

BAB I

RUMUSAN MASALAH DAN TUJUAN

A. Rumusan Masalah

1. Bagaimana fungsi *calculate_items(\$cart)* untuk menghitung jumlah item yang ada di shopping cart, *calculate_price(\$cart)* untuk menghitung total harga item yang ada di shopping cart, dan *display_cart(\$cart)* untuk menampilkan isi shopping cart?
2. Bagaimana membuat form *login* dengan masukan username dan password serta file *logout*, dan bagaimana membuat kode program pada file *view_books.php* sehingga hanya dapat dibuka oleh customer yang sudah login?
3. Bagaimana membuat program untuk menampilkan detail data customer dari select item HTML?

B. Tujuan

1. Mengetahui fungsi *calculate_items(\$cart)* untuk menghitung jumlah item yang ada di shopping cart, *calculate_price(\$cart)* untuk menghitung total harga item yang ada di shopping cart, dan *display_cart(\$cart)* untuk menampilkan isi shopping cart.
2. Mengetahui cara membuat form *login* dengan masukan username dan password serta file *logout*, dan mengetahui cara membuat kode program pada file *view_books.php* sehingga hanya dapat dibuka oleh customer yang sudah login.
3. Mengetahui cara membuat program untuk menampilkan detail data customer dari select item HTML.

BAB II

DASAR TEORI

A. Cookies dan Session

Cookies atau bisa disebut HTTP cookies, web cookies, atau browser cookies adalah data atau informasi yang diciptakan oleh suatu website untuk disimpan di web browser, ketika user sedang menjelajahi website tersebut. Cookies yang telah disimpan pada browser bisa diambil untuk memberitahu website tentang aktifitas user pada waktu sebelumnya.

Sedangkan Session adalah salah satu fasilitas yang ada pada PHP yang digunakan untuk menyimpan data sementara ke dalam variabel (variabel session) sehingga data tadi dapat diakses oleh client selama variabel session tadi tidak di kosongkan atau dihilangkan. Nilai variabel di dalam session di simpan di sisi server (web server). Berbeda dengan cookies yang nilai variabelnya disimpan di sisi client (browser). Jadi session relatif lebih aman digunakan untuk menyimpan variabel nilai yang bersifat rahasia seperti username dan password pada saat login.

B. AJAX dan jQuery

AJAX, singkatan dari “*Asynchronous JavaScript and XML*“, merupakan metode suatu laman web menggunakan JavaScript untuk mengirim dan menerima data dari server tanpa harus menyegarkan (refresh) laman itu. XML adalah sejenis markup language – seperti HTML, yang kerap dipakai untuk mengirimkan data melalui internet.

Sedangkan jQuery adalah sebuah library yang dibangun dengan menggunakan JavaScript untuk mengotomasi dan menyederhanakan perintah-perintah umum. Meskipun ada banyak library lain semacamnya, namun jQuery jauh lebih populer karena kemampuannya untuk menjalankan perintah pada peramban lama. jQuery berjalan pada browser bersamaan dengan JavaScript biasa.

BAB III

PEMBAHASAN

A. Fungsi `calculate_items($cart)`, `calculate_price($cart)`, dan `display_cart($cart)` pada shopping cart.

1) Soal

Lengkapi kode program untuk fungsi `calculate_items($cart)` untuk menghitung jumlah item yang ada di shopping cart, `calculate_price($cart)` untuk menghitung total harga item yang ada di shopping cart, dan `display_cart($cart)` untuk menampilkan isi shopping cart.

2) Source Code

show_cart.php

```
<?php
//File : show_cart.php
//Deskripsi : untuk memproses dan menampilkan shopping cart
session_start();
$isbn = $_GET['isbn'];

// Mendapatkan nilai harga items
require_once('db_login.php');
$con = mysqli_connect($db_host, $db_username,
$db_password, $db_database);
if (mysqli_connect_errno()){
    die ("Could not connect to the database: <br />".
mysqli_connect_error( ));
}
$query = " SELECT * FROM books WHERE isbn = '$isbn' ";
$result = mysqli_query($con,$query);
if (!$result){
    die ("Could not query the database: <br />".
mysqli_error($con));
}
$row = mysqli_fetch_assoc($result);
$harga = $row['price'];

if($isbn){
    if (!isset($_SESSION['cart'])){
        $_SESSION['cart'] = array();
        $_SESSION['total_item'] = 0;
        $_SESSION['total_price'] = 0;
    }
    if (isset($_SESSION['cart'][$isbn])){
        $_SESSION['cart'][$isbn]++;
    } else {
        $_SESSION['cart'][$isbn]++;
    }
    $_SESSION['total_items'] =
calculate_items($_SESSION['cart']);
```

```

        $_SESSION['total_price'] =
        calculate_price($_SESSION['cart']);
        display_cart($_SESSION['cart']);
        echo '<br /><br /><a href="view_books.php">Continue
Shopping</a>';
    }

    function calculate_items($array) {
        return count($array);
    }

    function calculate_price($array) {
        global $harga;
        $price = $_SESSION['total_price'] + $harga;
        return $price;
    }

    function display_cart($array) {
        echo '
        border="1"><tr><th>ISBN</th><th>Jumlah</th></tr>';
        foreach ($array as $value) {
            echo '<tr><td>'.key($array).</td><td>'.$value.' buah
        </td><br/>';
            next($array);
        }
        echo '</table>';
        echo '
        <br/>Total Items :
        '.$_SESSION['total_items'].'<br/>';
        echo 'Total Harga : '.$_SESSION['total_price'].'<br/>';
    }
    ?>

```

3) Screenshot

ISBN	Jumlah
0-672-31509-2	1 buah
0-672-31697-8	2 buah

Total Items : 2
Total Harga : 94.97

[Continue Shopping](#)

4) Pembahasan

Pada fungsi yang pertama yaitu `calculate_items($array)` kita memasukkan `$_SESSION['cart']` sebagai nilai argumen yang dimana

didalam fungsi `calculate_items($array)` kita return dengan nilai fungsi `count` dari argumen yang dimasukkan, artinya fungsi `calculate_items($array)` akan mengembalikan nilai total jenis barang yang masuk ke dalam cart, bukan total semua jenis dikalikan jumlah per jenisnya. Hal ini karena informasi total barang pada tiap jenis sudah tersedia pada fungsi `display_cart($array)`.

Kemudian pada fungsi `calculate_price($array)`, kita mengambil nilai `$harga` dari barang yang sedang dimasukkan ke dalam cart, dengan mendapatkan `isbn` barang dari method `get`, kemudian dimasukkan ke variable `$isbn`. Kemudian nilai `$harga` ini kita tambahkan ke `$_SESSION['total_price']`, yang mana sudah memuat total harga sebelumnya.

Kemudian yang terakhir fungsi `display_cart($array)` memasukkan `$_SESSION['cart']` sebagai argumen, dengan menampilkan semua informasi yang tersimpan di `$_SESSION['cart']` yaitu nilai ISBN dan jumlah item tiap nilai ISBN-nya. Nilai ISBN kita dapat dengan fungsi `key($_SESSION['cart'])` karena fungsi `key()` akan mengembalikan nilai key nya, bukan nilai yang disimpan dalam key tersebut. Kemudian untuk mengambil nilai yang disimpan dalam key tersebut kita panggil dengan `$value`, karena pada perulangan `foreach` tersebut kita gunakan *`$array as $value`* yang dimana tiap nilai `$array` dikembalikan ke variable `$value`.

Kemudian pada bawahnya juga kita tampilkan total item dan total harga dengan mengambil dari `$_SESSION[total_items]` dan `$_SESSION['total_price']`.

B. Membuat file login , logout, dan membuat file view_books.php hanya dapat dibuka oleh customer yang sudah login

1) Soal

Latihan Session

Buatlah sebuah form login dengan masukan username dan password. Sebelumnya edit tabel customer, tambahkan field username dan password. Username berupa email.

Buatlah file untuk login dan logout, kemudian tambahkan kode program pada file view_books.php sehingga hanya dapat dibuka oleh customer yang sudah login.

2) Source Code

login.php

```
<html>
<head>
  <title>Login Page</title>
  <link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css">
  <script
src="https://ajax.googleapis.com/ajax/libs/jquery/1.12.4/jquery.min.js"></script>
  <script
src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js"></script>
  <style type="text/css">
    .login-form {
      width: 340px;
      margin: 50px auto;
    }
    .login-form form {
      margin-bottom: 15px;
      background: #f7f7f7;
      box-shadow: 0px 2px 2px rgba(0, 0, 0, 0.3);
      padding: 30px;
    }
    .login-form h2 {
      margin: 0 0 15px;
    }
    .form-control, .btn {
      min-height: 38px;
      border-radius: 2px;
    }
    .btn {
      font-size: 15px;
      font-weight: bold;
    }
  </style>
</head>
<body>
  <div class="login-form">
    <form name="formlogin" method="post" action="login_check.php">
```

```

        <?php
            if (isset($_GET['pesan'])) {
                if ($_GET['pesan']=="gagal") {
                    echo "<div style='text-align:center'
class='alert alert-danger' role='alert'>Username dan Password
tidak sesuai !</div>";
                } else if ($_GET['pesan']=="blmlogin") {
                    echo "<div style='text-align:center'
class='alert alert-danger' role='alert'>Kalo pengen liat buku
Login dulu !</div>";
                }
            }
        ?>
        <h2 class="text-center">Halaman Login</h2>
        <div class="form-group">
            <input type="text" class="form-control"
placeholder="Username" name="username" required="required">
        </div>
        <div class="form-group">
            <input type="password" class="form-control"
placeholder="Password" name="password" required="required">
        </div>
        <div class="form-group">
            <button type="submit" name="Submit" class="btn btn-
primary btn-block">Log in</button>
        </div>
    </form>
</div>
</body>
</html>

```

login_check.php

```

<?php
    session_start();
    require_once('db_login.php');
    $con = mysqli_connect($db_host, $db_username, $db_password,
$db_database);

    $username = $_POST['username'];
    $password = $_POST['password'];
    $login = mysqli_query($con,"SELECT * FROM customers WHERE
email='$username' AND password='$password'");
    $cek = mysqli_num_rows($login);
    if($cek > 0){
        $data = mysqli_fetch_assoc($login);
        $_SESSION['username'] = $username;
        $_SESSION['password'] = $password;
        header("location:view_books.php");
    } else {
        header("location:login.php?pesan=gagal");
    }
?>

```

logout.php

```

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

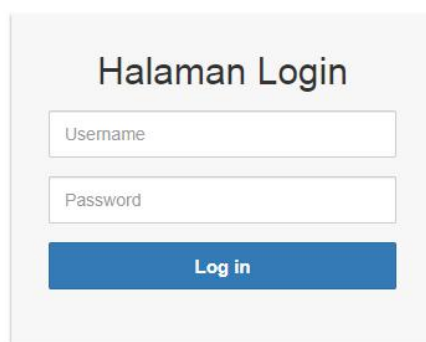
```

```
<h1>Anda berhasil Logut</h1>
<p>klik <a href="login.php">disini</a> untuk kembali ke halaman
login</p>
```

view_books.php

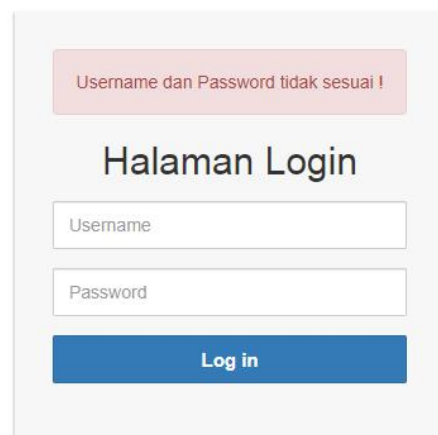
```
...
...
...
<body>
    <?php
        session_start();
        if($_SESSION['username'] == ""){
            header("location:login.php?pesan=blmlogin");
        }
    ?>
...
...
...
```

3) Screenshot



The screenshot shows a login form titled "Halaman Login". It contains two input fields: "Username" and "Password". Below the fields is a blue button labeled "Log in".

2.1 Login



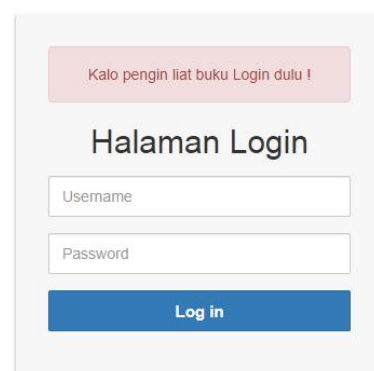
The screenshot shows the login form with a red error message at the top: "Username dan Password tidak sesuai !". The "Log in" button is still visible at the bottom.

2.2 Pesan Gagal

Anda berhasil Logut

klik [disini](#) untuk kembali ke halaman login

2.3 Logout



The screenshot shows the login form with a red message at the top: "Kalo pengen liat buku Login dulu !". The "Log in" button is still visible at the bottom.

2.4 Blm Login

4) Pembahasan

Pada file `login.php` akan terlihat seperti pada gambar **2.1 Login** diatas. Form akan diproses pada file `login_check.php`, yang mana jika pengguna memasukkan username dan password yang salah, diberikan kode `header("location:login.php?pesan=gagal");` yang akan membawa pengguna kembali ke halaman `login.php` dengan nilai `$_GET['pesan'] = gagal` yang mana akan menampilkan pesan seperti pada gambar **2.2 Pesan Gagal**.

Jika file `logout.php` diproses, maka akan terlihat pada gambar **2.3 Logout** dan disediakan link untuk kembali ke halaman `login.php`.

Kemudian untuk file `view_books.php` supaya hanya dapat dibuka oleh pengguna yang sudah login, kita tambahkan kode seperti pada file `view_books.php` di atas. Yang jika session bernilai "" alias kosong, akan memproses kode `header("location:login.php?pesan=blmlogin");` yang akan membawa pengguna ke halaman login dengan nilai `$_GET['pesan'] = blmlogin`. Tampilannya akan seperti pada gambar **2.4 Blm Login**.

C. Program untuk menampilkan detail data customer dari select item HTML

1) Soal

4. Latihan

Buatlah program untuk menampilkan detail data customer. Sebelumnya nama-nama customer ditampilkan dalam select item. Saat user memilih nama customer, maka program akan menampilkan detail data customer di bawahnya.

Customers Data

Julie Smith ▼

Name = Julie Smith
Address = 25 Oak Street
City = Airport West

2) Source Code

latihan.php

```
<!--
File : latihan.php
Deskripsi : menampilkan data customer
-->
<!DOCTYPE HTML>
<html>
  <head>
    <script>
      function getXMLHttpRequest() {
        var req = false;
        try {
          /* for Firefox */
          req = new XMLHttpRequest();
        } catch (err) {
          try {
            /* for some versions of IE */
            req = new ActiveXObject("Msxml2.XMLHTTP");
          } catch (err) {
            try {
              /* for some other versions of IE */
              req = new
ActiveXObject("Microsoft.XMLHTTP");
            } catch (err) {
              req = false;
            }
          }
        }
        return req;
      }
      function get_customers(sel){
        var xmlhttp = getXMLHttpRequest();
        //get input value
        var id = sel.value;
        //set url and inner
        var url = "latihan_get.php?id=" + id;
        var inner = "user_input";
        //open request
        xmlhttp.open('GET', url, true);
        xmlhttp.onreadystatechange = function() {
          if ((xmlhttp.readyState == 4) &&
(xmlhttp.status == 200)){
            document.getElementById(inner).innerHTML =
xmlhttp.responseText;
          }
          return false;
        }
        xmlhttp.send(null);
      }
    </script>
  </head>
  <body>
    <h2>Customers Data</h2>
    <select name="customers" id="customers"
onchange="get_customers(this);">
    <?php
```

```

// Include our login information
require_once('db_login.php');

// Connect
$con = mysqli_connect($db_host, $db_username,
$db_password,$db_database);
if (mysqli_connect_errno()){
    die ("Could not connect to the database: <br />".
mysqli_connect_error( ));
}

//Assign a query
$query = " SELECT * FROM customers ";

// Execute the query
$result = mysqli_query($con,$query);
if (!$result){
    die ("Could not query the database: <br />".
mysqli_error($con));
}
// Fetch and display the results
while ($row = mysqli_fetch_array($result)){
    echo
value="'. $row['customerid'].'">'. $row['name']. '</option>';
}
?>
</select>
<div id="user_input"></div>
</body>
</html>

```

latihan_get.php

```

<?php
// Include our login information
require_once('db_login.php');

// Connect
$con = mysqli_connect($db_host, $db_username,
$db_password,$db_database);
if (mysqli_connect_errno()){
    die ("Could not connect to the database: <br />".
mysqli_connect_error( ));
}

//Assign a query
$id = $_GET['id'];
$query = " SELECT * FROM customers WHERE customerid = '$id' ";

// Execute the query
$result = mysqli_query($con,$query);
if (!$result){
    die ("Could not query the database: <br />".
mysqli_error($con));
}
// Fetch and display the results
$row = mysqli_fetch_array($result);
echo "<br/><br/>";
echo 'Name = '. $row['name']. '<br />';

```

```
echo 'Address = '.$row['address'].'<br />';  
echo 'City = '.$row['city'].'<br />';  
?>
```

3) Screenshot

Customers Data

Linggar Maretva ▼

Name = Linggar Maretva
Address = Bandingan
City = Banjarnegara

4) Pembahasan

Pada file latihan.php, dibutuhkan koneksi ke database untuk mengambil nilai yang digunakan di form select HTML, kemudian pada form select tersebut, kita populate dengan nilai yang telah didapat dari koneksi ke database dengan kode `echo '<option value="'.$row['customerid'].'">'.$row['name'].'</option>';`. Yang akan membuat value dari tiap option adalah id customer, dan Nilai yang ditampilkan adalah Nama Customer.

Kemudian kita tambahkan juga di bawah form select kode `<div id="user_input"></div>` untuk diisi dengan file latihan_get.php yang diproses fungsi `get_customers(sel)` menggunakan AJAX. Fungsi `get_customers(sel)` ini akan menggunakan nilai id dari value option untuk dikirim ke file `latihan_get.php` dengan method GET. Kemudian pada file `latihan_get.php`, kita tinggal mengambil data customers dengan koneksi ke database seperti biasa, dan menampilkannya dengan kode echo seperti berikut.

```
echo "<br/><br/>";  
echo 'Name = '.$row['name'].'<br />';  
echo 'Address = '.$row['address'].'<br />';  
echo 'City = '.$row['city'].'<br />';
```

BAB IV

KESIMPULAN

A. Kesimpulan

Session adalah salah satu fasilitas yang ada pada PHP yang digunakan untuk menyimpan data sementara ke dalam variabel (variabel session) sehingga data tadi dapat di akses oleh client selama variabel session tadi tidak di kosongkan atau dihilangkan. Sehingga Session dapat menyimpan data seperti pada soal pertama dalam pembuatan fungsi `calculate_items($cart)`, `calculate_price($cart)`, dan `display_cart($cart)`.

Selain itu, nilai variabel di dalam session di simpan di sisi server (web server). Sehingga dapat digunakan untuk membuat fitur login, logout, dan pengecekan apakah sudah login atau belum, seperti pada soal kedua.

AJAX, merupakan metode suatu laman web menggunakan JavaScript untuk mengirim dan menerima data dari server tanpa harus menyegarkan (refresh) laman itu. Sehingga pada penerapannya pada soal ketiga, kita tidak perlu me-reload halaman untuk menampilkan data customer yang dipilih menggunakan form select. Data dapat ditampilkan secara langsung dengan memilihnya di form select tersebut.

DAFTAR PUSTAKA

- Informatics (25 Agustus 2018). "*Praktikum 6 - Cookies, Session, AJAX*". Makalah disajikan dalam Praktikum Pengembangan Berbasis Platform di laboratorium komputer gedung A Universitas Diponegoro. Semarang, 26 September 2019.
- Eko ae (30 Januari 2016). "*Pengertian Cookie Dan Session Berserta penggunaannya*". Retrieved 17 Oktober 2019. from EKOAE17 : <https://www.ekoe17.com/2016/01/pengertian-cookie-dan-session-berserta.html>
- Ekajogja (8 April 2014). "*MENGENAL JAVASCRIPT, AJAX, JQUERY, ANGULARJS, DAN NODE.JS*". Retrieved 17 Oktober 2019. from ARSIP EKAJOGJA : <https://ekajogja.com/arsip/mengenal-javascript-ajax-jquery-angularjs-dan-node-js/>