# K.S.R. COLLEGE OF ENGINEERING: TIRUCHENGODE - 637 215

## (Autonomous)

## DEPARTMENT OF COMPUTER SCIENCE AND ENIGNEERING

## LABORATORY MANUAL

## Sub. Code / Sub. Name: 18CS522 - WEB AND OPEN SOURCE LABORATORY

## YEAR / SEM : III / V

**Faculty in-charge** **HoD**

# LIST OF EXPERIMENTS

| EX.NO : 01 | Design a webpage for online shopping cart |
|---|---|
| DATE: | |

**Aim:**

To design a webpage for online shopping cart.

**PROGRAM:**

```
<html>
    <head>
        <title>Shopping cart </title>
        <style>
            .product-image {
              float: left;
              width: 20%;
            }
            .product-details {
              float: left;
              width: 37%;
            }
            .product-price {
              float: left;
              width: 12%;
            }
            .product-quantity {
              float: left;
              width: 10%;
```

```
}
.product-removal {
  float: left;
  width: 9%;
}
.product-line-price {
  float: left;
  width: 12%;
  text-align: right;
}
.group:before, .shopping-cart:before, .column-labels:before, .product:before, .totals-item:before,
.group:after,
.shopping-cart:after,
.column-labels:after,
.product:after,
.totals-item:after {
  content: '';
  display: table;
}
.group:after, .shopping-cart:after, .column-labels:after, .product:after, .totals-item:after {
  clear: both;
}
.group, .shopping-cart, .column-labels, .product, .totals-item {
  zoom: 1;
}
```

```
.product .product-price:before, .product .product-line-price:before, .totals-value:before {
  content: '$';
}
body {
  padding: 0px 30px 30px 20px;
  font-family: "HelveticaNeue-Light", "Helvetica Neue Light", "Helvetica Neue", Helvetica, Arial, sans-serif;
  font-weight: 100;
}
h1 {
  font-weight: 100;
}
label {
  color: #aaa;
}
.shopping-cart {
  margin-top: -45px;
}
.column-labels label {
  padding-bottom: 15px;
  margin-bottom: 15px;
  border-bottom: 1px solid #eee;
}
.column-labels .product-image, .column-labels .product-details, .column-labels .product-removal {
  text-indent: -9999px;
}
```

```
.product {
  margin-bottom: 20px;
  padding-bottom: 10px;
  border-bottom: 1px solid #eee;
}
.product .product-image {
  text-align: center;
}
.product .product-image img {
  width: 100px;
}
.product .product-details .product-title {
  margin-right: 20px;
  font-family: "HelveticaNeue-Medium", "Helvetica Neue Medium";
}
.product .product-details .product-description {
  margin: 5px 20px 5px 0;
  line-height: 1.4em;
}
.product .product-quantity input {
  width: 40px;
}
.product .remove-product {
  border: 0;
  padding: 4px 8px;
  background-color: #c66;
```

```
    color: #fff;
    font-family: "HelveticaNeue-Medium", "Helvetica Neue Medium";
    font-size: 12px;
    border-radius: 3px;
  }
  .product .remove-product:hover {
    background-color: #a44;
  }
  .totals .totals-item {
    float: right;
    clear: both;
    width: 100%;
    margin-bottom: 10px;
  }
  .totals .totals-item label {
    float: left;
    clear: both;
    width: 79%;
    text-align: right;
  }
  .totals .totals-item .totals-value {
    float: right;
    width: 21%;
    text-align: right;
  }
  .totals .totals-item-total {
```

```
    font-family: "HelveticaNeue-Medium", "Helvetica Neue Medium";
  }
  .checkout {
    float: right;
    border: 0;
    margin-top: 20px;
    padding: 6px 25px;
    background-color: #6b6;
    color: #fff;
    font-size: 25px;
    border-radius: 3px;
  }
  .checkout:hover {
    background-color: #494;
  }
  @media screen and (max-width: 650px) {
    .shopping-cart {
      margin: 0;
      padding-top: 20px;
      border-top: 1px solid #eee;
    }
    .column-labels {
      display: none;
    }
    .product-image {
      float: right;
```

```
  width: auto;
}
.product-image img {
  margin: 0 0 10px 10px;
}
.product-details {
  float: none;
  margin-bottom: 10px;
  width: auto;
}
.product-price {
  clear: both;
  width: 70px;
}
.product-quantity {
  width: 100px;
}
.product-quantity input {
  margin-left: 20px;
}
.product-quantity:before {
  content: 'x';
}
.product-removal {
  width: auto;
}
.product-line-price {
```

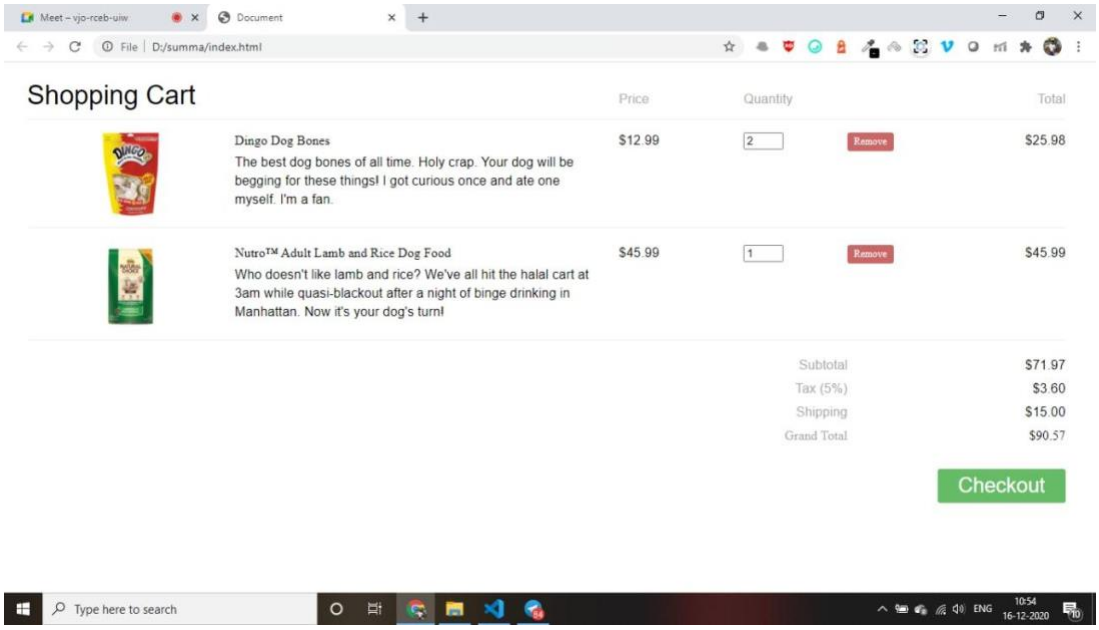
```
            float: right;
            width: 70px;
          }
        }
        @media screen and (max-width: 350px) {
          .product-removal {
            float: right;
          }
          .product-line-price {
            float: right;
            clear: left;
            width: auto;
            margin-top: 10px;
          }
          .product .product-line-price:before {
            content: 'Item Total: $';
          }
          .totals .totals-item label {
            width: 60%;
          }
          .totals .totals-item .totals-value {
            width: 40%;
          }
        }
    </style>
</head>
<body>
```

```
<h1>Shopping Cart</h1>
<div class="shopping-cart">
<div class="column-labels">
<label class="product-image">Image</label>
<label class="product-details">Product</label>
<label class="product-price">Price</label>
<label class="product-quantity">Quantity</label>
<label class="product-removal">Remove</label>
<label class="product-line-price">Total</label>
</div>
<div class="product">
<div class="product-image">
<img src="https://s.cdpn.io/3/dingo-dog-bones.jpg">
</div>
<div class="product-details">
<div class="product-title">Dingo Dog Bones</div>
<p class="product-description">The best dog bones of all time. Holy crap. Your dog will be begging for these things! I got curious once and ate one myself. I'm a fan.</p>
</div>
<div class="product-price">12.99</div>
<div class="product-quantity">
<input type="number" value="2" min="1">
</div>
<div class="product-removal">
<button class="remove-product">
    Remove
</button>
```

```
</div>

<div class="product-line-price">25.98</div>

</div>


<div class="product">

<div class="product-image">

<img src="https://s.cdpn.io/3/large-NutroNaturalChoiceAdultLambMealandRiceDryDogFood.png">

</div>

<div class="product-details">

<div class="product-title">Nutro™ Adult Lamb and Rice Dog Food</div>

<p class="product-description">Who doesn't like lamb and rice? We've all hit the halal cart at 3am while quasi-blackout after a night of binge drinking in Manhattan. Now it's your dog's turn!</p>

</div>

<div class="product-price">45.99</div>

<div class="product-quantity">

<input type="number" value="1" min="1">

</div>

<div class="product-removal">

<button class="remove-product">

    Remove

</button>

</div>

<div class="product-line-price">45.99</div>

</div>


<div class="totals">
```

```
                <div class="totals-item">
                <label>Subtotal</label>
                <div class="totals-value" id="cart-subtotal">71.97</div>
                </div>
                <div class="totals-item">
                <label>Tax (5%)</label>
                <div class="totals-value" id="cart-tax">3.60</div>
                </div>
                <div class="totals-item">
                <label>Shipping</label>
                <div class="totals-value" id="cart-shipping">15.00</div>
                </div>
                <div class="totals-item totals-item-total">
                <label>Grand Total</label>
                <div class="totals-value" id="cart-total">90.57</div>
                </div>
                </div>

                <button class="checkout">Checkout</button>
                </div>
        </body>
</html>
```

OUTPUT:




Result:

Thus the webpage for online shopping cart was completed successfully.

| Ex.No:02 | Design a Webpage for multimedia library with CSS and bootstrap |
|---|---|
| Date: | |

**Aim:**

To Design a Webpage for multimedia library with CSS and bootstrap

**Index.html:**

```
<link href="//netdna.bootstrapcdn.com/bootstrap/3.1.0/css/bootstrap.min.css" rel="stylesheet" id="bootstrap-css">

<script src="//netdna.bootstrapcdn.com/bootstrap/3.1.0/js/bootstrap.min.js"></script>

<script src="//code.jquery.com/jquery-1.11.1.min.js"></script>

<!------ Include the above in your HEAD tag ---------->



<!--

Images Credits:

https://www.iconfinder.com/icons/44703/audio_speakers_icon

https://www.iconfinder.com/icons/71615/tv_icon

-->

<div class="container">

<div class="row">


<h2>Mr.M.'s media control buttons</h2>
```

```
<div class="col-md-6 col-md-offset-3">


<div class="tcenter">

<img
src="https://cdn1.iconfinder.com/data/icons/IMPRESSIONS/multimedia/png/128/audio_speakers.png" alt="">

<img src="https://cdn1.iconfinder.com/data/icons/MediaPack_ICON/256/TV.png" alt="">

<img
src="https://cdn1.iconfinder.com/data/icons/IMPRESSIONS/multimedia/png/128/audio_speakers.png" alt="">

</div><!-- /.tcenter -->


<div class="tcenter">

<a href="#" class="btn btn-default"><i class="glyphicon glyphicon-fast-backward"></i></a>

<a href="#" class="btn btn-default"><i class="glyphicon glyphicon-backward"></i></a>

<a href="#" class="btn btn-default"><i class="glyphicon glyphicon-stop"></i></a>

<a href="#" class="btn btn-default"><i class="glyphicon glyphicon-play"></i></a>

<a href="#" class="btn btn-default"><i class="glyphicon glyphicon-pause"></i></a>

<a href="#" class="btn btn-default"><i class="glyphicon glyphicon-forward"></i></a>

<a href="#" class="btn btn-default"><i class="glyphicon glyphicon-fast-forward"></i></a>

</div><!-- /.tcenter -->


<div class="tcenter">

<a href="#" class="btn btn-default"><i class="glyphicon glyphicon-volume-down"></i></a>

<a href="#" class="btn btn-default"><i class="glyphicon glyphicon-volume-up"></i></a>

<a href="#" class="btn btn-default"><i class="glyphicon glyphicon-volume-off"></i></a>

<hr>

<a href="http://validator.w3.org/check?uri=http://bootsnipp.com/iframe/V1WP"><span
class="glyphicon glyphicon-check green"></span> HTML<sup>5</sup></a>
```

```
</div><!-- /.tcenter -->

</div><!-- /.col-mid -->

</div><!-- /.row -->

</div><!-- /.container -->
```

Style.css:

```
.tcenter{
padding: 10px;
text-align:center;
}

/* for validator */
a:link {
text-decoration:none;
}

.green, .green a {
color: #339900;
}
```

**Output:**

Multimedia library in boostrap

**Result:**

Thus web page for multimedia library with CSS and Bootstrap is created sucessfully.

| EX.NO:03 | **Online registration forms with JavaScript validation.** |
|---|---|
| DATE: | |

**Aim:**

To Create the html prage for user interactive and Use java script to perform validation

**PROGRAM:**

```
<html>
<head>
<script>
function valid()  {
   var name = document.forms["RegForm"]["Name"];
   var email = document.forms["RegForm"]["EMail"];
   var phone = document.forms["RegForm"]["Telephone"];
if (name.value=="") {
      window.alert("Please enter your name.");
      name.focus();
      return false;
   }
   else if (email.value == "") {
```

```
        window.alert("Please enter a valid e-mail address.");
        email.focus();
        return false;
    }
    else if (email.value.indexOf("@", 0) < 0)    {
       window.alert("Please enter a valid e-mail address.");
        email.focus();
        return false;
    }
else if (phone.value == "") {
        window.alert("Please enter your telephone number.");
        phone.focus();
        return false;
    }
location.reload(true);
}</script>
<style>
h1{
color:grey;
font-family:Comic Sans MS;
}
p {
```

```
text-align:center;

color:blue;

}

</style>

</head>

<body>

<h1 style="text-align: center"> Customer care</h1>

<form name="RegForm"   method="get" >

<p>Name: <input type="text" size="45" name="Name" placeholder="enter the name"></p><br>


<p>E-mail Address: <input type="text" size="35" name="EMail" placeholder="enter email address"></p><br>

<p>Telephone: <input type="text" size="40" name="Telephone" placeholder="enter phone number"></p><br>

<p>Comments:  <textarea cols="35" name="Comment"></textarea></p>

<p><input type="button" value="send" name="Submit" onclick="return valid()"></p>

</form>

</body>

</html>
```

**OUTPUT:**


Result:

Thus Online registration form with JavaScript validation was created sucessfully.

| EX.NO: 04 | Develop a Web Application Using JDBC or MONGODB |
|---|---|
| Date: | |

**Aim:**

To Develop a Web Application Using JDBC or MONGODB.

**PROGRAM:**

## Procedure

**To start with interfacing Java Servlet Program with JDBC Connection:**

1. Proper JDBC Environment should set-up along with database creation.
2. To do so, download the mysql-connector.jar file from the internet,
3. As it is downloaded, move the jar file to the apache-tomcat server folder,
4. Place the file in lib folder present in the apache-tomcat directory.
5. To start with the basic concept of interfacing:

**Step 1: Creation of Database and Table in MySQL**

As soon as jar file is placed in the folder, create a database and table in MySQL,

```
mysql> create database demoprj;

Query OK, 1 row affected (4.10 sec)

mysql> use demoprj

Database changed

mysql> create table demo(id int(10), string varchar(20));

Query OK, 0 rows affected (1.93 sec)

mysql> desc demo;
```

```
+--------+-------------+------+-----+---------+-------+
| Field  | Type        | Null | Key | Default | Extra |
+--------+-------------+------+-----+---------+-------+
| id     | int(10)     | YES  |     | NULL    |       |
| string | varchar(20) | YES  |     | NULL    |       |
+--------+-------------+------+-----+---------+-------+
2 rows in set (0.40 sec)
```

**Step 2: Implementation of required Web-pages**
**Create a form in HTML file, where take all the inputs required to insert data into the database.**

```
<!DOCTYPE html>
<html>
<head>
<title>Insert Data</title>
</head>
<body>
<!-- Give Servlet reference to the form as an instances
GET and POST services can be according to the problem statement-->
<form action="./InsertData" method="post">
<p>ID:</p>
<!-- Create an element with mandatory name attribute,
so that data can be transfer to the servlet using getParameter() -->
<input type="text" name="id"/>
<br/>
<p>String:</p>
```

```
<input type="text" name="string"/>

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

<input type="submit"/>

</form>

</body>

</html>
```

**Step 3: Creation of Java Servlet program with JDBC Connection**

To create a JDBC Connection steps are

1. Import all the packages
2. Register the JDBC Driver
3. Open a connection
4. Execute the query, and retrieve the result
5. Clean up the JDBC Environment

```
import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.SQLException;

// This class can be used to initialize the database connection

public class DatabaseConnection {

protected static Connection initializeDatabase()

throws SQLException, ClassNotFoundException

{

// Initialize all the information regarding

// Database Connection

String dbDriver = "com.mysql.jdbc.Driver";

String dbURL = "jdbc:mysql:// localhost:3306/";

// Database name to access
```

```
String dbName = "demoprj";
String dbUsername = "root";
String dbPassword = "root";
Class.forName(dbDriver);
Connection con = DriverManager.getConnection(dbURL + dbName, dbUsername,     dbPassword);
return con;
}
}
```

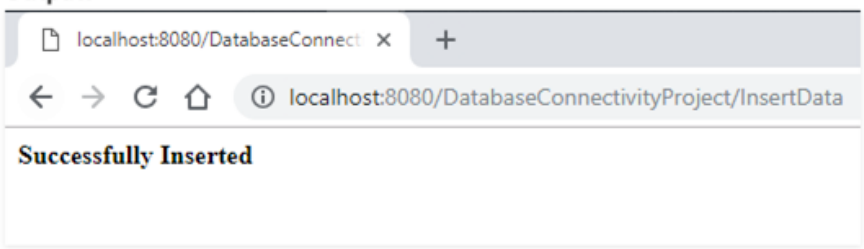
### Step 4: To use this class method, create an object in Java Servlet program

```
import java.io.IOException;
import java.io.PrintWriter;
import java.sql.Connection;
import java.sql.PreparedStatement;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
// Import Database Connection Class file
import code.DatabaseConnection;
// Servlet Name
@WebServlet("/InsertData")
public class InsertData extends HttpServlet {
private static final long serialVersionUID = 1L;
protected void doPost(HttpServletRequest request,
```

```
HttpServletResponse response)
throws ServletException, IOException
{
try {

// Initialize the database
Connection con = DatabaseConnection.initializeDatabase();
// Create a SQL query to insert data into demo table
// demo table consists of two columns, so two '?' is used
PreparedStatement st = con
.prepareStatement("insert into demo values(?, ?)");
// For the first parameter,
// get the data using request object
// sets the data to st pointer
st.setInt(1, Integer.valueOf(request.getParameter("id")));
// Same for second parameter
st.setString(2, request.getParameter("string"));
// Execute the insert command using executeUpdate()
// to make changes in database
st.executeUpdate();
// Close all the connections
st.close();
con.close();
// Get a writer pointer
// to display the successful result
```

```
PrintWriter out = response.getWriter();

out.println("<html><body><b>Successfully Inserted"

+ "</b></body></html>");

}

catch (Exception e) {

e.printStackTrace();

}

}

}
```

**Output:**




**Result in MySQL Interface**

```
mysql> select * from demo;
+------+---------------+
| id   | string        |
+------+---------------+
|    1 | Kumar         |
+------+---------------+
1 row in set (0.06 sec)
```

Result

Thus the web page using JDBC was created successfully

| EX.NO:05 | Develop a web application using database with AJAX |
|---|---|
| DATE: | |

**Aim:**

To construct a web application using database with AJAX

**PROGRAM:**

**table1.html**

```
<html>

<head>

<script>

var request;

function sendInfo()

{

var v=document.vinform.t1.value;

var url="index.jsp?val="+v;


if(window.XMLHttpRequest){

request=new XMLHttpRequest();

}

else if(window.ActiveXObject){

request=new ActiveXObject("Microsoft.XMLHTTP");

}
```

```
try{
request.onreadystatechange=getInfo;
request.open("GET",url,true);
request.send();
}catch(e){alert("Unable to connect to server");}
}

function getInfo(){
if(request.readyState==4){
var val=request.responseText;
document.getElementById('amit').innerHTML=val;
}
}

</script>
</head>
<body>
<marquee><h1>This is an example of ajax</h1></marquee>
<form name="vinform">
Enter id:<input type="text" name="t1" onkeyup="sendInfo()">
</form>
```

```
<span id="amit"></span>


</body>

</html>
```

create server side page to process the request

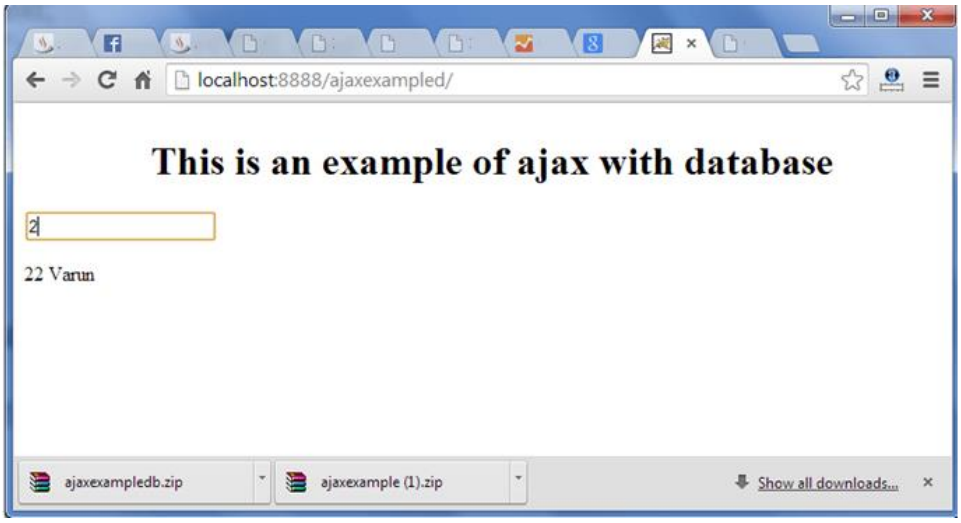
In this jsp page, we printing the id and name of the employee for the given id.

## index.jsp

```
<%@ page import="java.sql.*"%>


<%

String s=request.getParameter("val");

if(s==null || s.trim().equals("")){

out.print("Please enter id");

}else{

int id=Integer.parseInt(s);

out.print(id);

try{

Class.forName("com.mysql.jdbc.Driver");

Connection con=DriverManager.getConnection("jdbc:mysql://localhost:3306/mdb","root","root");
```

```
PreparedStatement ps=con.prepareStatement("select * from emp where id=?");

ps.setInt(1,id);

ResultSet rs=ps.executeQuery();

while(rs.next()){

out.print(rs.getInt(1)+" "+rs.getString(2));

}

con.close();

}catch(Exception e){e.printStackTrace();}

}

%>
```

## Output


localhost:8888/ajaxexampled/
This is an example of ajax with database
2
22 Varun
ajaxexampledb.zip
ajaxexample (1).zip
Show all downloads...


localhost:8888/ajaxexampled/
This is an example of ajax with database
3
33 Jai
ajaxexampledb.zip
ajaxexample (1).zip
Show all downloads...

**Result**

Thus the web application using AJAX with database connectivity was created successfully.

| Ex.No: 06 | PHP PROGRAMS |
| --- | --- |
| Date: | |

**AIM:**

Develop a PHP program to implement the following,

a.Variables-Constants-Data Types

b.Operators-Statements

c.Functions-Arrays

**PHP Datatypes:**

**Example:**

```
<?php
echo PHP_INT_MAX;

?>
```

**Output:**

```
9223372036854775807
```

# PHP Variable:

**Example:**

```
<?php
$my_var = 1;
echo $my_var;
?>
```

**Output:**

```
1
```

**PHP if statement:**

Example:

```
<?php
```

```
$t = date("H");
if ($t < "20") {
  echo "Have a good day!";
}
?>
```

**Output:**

Have a good day!

**PHP if…else statement:**

```
<?php
$t = date("H");

if ($t < "20") {
  echo "Have a good day!";
} else {
  echo "Have a good night!";
}
?>
```

**OUTPUT:**

Have a good day!

PHP if…elseif…else statement

```
<?php
$t = date("H");
echo "<p>The hour (of the server) is " . $t;
echo ", and will give the following message:</p>";

if ($t < "10") {
```

```
  echo "Have a good morning!";
} elseif ($t < "20") {
  echo "Have a good day!";
} else {
  echo "Have a good night!";
}
?>
```

**Output:**

The hour (of the server) is 10, and will give the following message: Have a good day!

**PHP FUNCTIONS:**

```
<?php
    function writeMsg() {
      echo "Hello world!";
    }

    writeMsg();
?>
```

**OUTPUT:**

Hello world!

**PHP Arrays:**

An array is a special variable, which can hold more than one value at a time.

Example:

```
<?php
        $cars = array("Volvo", "BMW", "Toyota");
        echo "I like " . $cars[0] . ", " . $cars[1] . " and " . $cars[2] . ".";
?>
```

**Output:**

I like Volvo, BMW and Toyota.

**Result:**

Thus the PHP program was executed successfully.

| EX.NO:07 | PHP program to connect MySQL DB and retrieve a record in HTML table |
|---|---|
| DATE: | |

**Aim:**

To construct a PHP program to connect MySQL DB and retrieve a record in HTML table.

**PROGRAM:**

```
//UserData.html<?xml version="1.0" encoding="ISO-8859-1"?>
<html xmlns="http://www.w3.org/1999/xhtml">
<body>
<h3> Program to collect the customer-information </h3>
<form action ="Display.php" method="get">
<table border="5">
<tr><td> Enter Name:</td>
<td><input type="text" name="name"></td>
</tr><tr><td> Enter Address Line1: </td>
<td><input type="text" name="address1"></td>
</tr>
<tr><td> Enter Address line2: </td
<td><input type="text" name="address2"></td>
</tr>
<tr><td> Enter Email-id: </td>
<td><input type="text" name="email"></td>
</tr><tr><td></td></tr><tr></tr><tr><td>
<input type="submit" value="Submit"></td>
<td><input type="Reset" value="Reset"></td></tr></br>
<a href="<a href="https://krazytech.com/programs/simple-library-management-system-
php-mysql" target="_blank" rel="noopener">Search.html</a>">
To search click here </a></td></table>
</form>
</body>
```

```
</html>
```

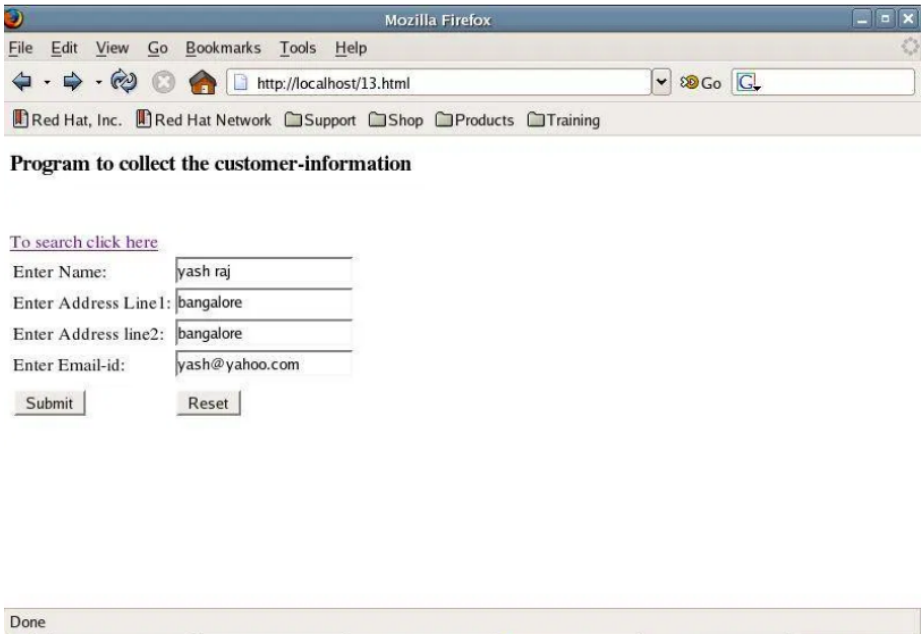
**PHP CODE:**

```
//Display.php<html>
<head><title> Display.php </title>
</head><body bgcolor="aabbcc">
<?php$name1=$_REQUEST["name"];
$address1=$_REQUEST["address1"];
$address2=$_REQUEST["address2"];
$email=$_REQUEST["email"];
define('DB_SERVER', 'localhost:3306');
define('DB_USERNAME', 'root');
define('DB_PASSWORD', 'root123');
define('DB_DATABASE', 'customers');
 //where customers is the database$db =
mysqli_connect(DB_SERVER,DB_USERNAME,DB_PASSWORD,DB_DATABASE);
$query= "insert into address values('$name1','$address1','$address2','$email');
 //to insert input records into a table - address$enter=
mysqli_query($db,$query);$query="select * from address";
// Fetch all the records from the table address$result=mysqli_query($db,$query);
?>
<h3> Page to display the stored data </h3>
<table border="1">
<tr><th> NAME </th><th> ADDRESS Line1 </th>
<th> ADDRESS Line2 </th><th> EMAIL-id </th>
</tr>
<?php while($array=mysqli_fetch_row($result)) ?>
<tr><td><?echo $array[0];?></td>
<td><?echo $array[1];?></td>
<td><?echo $array[2];?></td><td><?echo $array[3];?></td>
</tr><?php endwhile; ?>
<?php mysqli_free_result($result); ?>
<?php mysqli_close($db); ?></table>
```

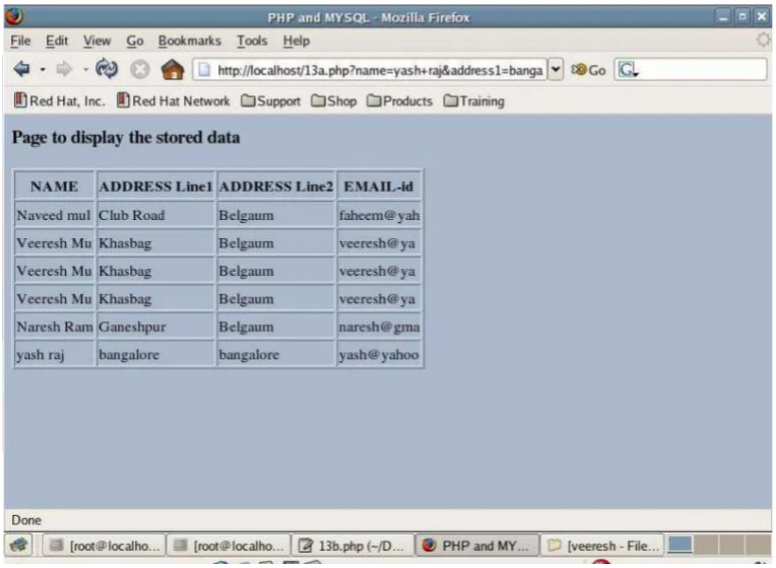
```
</body>
</html>
```

**Output:**

**The following image shows how the UserData.html looks on the browser:**



**The following image shows how the Display.php looks on the browser:**




**Result:**

PHP program to connect MySQL database and retrieve a record in HTML table was created successfully.

| EX.NO: 08 a) | PERL PROGRAMS Variables and Data types - Statements and Control Structures |
|---|---|
| DATE: | |

**Aim:**

To create a PERL programs using Variables and Data types - Statements and Control Structures

**Programs**

```
#!usr/bin/perl   # the above line is shebang directive


$name=<STDIN>;

chomp($name);

print "$name\n";
```

**output:**

```
[linuxpert@localhost ~]$ perl first.pl

gokul

gokul
```

**2. Array in PERL**

```
#!usr/bin/perl

my @animal=("cow","Buffalo","Camel");

print "@animal\n"; # list all elements in array

print "$#animal\n"; # list last element position

print "$animal[0]\n"; #list 0th position element

$count=@animal;
```

```
print "$count"; # count no of elements in array
```

**output:**

```
[linuxpert@localhost ~]$ perl array.pl
cow Buffalo Camel
2cow
3
```

**3.If loop in perl**

```
#!usr/bin/perl
my $a=10;
$condition=1;
if($condition)
{
my $y=100;
print "$a\n";
print "$y\n";
}
print "$a\n";
print "$y\n";
```

**output:**

```
[linuxpert@localhost ~]$ perl ifloop.pl
10
```

100

10

**4. While loop (until) in perl**

```
#!usr/bin/perl

$a=0;

until($a>10) #is equal to while

{

print "$a\n";

$a++;

}
```

**out put:**

```
[linuxpert@localhost ~]$ perl unless.pl

a less than 10[linuxpert@localhost ~]$ perl until.pl

0 1 2 3 4 5 6 7 8  9 10
```

**5. for each loop (upper limit is not fixed )in perl**

```
#!usr/bin/perl

my @animals=("cow","buffalo","camel",123,100,243,300);

foreach $key(@animals)

{

print "$key\n";

}
```

**output:**

[linuxpert@localhost ~]$ perl foreach.pl

cow

buffalo

camel

123

100

243

300

**6. String operation:**

#!usr/bin/perl

$a="hello";

$b="world";

print $a.$b,"\n";

$str="-";

print $str x 80,"/n";

@a=(10..25);

print "@a\n";

**output**

[linuxpert@localhost ~]$ perl string.pl

helloworld

--------------------------------------------------------------------------------/n10 11

12 13 14 15 16 17 18 19 20 21 22 23 24 25

**Result:**

Thus Perl Programs using Variables and Data types - Statements and Control Structures was executed successfully

| EX.NO: 08 b) | PERL PROGRAMS Subroutines - Packages |
| --- | --- |
| DATE: | |

**Aim:**

To create a PERL programs using Subroutines - Packages

## Subroutines

```
#!usr/bin/perl

sub sayHello()

{

print "Hello\n";

}

&sayHello();
```

**output:**

```
[linuxpert@localhost ~]$ perl function.pl

Hello
```

**Packages**

```
#Greatest among 3 number

Live Demo

#!/usr/bin/perl
```

```
package Foo;

print "Begin and Block Demo\n";


BEGIN {

  print "This is BEGIN Block\n"

}


END {

  print "This is END Block\n"

}


1;
```

**output:**

```
This is BEGIN Block

Begin and Block Demo

This is END Block
```

**Result:**

Thus Perl Programs for subroutines and packages was executed successfully

| EX.NO:09 | Login form using perl and perform a validation |
|---|---|
| DATE: | |

**Aim:**

To develop a login form using perl and perform a validation.

**PROGRAM:**

**HTML code:**

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<htmlxmlns="http://www.w3.org/1999/xhtml">
<head>
<metahttp-equiv="Content-Type"content="text/html; charset=utf-8"/>
<title>Very simple login using Perl, jQuery, Ajax, JSON and MySQL</title>
<linkrel="stylesheet"type="text/css"media="screen, projection"
href="http://www.blueprintcss.org/blueprint/screen.css"/>
<linkrel="stylesheet"type="text/css"media="screen, projection"
href="http://www.blueprintcss.org/blueprint/plugins/buttons/screen.css"/>
<linkrel="stylesheet"type="text/css"media="print"
href="http://www.blueprintcss.org/blueprint/print.css"/>
<!--[if IE]><link rel="stylesheet" type="text/css" media="screen, projection"
href="http://www.blueprintcss.org/blueprint/ie.css"><![endif]-->
<scripttype="text/javascript"
src="//code.jquery.com/jquery-1.4.4.min.js"></script>
<script type="text/javascript" src="login.js"></script>
<styletype="text/css">
#loginContent { width: 350px; margin: 100px auto; }
```

```
button[type] { margin: 0.5em 0; }
</style>
</head>
<body>
<divid="loginContent"class="container">
<divid="loginResult"style="display:none;">
</div>
<formid="loginForm"name="loginForm"method="post"action="">
<fieldset>
<legend>Enter information</legend>
<p>
<labelfor="username">Username</label>
<br/>
<inputtype="text"id="username"name="username"class="text"size="20"/>
</p>
<p>
<labelfor="password">Password</label>
<br/>
<inputtype="password"id="password"name="password"class="text"size="20"/>
</p>
<p>
<buttontype="submit"class="button positive">
<imgalt="ok"src=
"http://www.blueprintcss.org/blueprint/plugins/buttons/icons/tick.png"/>
Login
</button>
</p>
</fieldset>
</form>
</div>
```

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

**JS code:**

```
$(document).ready(function(){
$("form#loginForm").submit(function() { // loginForm is submitted
var username = $('#username').attr('value'); // get username
var password = $('#password').attr('value'); // get password
if (username && password) { // values are not empty
$.ajax({
type: "GET",
url: "/cgi-bin/login.pl", // URL of the Perl script
contentType: "application/json; charset=utf-8",
dataType: "json",
// send username and password as parameters to the Perl script
data: "username=" + username + "&password=" + password,
// script call was *not* successful
error: function(XMLHttpRequest, textStatus, errorThrown) {
$('div#loginResult').text("responseText: " + XMLHttpRequest.responseText
+ ", textStatus: " + textStatus
+ ", errorThrown: " + errorThrown);
$('div#loginResult').addClass("error");
}, // error
// script call was successful
// data contains the JSON values returned by the Perl script
success: function(data){
if (data.error) { // script returned error
$('div#loginResult').text("data.error: " + data.error);
```

```
$('div#loginResult').addClass("error");
} // if
else { // login was successful
$('form#loginForm').hide();
$('div#loginResult').text("data.success: " + data.success
+ ", data.userid: " + data.userid);
$('div#loginResult').addClass("success");
} //else
} // success
}); // ajax
} // if
else {
$('div#loginResult').text("enter username and password");
$('div#loginResult').addClass("error");
} // else
$('div#loginResult').fadeIn();
return false;
});
});
```

**PERL code:**

```
#!/usr/bin/perl -T
use CGI;
use DBI;
use strict;
use warnings;
# read the CGI params
my $cgi = CGI->new;
my $username = $cgi->param("username");
```

```
my $password = $cgi->param("password");

# connect to the database
my $dbh = DBI->connect("DBI:mysql:database=;host=;port=",
"", "")
or die $DBI::errstr;

# check the username and password in the database
my $statement = qq{SELECT id FROM users WHERE username=? and password=?};
my $sth = $dbh->prepare($statement)
or die $dbh->errstr;
$sth->execute($username, $password)
or die $sth->errstr;
my ($userID) = $sth->fetchrow_array;

# create a JSON string according to the database result
my $json = ($userID) ?
qq{{"success" : "login is successful", "userid" : "$userID"}} :
qq{{"error" : "username or password is wrong"}};

# return JSON string
print $cgi->header(-type => "application/json", -charset => "utf-8");
print $json;
```

OUTPUT :

{"success" : "login is successful", "userid" : "1"}

Result:

Thus a Login form using Perl and perform a validation was created successfully.

| EX.NO:10 | Perl program to connect MySQL DB & retrieve data. |
|---|---|
| DATE: | |

**Aim:**

To create a PERL program to connect MySQL DB & retrieve data.

**PROGRAM:**

```
$ vi connect.pl
#!/usr/bin/perl
use DBI;
$source = "DBI:mysql:mycompany:localhost";
$username = "root";
$password = "MySecretPassword";
$dbc = DBI->connect($source, $username, $password)
or die "Unable to connect to mysql: $DBI::errstr\n";
$sql = $dbc->prepare("select id, name from employee");
$out = $sql->execute()or die "Unable to execute sql: $sql->errstr";
while (($id, $name) = $sql->fetchrow_array()){
        print "Id: $id Name: $name\n";
}
```

**OUTPUT :**

**$ ./connect.plId: 100 Name: ThomasId: 200 Name: JasonId: 300 Name: SanjayId: 400 Name: NishaId: 500 Name: RandyId: 501 Name: Ritu**

**Result:**

Thus Perl program was connected to MYSQL and the data is retrieved.