# Contents

Expectations

Before starting this course, I had a few expectations. Before jumping directly to the expectations, I need to explain where my level was at before starting the course. I have built a few WordPress websites which means I have a good foundation and knowledge of how a website should be designed and structured. However, I had zero knowledge in writing code and creating databases. I expected from this course to learn the basics of HTML, CSS, PHP, SQL, and JS. I also wanted to be able to build a website that has a great design and has a fully functional back end.

What is my website and how does it work?

I created a food order website with a fully functional back end. The Restaurant that I took the images from is 9 pyramids lounge because they had amazing photography which will help me make my website look amazing.

The front end of the website is divided into 3 main pages the home, Menu, and contact us. All the three websites have the same navigation menu and hero section
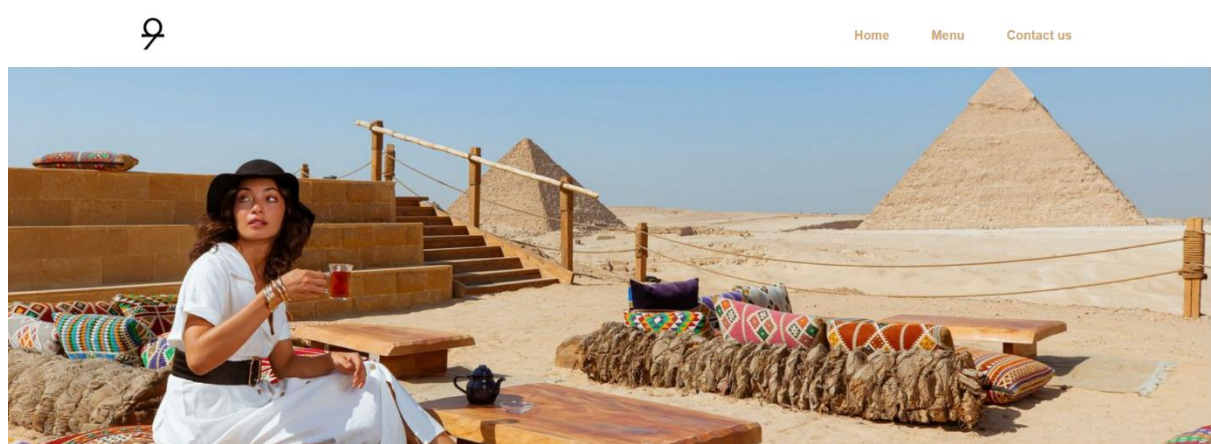
The back end is also divided into 3 main pages that do different functionalities mange admins, mange feedback and mange orders. The back end is also only accessible if you are an admin and to access it you type /admin after the URL and it will redirect you to the login page you can use (farou2,133) as your admin username and password.
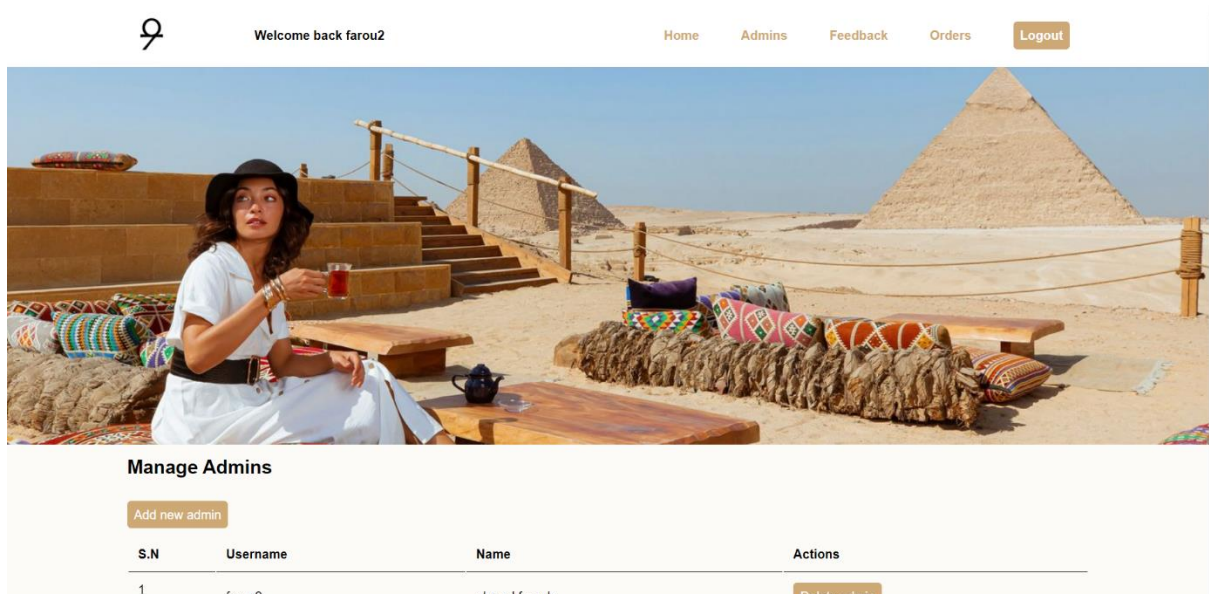
The design of my website

As I mentioned in the expectations section, I do have some knowledge in designing websites, so the first step was to create a color pallet for the website to do so I used (color.adobe.com) to help me choose the right colors the final 5 colors I used where black, white, and three different shades of yellow.

After choosing the colors I had to choose the fonts and the button design to do so I used adobe XD a free and easy tool that helped me to visualize my website before writing any code.

 Now I am ready to start designing the actual website in XD because I have all the assets that I need.
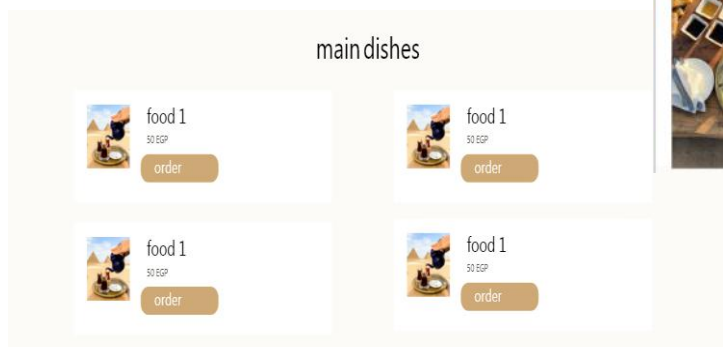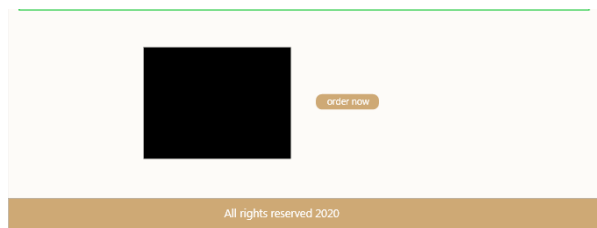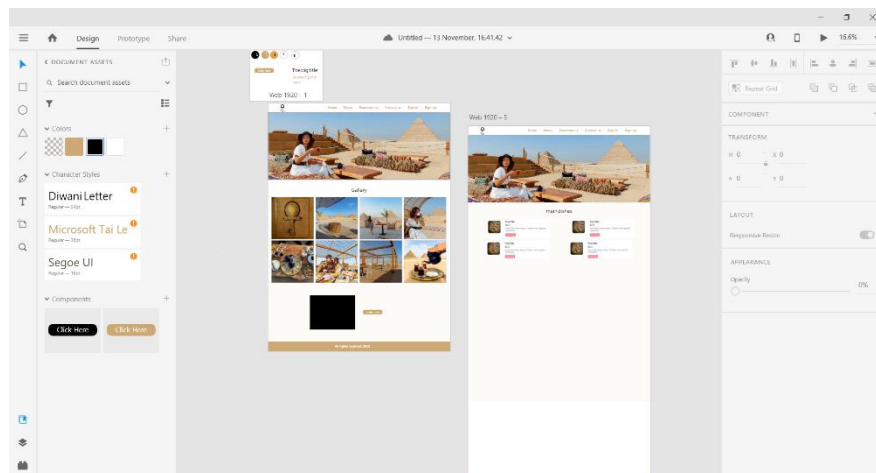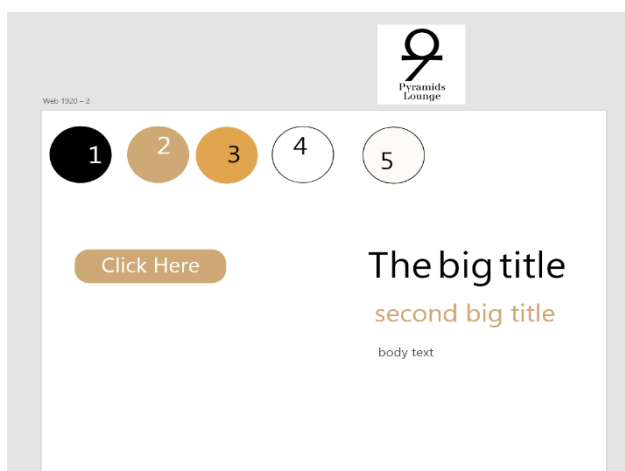
The structure and the code

Html

```
    <!-- start of top nav --><!-- ----------------------------------------
------------------------------------------------------ -->
    echo'
        <section class="top_nav">
       <div class="container">
         <div class="logo">
           <a href="http://localhost/course_work_one/" title="Logo">
           <img src="imges/logo.png" alt="Restaurant Logo" width="100%">
           </a>
         </div>

           <div class="menu-right">
           <ul>
               <li>
                   <a href="index.php">Home</a>
               </li>
               <li>
                   <a href="menu.php">Menu</a>
               </li>
               <li>
                   <a href="contact.php">Contact us</a>
               </li>
           </ul>
           <div class="clear-fix"></div>
         </div>
         </div>
      </section>'
    <! -- end of top nav --><! -- ----------------------------------------
------------------------------------------------------ -->
```
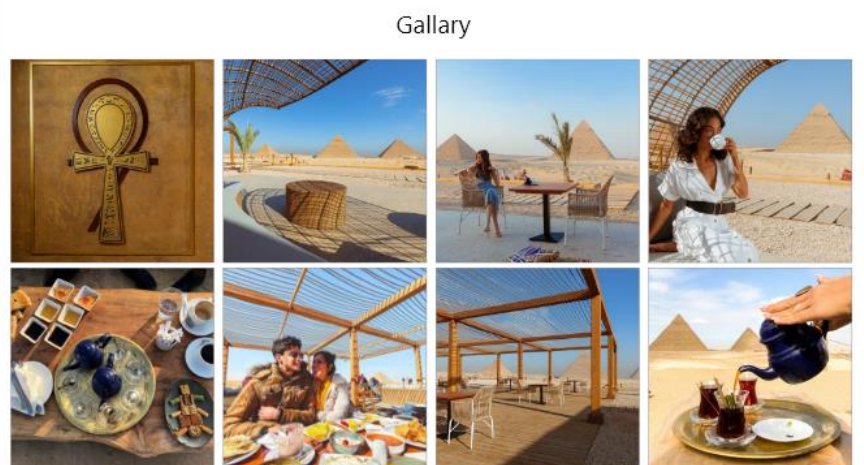
The above section is the navigation every section in the page has a </section> flowed by a container div the reason I have the div container is because It will help me style the page giving all the website the same container div.

The second div is the logo div very simple with just an <a> attribute the reason I put the image in an <a> attribute is that I wanted the logo to redirect the user to the home page.

The third part is the menu itself it is also very simple with just an unordered list with links inside it to allow me to navigate around the website.

```
// <!-- hero section --><!-- ------------------------------------------------------------------------ -->
    echo'

    <section class="hero">
      <div class="container"></div>
    </section>'
    // <!-- end of hero section --><!-- -------------------------------------------------------------- -->
```

after the nav is a very simple div that I then styled to have an image as a background.

```
// <!-- hero section  --><!-- ----------------------------------------------------------------------- -->
    echo'

    <section class="footer">
    <div class="container">
      <p class="all-rights">All rights reserved</p>
    </div>

    </section>'
    // <!-- end of hero section  --><!-- ------------------------------------------------------------- -->
```

The footer of the website is also very simple with just one <p> paragraph tag.

The 3 pervious sections will be the same threw out all the websites.

```
    <!-- gallary section  -->
    <section class="gallary">
      <div class="container">
        <h2 class="text-center">Gallary</h2>

        <div class="img"><img src="imges/1.jpg" alt="pic 1" width="100%"></div>
        <div class="img"><img src="imges/2.jpg" alt="pic 1" width="100%"></div>
        <div class="img"><img src="imges/3.jpg" alt="pic 1" width="100%"></div>
        <div class="img"><img src="imges/4.jpg" alt="pic 1" width="100%"></div>
        <div class="img"><img src="imges/5.jpg" alt="pic 1" width="100%"></div>
        <div class="img"><img src="imges/6.jpg" alt="pic 1" width="100%"></div>

        <div class="clear-fix"></div>
        <div class="clear-fix"></div>

      </div>

    </section>
    <!-- end of gallary  -->
```

The gallery section is a simple div with a image inside of it flowed by a div that has a clear-fix class I will explain why we need this div in the CSS part.

```
    <!-- start of contact  --><!-- ------------------------------------------------------------------- -->
    <section class="contact-s">
      <div class="container">
        <div class="contact">
          <form action="contact.php" method="POST" name="contact_us" onsubmit="return validateContact()">
            <label for="fname">First Name</label>
            <input type="text" id="fname" name="first_name" placeholder="Enter Your First Name">

            <label for="lname">Last Name</label>
            <input type="text" id="lname" name="last_name" placeholder="Enter Your Last Name">

            <label for="email">Email</label>
            <input type="email" id="email" name="email" placeholder=" Enter Your Email">

            <label for="subject">Subject</label>
            <input id="subject" type="text" name="subject" style="height:200px">

            <input type="submit" name="submit" value="Submit" class="btn" style="padding: 1%;">
          </form>
        </div>
      </div>
    </section>
    <!-- end of contact  --><!-- --------------------------------------------------------------------- -->
```

The contact form is another simple html structure of a form that uses the post method(explained in the php section) and an on submit attribute to help me in the java script.

```html
<!-- table section  -->
<section class="action main">
  <div class="container">
    <h1>Manage Admins </h1>

    <br>
    <br>

    <a href="add_admin.php" class="btn" style="padding: 0.7%;">Add new admin</a>
    <br>
    <br>


    <table class="full-tbl">
      <tr>
        <th>S.N</th>
        <th>Username</th>
        <th>Name</th>
        <th>Actions</th>
      </tr>
    <!-- ---------------------------------------------------------------------- -->
```

this table is displayed multiple times in the admin panel and the structure of it is one simple heading <h1> and it has 4 main table headings.

CSS

The CSS was one of the challenging parts of the website because one of my objectives was to create a great looking website and some times thigs just did not work out. But fortunately every thing worked in the end

```css
*{
  padding: 0 0;
  margin: 0 0;
  font-family: Arial, Helvetica, sans-serif;
}
```

 The first part is to use the * to target all the elements in the page I then choose my font and 2 more backup fonts if the browser was unable to display one of them. And I also removed all the padding and margins to give me a cleaner design.

```css
.container{
  width: 80%;
  margin: 0 auto;
  padding: 1%;
}
```

 The container class is the class that all parts of the website have and I gave it 1% of padding to make the website have some space and to make it look better.

```css
.btn{
  padding: 2%;
  border: none;
  font-size: 1rem;
  border-radius: 5px;
  background-color: #CEA975;
```

this is the style of all the buttons in the page and I styled it according to my Adobe XD design.

```
.text-center{
    text-align: center;
    padding: 1%;
    font-weight: 100;
}
```

The text centre is a class that I gave to any subheading on the website and it allowed me to put it in the centre.

The nav styles

```
.logo{
    width: 5%;
    float: left;
}
```

```
.menu-right{
    text-align: right;
    line-height: 50px;
}
```

```
ul{
    list-style-type: none;
}
```

```
li{
    display: inline;
    padding: 2%;
    font-weight: bold;
}
```

The logo I gave a 5% width I found this to be the perfect width for this certain logo. The menu-right is the part that made the menu to the right. The ul had to be in this style to make sure it does not add a dot after every menu item. The li had an inline display to make sure they are all displayed next to each other.

```
/* footer section */
.footer{
    background-color: #CEA975;
    color: white;
}
```

```
/* hero section */
.hero{
    background-image: url(../imges/hero.jpg);
    background-size: cover;
    background-repeat: no-repeat;
    background-position: center;
    padding: 15%;
}
```

The footer has a and a white and gold colors and the hero section had the simple hero image in the images folder.

The above sections of CSS where the main classes that I used threw out all the website

```
.full-tbl{
    width: 100%;
}
```

```
table tr td{
    padding: 1%;
}
```

```
table tr th{
    border-bottom: 1px solid black;
    padding: 1%;
    text-align: left;
}
```

This is the table CSS the width is 100% to make sure it fills all the page and a text align and a border to make it look more presentable.

```
.contact-s{
    padding: 4% 0;
    background-color: #FDFBF8;
}
input[type=text], select, textarea {
    width: 100%;
    padding: 12px;
```

```css
    border: 1px solid #CEA975;
    border-radius: 4px;
    margin-top: 6px;
    margin-bottom: 16px;
    resize: vertical;
}

 input[type=email], select, textarea {
   width: 100%;
   padding: 12px;
   border: 1px solid #CEA975;
   border-radius: 4px;
   margin-top: 6px;
   margin-bottom: 16px;
   resize: vertical;
}


input[type=submit] {
   padding: 1%;
   border: none;
   font-size: 1rem;
   border-radius: 5px;
   background-color: #CEA975;
   color: white;
}
```
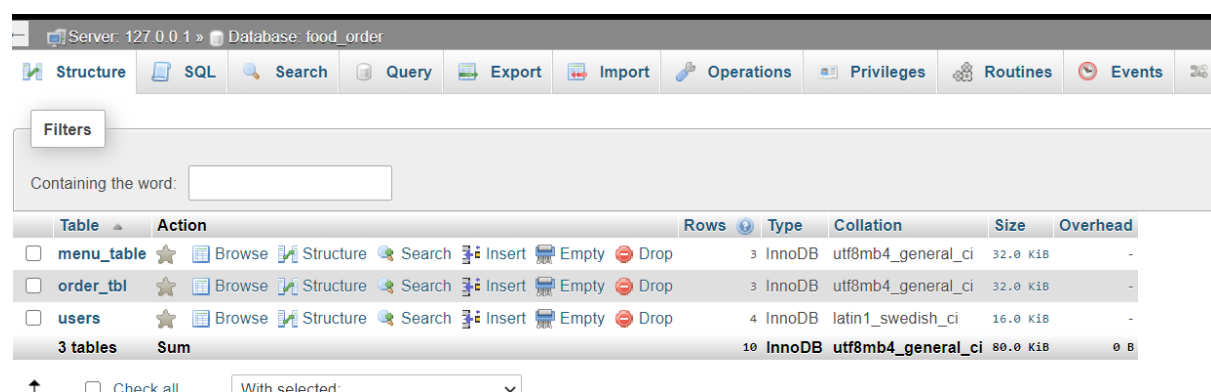
This is the styles for all the forms in the website a key element in designing a good website is to stay consistent use the same fonts the same colours and the same form styles.

PHP and SQL

My database is named food_order and it has 3 tables inside it one for feedback the other is for admins and the third is for orders. All the 3 tables have a row for id and it is auto increment which means that it is add one every time.

The php sections in my website is divided into a few actions a section that makes me login, sign up, and present data. Before we jump into all the different sections I will explain what partials are. As

you can see on the website there are some parts are the same in every page for example the nav and the footer. Instead of repeating the code in every page we put the html in a php file and call it with the include once tag. This is what I did for the nav, hero, and footer.

Adding to the data base

To add to the data base there are a few steps the first is to take the input from the form and put them in variables.

```php
$first_name = filter_input(INPUT_POST, 'first_name');
$last_name = filter_input(INPUT_POST, 'last_name');
$email = filter_input(INPUT_POST, 'email');
$subject = filter_input(INPUT_POST, 'subject');
```

The second step is to validate any of the inputs if I do not do this step every time I refresh the page a new empty row will appear in the data base.

```php
if (!empty($first_name)){
if (!empty($email)){
```

after validating the input on 2 of the variables it is time to create a connection.

```php
$host = "localhost";
$dbusername = "root";
$dbpassword = "";
$dbname = "food_order";

$conn = new mysqli ($host, $dbusername, $dbpassword, $dbname);
```

After creating the connection, I can now insert the data into the SQL

```php
$sql = "INSERT INTO menu_table (first_name,last_name,email,subject)
values ('$first_name','$last_name','$email','$subject')";
```

after the SQL command we should execute the command and then close the connection.

```php
if ($conn->query($sql)){
echo "Thank You";
}
else{
echo "eroor";
}
$conn->close();
```

Fetching from the data base

First, we need to create a connection

```php
$dbhost = 'localhost';
$dbuser = 'root';
$dbpass = '';
$dbname = 'food_order';
$conn = mysqli_connect($dbhost, $dbuser, $dbpass,$dbname);
```

After creating the connection, we need to wright to SQL code and execute it

```php
$sql = 'SELECT * FROM users';
$result = mysqli_query($conn, $sql);
```

Now we will have a if condition if there is data in the table it will loop around it and display it in my table if there is no data in the table it will not display anything.

```php
if (mysqli_num_rows($result) > 0) {
    while($row = mysqli_fetch_assoc($result)) {
        $id = $row["id"]. "<br>";
        $username = $row["username"]. "<br>";
        $name = $row["name"]. "<br>";
        ?>

        <tr>
        <td><?php echo $id; ?>. </td>
        <td><?php echo $username; ?></td>
        <td><?php echo $name; ?></td>
        <td><a href="execute_actions/delete_admin_exe.php" class="btn"
 style="padding: 2.5%;"> Delete admin</a></td>
        </tr>
```

The final step is to close the connection.

Sessions

To make sure that only admins can access the admin panel we need to have a login page that creates a session and if there is no session it will not allow you to access the page.

First, we need to create a connection like the previous examples.

Cheek if the submit button is clicked or not if it is clicked, we can then proceed to the next part

```php
if(isset($_POST['but_submit'])){
```
get the values from the form and put them In variables

```php
$uname = mysqli_real_escape_string($con,$_POST['txt_uname']);
$password = mysqli_real_escape_string($con,$_POST['txt_pwd']);
```

Write and execute the SQL command

```
        $sql_query = "select count(*) as cntUser from users where username='".
$uname."' and password='".$password."'";
        $result = mysqli_query($con,$sql_query);
        $row = mysqli_fetch_array($result);

        $count = $row['cntUser'];

        if($count > 0){
            $_SESSION['uname'] = $uname;
            header('Location: index.php');
            // echo "hell yeah";
        }else{
            echo "Invalid username and password";
        }
```

Like this we created a session with the username of the admin we can then use this to display it in the admin panel now it is time to restrict access to all the admin pages except if they are in a session to do that we go to every admin page and in the top of the page cheek if there is a session open if not redirect to the login admin page.

```
if(!isset($_SESSION['uname'])){
  header('Location: loginadmin.php');
}
```

Now lets see the logout functionality it executed when the admin clicks on the logout button

```
    session_destroy();
    header('Location:loginadmin.php');
```
and this here all It does is destroy the session the admin is in.

Java script
Java script is a very powerful programing language. But in this course, I only used java script to validate the inputs in a field and it is empty the border will change to the color red.

```
function validateContact() {
  var x = document.forms["contact_us"]["first_name"].value;
  var y = document.forms["contact_us"]["last_name"].value;
  var z = document.forms["contact_us"]["email"].value;
  var a = document.forms["contact_us"]["subject"].value;
  if (x == "") {
    document.getElementById("fname").style.border = "1px solid #E54059";
    return false;
  }
  else{document.getElementById("fname").style.border = "1px solid #CEA975";}

// ---------------------
```

Here I took the input in the form and put it in a variable I then check if the variable empty and if it is empty change the border to red and if it is not empty change, it back to the original color.

Learned outcomes and reflection

As I mentioned in the expectations, I had no experience in writing code nether in the front end nor the back end. And after finishing the course I was surprised by the work that I was able to produce It defiantly exceeded my own expectations. I was able to create an amazing design with a fully functional back end. I was able to gasp a lot of information in such a very short amount of time. I definitely do not regret any hour I studied in the course because it was all worth it at the end. The only drawback I have with this course is the timing as it is a very big course and we had to cover it in a short period of time I believe if we had more time I would have learned more and produced even better project.