name = "new-status" value =
"Change status" class = "btn btn-success text-white"/>
            </div>
        </form>
        <?php
        if(isset($_POST['new-status']))
        {
            $newstatus = $_POST['changestatus'];
            $update -> updateStatus($userid, $newstatus);
        }
        ?>
    </div>
</div>

<!-- Displays messages from other users and messages sent -->
<div id = "message-window" style = "margin: 5%; padding: 10px;
border-radius: 5px; border-style: dotted; border-width: thin; font-family:
serif;">
    <!-- Messages from other users -->
    <div id = "messageinbox">
        <h4 text-align: center;> Your private message inbox (messages
other forum members have sent to you): </h4>
        <table id = "usermessageinbox" style = "margin-top: 5%;">

```

```

        <tr>
            <th> From </th>
            <th> Title </th>
            <th> Date sent </th>
            <th> Message </th>
        </tr>
        <?php
            $update -> DisplayMessage2($_SESSION['userid']);
        ?>
    </table>
</div>

<!-- Messages you sent -->
<div id = "sentmessages" style = "margin-top: 10%;">
    <h4> Private messages you have sent to other forum members:
</h4>

    <table id = "messagessend" style = "margin-top: 10px;">
        <tr>
            <th> To </th>
            <th> Title </th>
            <th> Date sent </th>
            <th> Message </th>
        </tr>
        <?php
            $update -> DisplayMessages1($_SESSION['userid']);
        ?>
    </table>
</div>
</div>

<!-- Option for sending private message -->
<div id = "privatemessage-window" style = "margin: 5%; padding: 20px;
border-radius: 5px; border-style: dotted; border-width: thin; font-family:
serif;">
    <h3 style = "text-align: center;"> Send Private Message </h3>

    <form method = "POST" action = "userprofile.php" id = "sendpm">
        <div class = "form-group d-flex justify-content-center">
            <label> Enter username of forum member: <input type = "text"
name = "memberName" class = "form-control"/> </label>
        </div>
        <div class = "form-group d-flex justify-content-center">
            <label style = "margin-top: 10px;"> Enter title of private
message: <input type = "text" name = "messageTitle" class = "form-control"/>
</label>
        </div>
    </form>
</div>

```

```

        <div class = "form-group">
            <p style = "text-align: center;"> Enter Message: </p>
            <textarea name = "pm-message" rows = "10" style = "resize:
none;" class = "form-control"></textarea>
        </div>
        <div class = "form-group d-flex justify-content-center">
            <input type = "submit" name = "submitPM" value = "Send
PM" style = "margin-top: 20px;" class = "btn btn-success"/>
        </div>
    </form>
    <?php
        if(isset($_POST['submitPM']))
        {
            $datecreated = date("m/d/Y"); //when the message was created
            $username = $_SESSION['username']; //username of the member
            sending the message
            $membername = $_POST['memberName']; //username of the
            member who the message is being sent to
            $messagetitle = $_POST['messageTitle']; //title of the private
            message
            $messagecontent = $_POST['pm-message']; //content of the
            private message

            $update -> PrivateMessage($_SESSION['userid'], $messagetitle,
            $messagecontent, $membername, $username, $datecreated);
        }
    ?>
</div>

<!-- List of threads you last made -->
<div id = "user-posts" style = "margin: 5%; padding: 10px; border-
radius: 5px; border-style: dotted; border-width: thin; font-family: serif;">
    <h3 style = "margin-bottom: 15px;"> Some threads you have made
</h3>
    <table id = "threadsCreated">
        <tr>
            <th> Threads you have created: </th>
            <th> Subject of your threads: </th>
            <th> Topic of your threads </th>
            <th> Date of threads posted: </th>
        </tr>
    <?php
        include 'includes/ThreadHandler.php';

        $showThreads = new ThreadHandler();
        $showThreads -> showThreads($_SESSION['userid']);

```



```

        ?>
    </table>
</div>
</body>
</html>

```

- For the Members and Activities tabs, you want to change a few things. The methods that handle the functionality for those tabs, you want to move those methods from the Accounts class of the “AccountHandler.php” file to a new class called “UserActivity” and add that class to a new includes file called “UserActivity.php”. This is due to a few problems with the mysqli connection. In the UserActivity.php file of the includes folder, type the following code:

#### **UserActivity.php**

```

<?php
    include_once "database.php";

    class UserActivity
    {
        private $dbConn;

        //For the members.php page
        private $sql1; //For the RegisteredUsers() method. Used to obtain and
        display the users who just registered for a forum account
        private $sql2; //For the setLoggedUsers() method. Used to set the users
        who are currently logged on to the forum
        private $sql3; //For the setLoggedUsers() method. Used to set the users
        who are not logged on to the forum
        private $sql4; //For the displayLoggedUsers() method. Used to obtain and
        display the users who are currently logged on to the forum

        //For the activites.php page and for the RecentActivity() method
        private $sql5; //For looping through the accounts table. To see who just
        created an account
        private $sql6; //For looping through the messageinbox table. To see who
        just sent a message
        private $sql7; //For looping through the replies table. To see who just
        replied to a thread
        private $sql8; //For looping through the threads table. To see who just
        created a thread

        /*****
        *****/

        public function __construct()
        {

```

```

        $this->dbConn = new Connection();
        $this->dbConn = $this->dbConn->connect();
    }

    /**
     *
     */

    /* The following methods are for the members.php page */
    /* This method checks all the users and the dates they first registered */
    public function RegisteredUsers()
    {
        $this->sql1 = "SELECT * FROM accounts;";

        $stmt = $this->dbConn->stmt_init();

        if(!$stmt->prepare($this->sql1))
        {
            echo "<script> alert('SQL ERROR: " . $stmt->error . "'); </script>";
        }
        else
        {
            $stmt->execute();
            $result = $stmt->get_result();

            if($result->num_rows > 0)
            {
                while($row = $result->fetch_assoc())
                {
                    echo "<tbody>
                        <td>".$row['user_username']."</td>
                        <td>".$row['reg_date']."</td>
                    </tbody>";
                }
            }
            else
            {
                echo "<script>
document.getElementById('registeredusersError').innerHTML = 'There are no
new users at this time'; </script>";
            }
        }
    }
}

```

```

/*****
*****
*****/

/* Checks users who are currently logged into the forum. If logged in, set
the time they logged in. Else, output 'Offline' */
public function setLoggedUsers($id, $username)
{
    $loggeddate = date("m/d/Y h:i A"); //when the user first logged in

    if($id == $_SESSION['userid'] && $username ==
$_SESSION['username'])
    {
        //indicates that the user is logged on
        $this -> sql2 = "UPDATE activeaccounts SET user_id='$id',
log_status='Active', login_time='$loggeddate' WHERE
user_username='$username'";
        $stmt1 = $this -> dbConn -> stmt_init();
        if(!$stmt1 -> prepare($this -> sql2))
        {
            echo "<script> alert('".$stmt1 -> error()."'); </script>";
        }
        else
        {
            $stmt1 -> execute();
        }

        //indicates that the user is logged off
        $this -> sql3 = "UPDATE activeaccounts SET log_status='Offline',
login_time='Not Logged On' WHERE user_username!='$username'";
        $stmt2 = $this -> dbConn -> stmt_init();
        if(!$stmt2 -> prepare($this -> sql3))
        {
            echo "<script> alert('".$stmt2 -> error()."'); </script>";
        }
        else
        {
            $stmt2 -> execute();
        }
    }
}

/* Display all the number of users who are logged in and those who are not
*/
public function displayLoggedUsers()
{
    $this -> sql4 = "SELECT * FROM activeaccounts";

```

```

$stmt3 = $this -> dbConn -> stmt_init();
$stmt3 -> prepare($this -> sql4);
$stmt3 -> execute();

$result = $stmt3 -> get_result();
if($result -> num_rows > 0)
{
    while($row = $result -> fetch_assoc())
    {
        echo "<tbody>
            <td>".$row['user_username']."</td>
            <td>".$row['log_status']."</td>
            <td>".$row['login_time']."</td>
        </tbody>";
    }
}
else
{
    echo "<script>
document.getElementById('loggedusersError').innerHTML = 'There are no
users currently logged on'; </script>";
}
}

/*****
*****
*****/

/* This method is for the activities.php page */
/* Displays all the recent activities of registered forum members */
public function RecentActivity()
{
    //All SQL queries will only obtain the very last row of each table

    /*This query checks who just created an account*/
    $this -> sql5 = "SELECT * FROM accounts ORDER BY user_id DESC
LIMIT 1;";
    $stmt4 = $this -> dbConn -> stmt_init();
    if(!$stmt4 -> prepare($this -> sql5))
    {
        echo "<script> alert('SQL ERROR: " . $stmt4 -> error . "'); </script>";
    }
    else
    {
        $stmt4 -> execute();
    }
}

```

```

$result1 = $stmt4 -> get_result();

while($row1 = $result1 -> fetch_assoc())
{
    echo "<br/>".$row1['user_username']." just registered a forum
account!<br/>";
}
}

/*This query checks who just sent a message to another user*/
$this -> sql6 = "SELECT * FROM messageinbox ORDER BY user_id
DESC LIMIT 1;";
$stmt5 = $this -> dbConn -> stmt_init();
if(!$stmt5 -> prepare($this -> sql6))
{
    echo "<script> alert('SQL ERROR: " . $stmt5 -> error . "'); </script>";
}
else
{
    $stmt5 -> execute();
    $result2 = $stmt5 -> get_result();

    while($row2 = $result2 -> fetch_assoc())
    {
        echo "<br/>".$row2['messageFrom']." just sent a message to
".$row2['messageTo']. """;
    }
}

/*This query checks who replied to a certain thread*/
$this -> sql7 = "SELECT * FROM replies ORDER BY user_id DESC
LIMIT 1;";
$stmt6 = $this -> dbConn -> stmt_init();
if(!$stmt6 -> prepare($this -> sql7))
{
    echo "<script> alert('SQL ERROR: " . $stmt6 -> error . "'); </script>";
}
else
{
    $stmt6 -> execute();
    $result3 = $stmt6 -> get_result();

    while($row3 = $result3 -> fetch_assoc())
    {
        echo "<br/> ".$row3['user_username']." replied to the thread:
\\\"".$row3['thread_title']. "\\\" <br/>";
    }
}

```

```

    }
}

/*This query checks who just created a thread*/
$this->sql8 = "SELECT * FROM threads ORDER BY user_id DESC
LIMIT 1;";
$stmt7 = $this->dbConn->stmt_init();
if(!$stmt7->prepare($this->sql8))
{
    echo "<script> alert('SQL ERROR: " . $stmt7->error . "'); </script>";
}
else
{
    $stmt7->execute();
    $result4 = $stmt7->get_result();

    while($row4 = $result4->fetch_assoc())
    {
        echo "<br/>".$row4['thread_author']." just created a new
        ".$row4['thread_subject']." thread called \"\"".$row4['thread_title']."\"!<br/>";
    }
}

}

/*****
*****
*****/

//closes the database connection at the end of each class call
public function __destruct()
{
    $this->dbConn->close();
}
}

?>

```

Now in the members.php and activities.php pages, add the following:

**members.php**

```

<?php
    session_start();

    include "includes/UserActivity.php";
    $users = new UserActivity();
?>
<html lang = "en">
    <head>

```

```

<title> Check to see who's logged in and who just signed up to this
website </title>

<meta charset = "utf-8"/>
<meta name = "viewport" content = "width=device-width, initial-
scale=1"/>

<link href="https://fonts.googleapis.com/css?family=Lobster"
rel="stylesheet">
<link rel = "stylesheet" href = "css/main.css"/>

<!-- Link the Bootstrap 4 CDN CSS files (this is mainly for the navbar) --
>
<link rel='stylesheet'
href='https://maxcdn.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css'>
<!-- Link the Bootstrap 4 CDN JS and jQuery files (must be in this order:
jQuery, Popper, Bootstrap.js) -->
<script
src='https://ajax.googleapis.com/ajax/libs/jquery/3.4.1/jquery.min.js'></script>
<script
src='https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd/popper.min.js'
></script>
<script
src='https://maxcdn.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js'></scr
ipt>

<!-- For changing the background color of navbar elements when
hovering cursor on them -->
<style>
.nav-item:hover
{
background-color: blue;
}
</style>
</head>

<body>
<!-- Navbar -->
<nav class = "navbar navbar-expand-sm bg-dark navbar-dark fixed-top"
style = "border-radius: 10px; font-family: serif;">
<ul class = "navbar-nav mr-auto">
<li class = "nav-item">
<a href = "forumhome.php" class = "nav-link text-white"> Home
</a>
</li>

```

```

        <li class = "nav-item">
            <a href = "#" class = "nav-link text-white"> Members </a>
        </li>
        <li class = "nav-item">
            <a href = "activities.php" class = "nav-link text-white"> Activity
    </a>
    </li>
</ul>
<ul class = "navbar-nav ml-auto">
    <li class = "nav-item dropdown">
        <a href = "#" class = "nav-link dropdown-toggle text-white"
data-toggle = "dropdown"> Hello: <?php echo $_SESSION['username']; ?>
    </a>
        <div class = "dropdown-menu" style = "padding: 10px;">
            <span class = "dropdown-link d-flex justify-content-center"
style = "padding: 10px;"> <?php echo "<img src = '' . $_SESSION['pic'] . ''
height='70' width='70' />"; ?> </span>
            <span class = "dropdown-link d-flex justify-content-center">
<?php echo $_SESSION['firstname'] . " " . $_SESSION['lastname']; ?> </span>
            <span class = "dropdown-link d-flex justify-content-center">
<?php echo $_SESSION['useremail']; ?> </span>
            <span class = "dropdown-link d-flex justify-content-center">
<?php echo $_SESSION['status']; ?> </span>
            <span class = "dropdown-link d-flex justify-content-center">
<?php echo $_SESSION['regdate']; ?> </span>
            <a href = "userprofile.php" class = "dropdown-link btn btn-
primary d-flex justify-content-center"> Go to account settings </a>
        </div>
    </li>
    <li class = "nav-item">
        <a href = "includes/logout.php" class = "nav-link text-white">
Logout </a>
    </li>
</ul>
</nav>

```

```

<!-- Members window (displays both the number of users who just
signed up an account and the number of users who are currently logged in) -->
<div id = "memberswindow">
    <!-- This table displays the number of user who just registered for a
forum account -->
    <table id = "registeredUsers" class = "table">
        <h4 style = "color: white; font-family: 'Lobster', cursive; font-
size: 25px; text-align: center;"> Forum members </h4>
        <thead>
            <th> Username </th>

```



```

        <th> Date of registration </th>
    </thead>
    <?php
        $users -> RegisteredUsers();
    ?>
</table>
    <p id = "registeredusersError" class = "text-danger" style =
"font-family: serif; text-align: center;"></p>

    <div style = "margin-top: 15%;"></div>

    <!-- This table displays the number of users currently logged on to the
forum -->
    <table id = "loggedUsers" class = "table">
        <h4 style = "color: white; font-family: 'Lobster', cursive; font-
size: 25px; text-align: center;"> See who's currently logged in </h4>
        <thead>
            <th> Username </th>
            <th> Status </th>
            <th> Logged in at </th>
        </thead>
        <?php
            $users -> displayLoggedUsers();
        ?>
    </table>
    <p id = "loggedusersError" class = "text-danger" style = "font-
family: serif; text-align: center;"></p>
</div>
</body>
</html>

```

### activities.php

```

<?php
    session_start();

    include 'includes/AccountHandler.php';
    $recentactivity = new Accounts();
?>
<html lang = "en">
    <head>
        <title> See recent activity of forum members! </title>

        <meta charset = "utf-8"/>
        <meta name = "viewport" content = "width=device-width, initial-
scale=1"/>

        <link rel = "stylesheet" href = "css/main.css"/>
    </head>
    <body>

```

```

        <!-- Link the Bootstrap 4 CDN CSS files (this is mainly for the navbar) -
->
        <link rel='stylesheet'
href='https://maxcdn.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css'>
        <!-- Link the Bootstrap 4 CDN JS and jQuery files (must be in this
order: jQuery, Popper, Bootstrap.js) -->
        <script
src='https://ajax.googleapis.com/ajax/libs/jquery/3.4.1/jquery.min.js'></script>
        <script
src='https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd/popper.min.js'
></script>
        <script
src='https://maxcdn.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js'></scr
ipt>

        <!-- For changing the background color of navbar elements when
hovering cursor on them -->
        <style>
            .nav-item:hover
            {
                background-color: blue;
            }
        </style>
    </head>

    <body>
        <!-- Navbar -->
        <nav class = "navbar navbar-expand-sm bg-dark navbar-dark fixed-top"
style = 'border-radius: 10px; font-family: serif;'>
            <ul class = "navbar-nav mr-auto">
                <li class = "nav-item">
                    <a href = "forumhome.php" class = "nav-link text-white"> Home
                </a>
                </li>
                <li class = "nav-item">
                    <a href = "members.php" class = "nav-link text-white">
Members </a>
                </li>
                <li class = "nav-item">
                    <a href = "#" class = "nav-link text-white"> Activity </a>
                </li>
            </ul>
            <ul class = "navbar-nav ml-auto">
                <li class = "nav-item dropdown">

```

```

        <a href = "#" class = "nav-link dropdown-toggle text-white"
data-toggle = "dropdown"> Hello: <?php echo $_SESSION['username']; ?>
</a>

        <div class = "dropdown-menu" style = "padding: 10px;">
            <span class = "dropdown-link d-flex justify-content-center"
style = "padding: 10px;"> <?php echo "<img src = '" . $_SESSION['pic'] . '"
height='70' width='70'/>"; ?> </span>
            <span class = "dropdown-link d-flex justify-content-center">
<?php echo $_SESSION['firstname'] . " " . $_SESSION['lastname']; ?> </span>
            <span class = "dropdown-link d-flex justify-content-center">
<?php echo $_SESSION['useremail']; ?> </span>
            <span class = "dropdown-link d-flex justify-content-center">
<?php echo $_SESSION['status']; ?> </span>
            <span class = "dropdown-link d-flex justify-content-center">
<?php echo $_SESSION['regdate']; ?> </span>
            <a href = "userprofile.php" class = "dropdown-link btn btn-
primary d-flex justify-content-center"> Go to account settings </a>
        </div>
    </li>
    <li class = "nav-item">
        <a href = "includes/logout.php" class = "nav-link text-white">
Logout </a>
    </li>
</ul>
</nav>

<div style = "margin-top: 7%;"></div>

<!-- Forum activity table -->
<div id = "forumactivity">
    <h3>Recent Activity</h3> <p>Activity stream for all registered
members</p> </th>
    <div style = "border-bottom: 1px solid white;"></div>
    <?php
        include "includes/UserActivity.php";
        $recentactivity = new UserActivity();
        $recentactivity -> RecentActivity();
    ?>
</div>
</body>
</html>

```

Now in the forumhome.php page, add the following to the top of the page

**forumhome.php**

```
<?php
    session_start();

    include "includes/ThreadHandler.php";

    /* For the Culture threads */
    $philosophyThreads = new ThreadHandler(); //for displaying the most recent
philosophy threads
    $religionThreads = new ThreadHandler(); //for displaying the most recent
religion threads
    $artThreads = new ThreadHandler(); //for displaying the most recent art
threads

    /* For the Natural Science and Mathematics threads */
    $astronomyThreads = new ThreadHandler(); //for displaying the most recent
astronomy threads
    $biologyThreads = new ThreadHandler(); //for displaying the most recent
biology threads
    $chemistryThreads = new ThreadHandler(); //for displaying the most recent
chemistry threads
    $earthsciThreads = new ThreadHandler(); //for displaying the most recent
earth science threads
    $mathThreads = new ThreadHandler(); //for displaying the most recent
mathematics threads
    $physicsThreads = new ThreadHandler(); //for displaying the most recent
physics threads

    /* For the Social Science threads */
    $anthroThreads = new ThreadHandler(); //for displaying the most recent
anthropology threads
    $archThreads = new ThreadHandler(); //for displaying the most recent
archeology threads
    $lingThreads = new ThreadHandler(); //for displaying the most recent
linguistics threads
    $psyThreads = new ThreadHandler(); //for displaying the most recent
psychology threads
    $socThreads = new ThreadHandler(); //for displaying the most recent
sociology threads

    /* For the Technology threads */
    $comphardThreads = new ThreadHandler(); //for displaying the most recent
computer hardware threads
    $compssoftThreads = new ThreadHandler(); //for displaying the most recent
computer software threads
```

```
$phoneThreads = new ThreadHandler(); //for displaying the most recent
smartphones threads
$vidgmThreads = new ThreadHandler(); //for displaying the most recent
video game threads
```

```
/* For the International News threads */
$intnewsThreads = new ThreadHandler(); //for displaying the most recent
international news threads
```

**//if user is logged in, set their log\_status to active in the activeaccounts table**

```
include "includes/UserActivity.php";
$useractivity = new UserActivity();
```

```
$userid = $_SESSION['userid'];
$username = $_SESSION['username'];
$useractivity -> setLoggedUsers($userid, $username);
```

```
?>
```

```
<html lang = "en">
```

```
  <head>
```

```
    <title> Welcome to forum </title>
```

```
    <meta charset = "utf-8"/>
```

```
    <meta name = "viewport" content = "width=device-width, initial-
scale=1"/>
```

```
    <link href="https://fonts.googleapis.com/css?family=Lobster"
rel="stylesheet">
```

```
    <link rel = "stylesheet" href = "css/main.css"/>
```

```
    <!-- Link the Bootstrap 4 CDN CSS files (this is mainly for the navbar) --
>
```

```
    <link rel='stylesheet'
href='https://maxcdn.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css'>
```

```
    <!-- For changing the background color of navbar elements when
hovering cursor on them -->
```

```
    <style>
```

```
      .nav-item:hover
```

```
      {
```

```
        background-color: blue;
```

```
      }
```

```
    </style>
```

```
  </head>
```

```
  <body>
```

```
    <!-- Navbar -->
```

```

<nav class = "navbar navbar-expand-sm bg-dark navbar-dark fixed-top"
style = "border-radius: 10px; font-family: serif;">
    <ul class = "navbar-nav mr-auto">
        <li class = "nav-item">
            <a href = "#" class = "nav-link text-white"> Home </a>
        </li>
        <li class = "nav-item">
            <a href = "members.php" class = "nav-link text-white">
Members </a>
        </li>
        <li class = "nav-item">
            <a href = "activities.php" class = "nav-link text-white"> Activity
</a>
        </li>
    </ul>
    <ul class = "navbar-nav ml-auto">
        <li class = "nav-item dropdown">
            <a href = "#" class = "nav-link dropdown-toggle text-white"
data-toggle = "dropdown"> Hello: <?php echo $_SESSION['username']; ?>
</a>
            <div class = "dropdown-menu" style = "padding: 10px;">
                <span class = "dropdown-link d-flex justify-content-center"
style = "padding: 10px;"> <?php echo "<img src = '' . $_SESSION['pic'] . ''
height='70' width='70'/>"; ?> </span>
                <span class = "dropdown-link d-flex justify-content-center">
<?php echo $_SESSION['firstname'] . " " . $_SESSION['lastname']; ?> </span>
                <span class = "dropdown-link d-flex justify-content-center">
<?php echo $_SESSION['useremail']; ?> </span>
                <span class = "dropdown-link d-flex justify-content-center">
<?php echo $_SESSION['status']; ?> </span>
                <span class = "dropdown-link d-flex justify-content-center">
<?php echo $_SESSION['regdate']; ?> </span>
                <a href = "userprofile.php" class = "dropdown-link btn btn-
primary d-flex justify-content-center"> Go to account settings </a>
            </div>
        </li>
        <li class = "nav-item">
            <a href = "includes/logout.php" class = "nav-link text-white">
Logout </a>
        </li>
    </ul>
</nav>

<div style = "margin-top: 7%;"></div>

<!-- Search for forums -->

```

```

<div id = "searchwindow">
    <p> Search for your favorite topic thread </p>

    <form method = "POST" action = "forumhome.php">
        <input type = "text" name = "threadname" placeholder = "Enter
topic thread"/>
        <br/>
        <input type = "text" name = "subjectname" placeholder = "Enter
subject"/>
        <br/>
        <input type = "submit" name = "submit-search" value = "Search"/>
    </form>
    <?php
    if(isset($_POST['submit-search']))
    {
        $ThreadName = $_POST['threadname'];
        $SubjectName = $_POST['subjectname'];

        $searchThread = new ThreadHandler();
        $searchThread -> searchThreads($ThreadName, $SubjectName);
    }
    ?>
</div>

```

```

<!-- Forum window -->
<div id = "forumwindow">
    <!-- Culture -->
    <table class = "maintable" style = "margin-bottom: 10px;">
        <thead>
            <tr>
                <th class = "leftside"> Culture </th>
                <th class = "rightside"> Latest Posts </th>
            </tr>
        </thead>

        <tbody>
            <tr>
                <td class = "leftside"> <a href =
"Subjects/Culture/Philosophy.php">Philosophy</a> </td>
                <td class = "rightside"> <?php $philosophyThreads ->
obtainTopicThread("Culture", "Philosophy"); ?> </td>
            </tr>
            <tr>
                <td class = "leftside"> <a href =
"Subjects/Culture/Religion.php">Religion</a> </td>

```

```

        <td class = "rightside"> <?php $religionThreads ->
obtainTopicThread("Culture", "Religion"); ?> </td>
    </tr>
    <tr>
        <td class = "leftside"> <a href =
"Subjects/Culture/Art.php">Art</a> </td>
        <td class = "rightside"> <?php $artThreads ->
obtainTopicThread("Culture", "Art"); ?> </td>
    </tr>
</tbody>
</table>

```

```

<!-- Social Science -->
<table class = "maintable" style = "margin-bottom: 10px;">
    <thead>
        <tr>
            <th class = "leftside"> Social Science </th>
            <th class = "rightside"> Latest Posts </th>
        </tr>
    </thead>

    <tbody>
        <tr>
            <td class = "leftside"> <a href = "Subjects/Social
Science/Anthropology.php">Anthropology</a> </td>
            <td class = "rightside"> <?php $anthroThreads ->
obtainTopicThread("Social Science", "Anthropology"); ?> </td>
        </tr>
        <tr>
            <td class = "leftside"> <a href = "Subjects/Social
Science/Archeology.php">Archeology</a> </td>
            <td class = "rightside"> <?php $archThreads ->
obtainTopicThread("Social Science", "Archeology"); ?> </td>
        </tr>
        <tr>
            <td class = "leftside"> <a href = "Subjects/Social
Science/Sociology.php">Sociology</a></td>
            <td class = "rightside"> <?php $socThreads ->
obtainTopicThread("Social Science", "Sociology"); ?> </td>
        </tr>
        <tr>
            <td class = "leftside"> <a href = "Subjects/Social
Science/Psychology.php">Psychology</a></td>
            <td class = "rightside"> <?php $psyThreads ->
obtainTopicThread("Social Science", "Psychology"); ?> </td>
        </tr>
    </tbody>
</table>

```



```

</tr>
<td class = "leftside"> <a href = "Subjects/Social
Science/Linguistics.php">Linguistics</a></td>
<td class = "rightside"> <?php $lingThreads ->
obtainTopicThread("Social Science", "Linguistics"); ?> </td>
</tr>
</tbody>
</table>

```

```

<!-- Natural Science and Mathematics -->
<table class = "maintable" style = "margin-bottom: 10px;">
<thead>
<tr>
<th class = "leftside"> Natural Science and Mathematics </th>
<th class = "rightside"> Latest Posts </th>
</tr>
</thead>

<tbody>
<tr>
<td class = "leftside"> <a href = "Subjects/Natural Science and
Mathematics/Biology.php">Biology</a> </td>
<td class = "rightside"> <?php $biologyThreads ->
obtainTopicThread("Natural Science and Mathematics", "Biology"); ?> </td>
</tr>
<tr>
<td class = "leftside"> <a href = "Subjects/Natural Science and
Mathematics/Chemistry.php">Chemistry</a> </td>
<td class = "rightside"> <?php $chemistryThreads ->
obtainTopicThread("Natural Science and Mathematics", "Chemistry"); ?>
</td>
</tr>
<tr>
<td class = "leftside"> <a href = "Subjects/Natural Science and
Mathematics/Physics.php">Physics</a> </td>
<td class = "rightside"> <?php $physicsThreads ->
obtainTopicThread("Natural Science and Mathematics", "Physics"); ?> </td>
</tr>
<tr>
<td class = "leftside"> <a href = "Subjects/Natural Science and
Mathematics/EarthScience.php">Earth Science</a> </td>
<td class = "rightside"> <?php $earthsciThreads ->
obtainTopicThread("Natural Science and Mathematics", "Earth Science"); ?>
</td>
</tr>
</tbody>
</table>

```

```
<td class = "leftside"> <a href = "Subjects/Natural Science and
Mathematics/Astronomy.php">Astronomy</a> </td>
```

```
<td class = "rightside"> <?php $astronomyThreads ->
obtainTopicThread("Natural Science and Mathematics", "Astronomy"); ?>
</td>
```

```
</tr>
```

```
<tr>
```

```
<td class = "leftside"> <a href = "Subjects/Natural Science and
Mathematics/Mathematics.php">Mathematics</a> </td>
```

```
<td class = "rightside"> <?php $mathThreads ->
obtainTopicThread("Natural Science and Mathematics", "Mathematics"); ?>
</td>
```

```
</tr>
```

```
</tbody>
```

```
</table>
```

```
<!-- International News -->
```

```
<table class = "maintable" style = "margin-bottom: 10px;">
```

```
<thead>
```

```
<tr>
```

```
<th class = "leftside"> Things happening outside the Internet
</th>
```

```
<th class = "rightside"> Latest Posts </th>
```

```
</tr>
```

```
</thead>
```

```
<tbody>
```

```
<tr>
```

```
<td class = "leftside"> <a href = "Subjects/World
News/News.php">International News</a> </td>
```

```
<td class = "rightside"> <?php $intnewsThreads ->
obtainTopicThread("World News", "International News"); ?> </td>
```

```
</tr>
```

```
</tbody>
```

```
</table>
```

```
<!-- Technology -->
```

```
<table class = "maintable">
```

```
<thead>
```

```
<tr>
```

```
<th class = "leftside"> What's going on now in Tech </th>
```

```
<th class = "rightside"> Latest Posts </th>
```

```
</tr>
```

```
</thead>
```

```
<tbody>
```

```

        <tr>
            <td class = "leftside"> <a href =
"Subjects/Technology/CompSoft.php">Computer Software/Programs</a>
</td>
            <td class = "rightside"> <?php $compsoftThreads ->
obtainTopicThread("Technology", "Computer Software"); ?> </td>
        </tr>
        <tr>
            <td class = "leftside"> <a href =
"Subjects/Technology/CompHard.php">Computer Hardware</a> </td>
            <td class = "rightside"> <?php $comphardThreads ->
obtainTopicThread("Technology", "Computer Hardware"); ?> </td>
        </tr>
        <tr>
            <td class = "leftside"> <a href =
"Subjects/Technology/VidGm.php">Video Games</a> </td>
            <td class = "rightside"> <?php $vidgmThreads ->
obtainTopicThread("Technology", "Video Games"); ?> </td>
        </tr>
        <tr>
            <td class = "leftside"> <a href =
"Subjects/Technology/Phones.php">Smartphones and Tablets</a> </td>
            <td class = "rightside"> <?php $phoneThreads ->
obtainTopicThread("Technology", "Smartphones"); ?> </td>
        </tr>
    </tbody>
</table>
</div>

```

```

    <!-- Link the Bootstrap 4 CDN JS and jQuery files (must be in this order:
jQuery, Popper, Bootstrap.js) -->

```

```

    <script
src='https://ajax.googleapis.com/ajax/libs/jquery/3.4.1/jquery.min.js'></script>
    <script
src='https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd/popper.min.js'
></script>
    <script
src='https://maxcdn.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js'></scr
ipt>
    </body>

```

```

</html>

```

6. What you also want to do is make sure to close the database connections on all the includes folder files. To do this, at the very end of each class call, you want to add a destruct method that will be called at the end of each class call and will close the database connection. Add the follow to the end of each class in each includes file:

```
//closes the database connection at the end of each class call
public function __destruct()
{
    $this -> dbConn -> close();
}
```

Or in some pages

```
//closes the database connection at the end of each class call
public function __destruct()
{
    $this -> db -> close();
}
```

This will close the database connection after the end of each method call

(make sure not to add this to the Connection class of the database.php page, else it will close the database connection immediately before any class can even call the Connection class)

**10/12/19**

Before uploading this website, what you want to do is implement the code for the “Forgot Password” page. You want users to be able to change their passwords in the event that they forget their password.

Create a new page in the Discussion Forum folder called “forgotpassword.php” and the corresponding CSS file, and this page will handle giving the users the ability to change their password:

**forgotpassword.php**

```
<?php
    include "includes/AccountHandler.php";
    $account = new Accounts();
?>
<html lang = "en">
    <head>
        <title> Forgot Password? Change password right away </title>
        <meta charset = "utf-8"/>

        <link rel = "stylesheet" href = "css/forgotpassword.css"/>
    </head>

    <body>
        <div id = "forgotpwdWindow">
            <h4> Forgot password? Enter your username and email to change your password
            right away </h4>

            <form method = "POST" action = "forgotpassword.php" id = "information-input">
                <input type = "text" name = "uname" placeholder = "Enter Username"/>
                <input type = "email" name = "uemail" placeholder = "Enter Email"/>
                <input type = "password" name = "newPwd" placeholder = "Enter new
password"/>
                <input type = "submit" value = "Submit Info" name = "submitButton"/>
            </form>
            <p id = "errorMessage" style = "font-family: serif;"></p>
            <?php
                if(isset($_POST['submitButton']))
                {
                    $UserName = $_POST['uname'];
                    $Email = $_POST['uemail'];
                    $NewPwd = $_POST['newPwd'];

                    $account -> ChangePwd($UserName, $Email, $NewPwd);
                }
            ?>
```

```
        <a href = "loginPage.php" id = "backtoLogin"> <h5>Go back to the login page</h5>
    </a>
</div>
</body>
</html>
```

### **forgotpassword.css**

```
body
{
    background-color: #191970;
}

#forgotpwdWindow
{
    background-image:
url("https://png.pngtree.com/thumb_back/fh260/back_pic/04/00/23/4757fc9e77387db.jpg");
    background-repeat: no-repeat;
    background-size: cover;

    border: 1px solid black;

    text-align: center;

    margin-top: 5%;
    margin-right: 20%;
    margin-left: 20%;

    box-shadow: 5px 5px black;
}

#forgotpwdWindow h4
{
    color: #EEE8AA;
    text-align: center;
}

#forgotpwdWindow #backtoLogin
{
    text-decoration: none;

    font-family: serif;

    color: white;
}

#forgotpwdWindow #information-input input
{
```

```
display: block;
}

#forgotpwdWindow #information-input input[type=text], #forgotpwdWindow #information-
input input[type=email],
#forgotpwdWindow #information-input input[type=password]
{
    width: 50%;
    height: 5%;

    font-size: 20px;
    font-family: "Times New Roman";
    background-color: #FAF0E6;

    border-radius: 5px;

    margin-left: 20%;
}
#forgotpwdWindow #information-input input[type=submit]
{
    background-color: #4169E1;

    width: 30%;
    height: 5%;

    font-family: cursive, "Times New Roman";
    font-size: 20px;
    color: #F0FFFF;

    border: none;
    border-radius: 5px;

    cursor: pointer;

    margin-left: 20%;
}
```

In the AccountHandler.php includes file, in the Accounts class, create a new method called ChangePwd(), and this method will be responsible for changing the password of the user if they forgot it:

#### AccountHandler.php

```
<?php
    include_once 'database.php';

    /* This class will handle the login, logout and signup processes */
    class Accounts
    {
        private $dbConn;
        private $sql1; //For the login(). For logging in new users to the forumhome.php

        //Both are for the signup() method. For creating new user accounts
        private $sql2; //For the signup() method. For adding new users to the accounts table
        private $sql3; //For the signup() method. For adding new users to the activeaccounts table
        private $sql4; //For the signup() method. To check if a username already exists in the
        accounts database

        //The following variables and methods are to change user password if they forgot
        their password
        private $sql5; //For the ChangePwd() method. To retrieve the account information of
        the user
        private $sql6; //For the ChangePwd() method. To change the password of user who
        forgot their password

        /**
         * initialize connection to the discussionforum database upon calling the object of the
         class */
        public function __construct()
        {
            $this -> dbConn = new Connection();
            $this -> dbConn = $this -> dbConn -> connect();
        }

        /* This method is for logging in new users into the forum */
        public function login($username, $password)
        {
            if(empty($username) && empty($password))
            {
                echo "<p id = 'errormessage'> Do not submit blank information! </p>";
            }
        }
    }
}
```



```

    }
    else if(empty($username))
    {
        echo "<p id = 'errormessage'> You must enter your username! </p>";
    }
    else if(empty($password))
    {
        echo "<p id = 'errormessage'> You must enter your password! </p>";
    }
    else
    {
        $this -> sql1 = "SELECT * FROM accounts WHERE user_username=? OR
user_password=?;";

        $stmt = $this -> dbConn -> stmt_init();

        if(!$stmt -> prepare($this -> sql1))
        {
            echo "<script> alert('SQL ERROR: " . $stmt -> error . "'); </script>";
        }
        else
        {
            $stmt -> bind_param("ss", $username, $password);
            $stmt -> execute();

            $result = $stmt -> get_result();

            if($row = $result -> fetch_assoc()) //store the results in an associative array
            {
                session_start();

                /*obtain the id, username, email, firstname, lastname as well as profile picture
                and status and registration date, and stores them in session variables*/
                $_SESSION['userid'] = $row['user_id'];
                $_SESSION['firstname'] = $row['user_firstname'];
                $_SESSION['lastname'] = $row['user_lastname'];
                $_SESSION['useremail'] = $row['user_email'];
                $_SESSION['password'] = $row['user_password'];
                $_SESSION['username'] = $row['user_username'];
                $_SESSION['pic'] = $row['profile_pic'];
                $_SESSION['regdate'] = $row['reg_date'];
                $_SESSION['status'] = $row['user_status'];

                //checks to see if the password the user typed in matches the hashed password in
                the accounts table
                if(password_verify($password, $_SESSION['password']))

```

```

        {
            //sends user to the forum homepage upon successful login
            header("Location: forumhome.php?login=successful");
            exit();
        }
        else
        {
            echo "<p id = 'errormessage'> Username exists but password is incorrect! </p>";
        }
    }
    else
    {
        echo "<p id = 'errormessage'> There is no such account! </p>";
    }
}
}

/*****
*****/

/* This method is for registering new users and adding their information to the accounts
table */
public function signup($accountCreated, $firstname, $lastname, $email, $username,
$password)
{
    //This is to check if the username a user entered already exists in the accounts table
    $this -> sql4 = "SELECT user_username FROM accounts where
user_username='$username'";
    $stmt3 = $this -> dbConn -> stmt_init();
    if(!$stmt3 -> prepare($this -> sql4))
    {
        echo "<script> alert('SQL ERROR: " . $stmt3 -> error . "'); </script>";
    }
    else
    {
        $stmt3 -> execute();
        $result = $stmt3 -> get_result();

        if(empty($firstname) && empty($lastname) && empty($email) &&
empty($username) && empty($password))
        {
            echo "<p id = 'errormessage'> Do not submit empty information! </p>";
        }
        else if(empty($firstname) || empty($lastname) || empty($email) || empty($username) ||
empty($password))

```

```

{
    echo "<p id = 'errormessage'> You must fill out everything! </p>";
}
else
{
    //if a username is already in the accounts table
    if($result -> num_rows == 1)
    {
        echo "<p id = 'errormessage'> Username already exists! </p>";
    }
    //if username does not exist in the accounts table
    else
    {
        $this -> sql2 = "INSERT INTO accounts(reg_date, user_firstname, user_lastname,
user_email, user_username, user_password)
                        VALUES(?, ?, ?, ?, ?, ?);";

        $stmt1 = $this -> dbConn -> stmt_init();

        if(!$stmt1 -> prepare($this -> sql2))
        {
            echo "<script> alert('SQL ERROR: " . $stmt1 -> error . "'); </script>";
        }
        else
        {
            $hashedPwd = password_hash($password, PASSWORD_DEFAULT);

            $stmt1 -> bind_param("ssssss", $accountCreated, $firstname, $lastname, $email,
$username, $hashedPwd);
            $stmt1 -> execute();

            //now insert the username in the activeaccounts table
            $this -> sql3 = "INSERT INTO activeaccounts(user_username)
                            VALUES('$username');";

            $stmt2 = $this -> dbConn -> stmt_init();

            if(!$stmt2 -> prepare($this -> sql3))
            {
                echo "<script> alert('SQL ERROR: " . $stmt2 -> error . "'); </script>";
            }
            else
            {
                $stmt2 -> execute();
            }
        }
    }
}

```

```
        echo "<p id = 'successmessage'> You have successfully signed up! Sign in now  
to enter forum discussion! </p>";
```

```
    }  
  }  
}  
}
```

```
/******  
*****/
```

```
/* The method changes the password of users who forgot their password */  
public function ChangePwd($username, $email, $newpwd)  
{  
  if(empty($username) || empty($email) || empty($newpwd))  
  {  
    echo "<script>  
      var errmessage = document.getElementById('errMessage');  
  
      errmessage.innerHTML = 'You must enter both your username, email  
and new password!';  
      errmessage.style.color = 'red';  
    </script>";  
  }  
  else  
  {  
    //first obtain the user account information  
    $this -> sql5 = "SELECT * FROM accounts WHERE user_email='$email' AND  
user_username='$username'";  
  
    $stmt1 = $this -> dbConn -> stmt_init();  
  
    if(!$stmt1 -> prepare($this -> sql5))  
    {  
      echo "<script> alert('SQL ERROR: " . $stmt1 -> error . "'); </script>";  
    }  
    else  
    {  
      $stmt1 -> execute();  
      $result = $stmt1 -> get_result();  
  
      if($result -> num_rows == 0)  
      {  
        echo "<script>  
          var errmessage = document.getElementById('errMessage');
```

```

        errmessage.innerHTML = 'The information provided does not match!';
        errmessage.style.color = 'red';
    </script>";
}
//if there is a match, change the password of that user
else
{
    $hashedPwd = password_hash($newpwd, PASSWORD_DEFAULT);

    $this -> sql6 = "UPDATE accounts SET user_password='$hashedPwd'
WHERE user_email='$email' AND user_username='$username'";

    $stmt2 = $this -> dbConn -> stmt_init();

    if(!$stmt2 -> prepare($this -> sql6))
    {
        echo "<script> alert('SQL ERROR: " . $stmt2 -> error . "'); </script>";
    }
    else
    {
        $stmt2 -> execute();

        echo "<script>
            var errmessage = document.getElementById('errMessage');

            errmessage.innerHTML = 'Password successfully changed!';
            errmessage.style.color = 'green';
        </script>";
    }
}
}
}
}

/*****
*****/

//closes the database connection at the end of each class call
public function __destruct()
{
    $this -> dbConn -> close();
}
}
?>

```

With all this finished, users can change their password if they forgot it.

**10/13/19**

Now that this project is completed, rename the website as “IntelliChat”. For now, you want to upload and host this site on *000webhost.com*. Since it’s only for putting in your portfolio, there’s no reason to buy a domain name and purchase a server for it. 000webhost is free of charge. It may be slow and unreliable, but again, it’s just for show.