

Assignment 2

Q 1. How to integrate web forms and database?

Explain with example

→ The easiest way to connect web forms to SQL Database use our simple database connector to securely integrate your web form to database. Easily create dynamic picklist and master details forms, perform lookups, and invoke stored procedure against any SQL database. Deliver engaging dynamic web experience to customers and partners without programming.

Data forms and workflow in 3 easy steps.

1. Define SQL query :- Configure the connector you can use a standard SQL query or a stored procedure.

2. Generate the form :- Add a data source to the form and generate controls corresponding to the SQL query.

3. Link to the query :- Configure the form or workflow so that it finds links to the query to read or write to the database.

ex:-

Code:-

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<title> Form site </title>
```

```
</head>
```

```
<body>
```

```
<form method = "POST" action = "connect.php" >
```

```
Username :- < Input type = "Password" name = "password" >
```

```
<br>
```

```
Password :- < Input type = "Password" name = "password" >
```

```
<br>
```

```
< Input type = "submit" value = "submit" >
```

```
</form>
```

```
</body>
```

```
</html>
```

Output:-

Username :-

Password :-

Submit

Date: _____
Page: _____

Your HTML form looks like this

PHP code (connect.php)

```
<?php
$username = filter_input(INPUT_POST, "username");

$password = filter_input(INPUT_POST, "password");

if (!empty($username)) {
    if (!empty($password)) {
        $host = "localhost";
        $dbusername = "root";
        $dbpassword = "";
        $dbname = "youtube";

        // to create connection
        $con = mysqli($host, $dbusername, $dbpassword,
            $dbname);

        if (mysqli_connect_error()) {
            die('connect_error(' . mysqli_connect_error()
                . ' . mysqli_connect_error());
        }
    }
}
```



```
else {  
    $sql = "INSERT INTO account (  
        username, password) values ('$username', 'password');  
  
    if ($con → query($sql)) {  
        echo "new record is inserted successfully",  
    }  
}
```

```
else {  
    echo "error:-" $sql . " " $con → error,  
}
```

```
$con → close();  
}
```

```
}  
else {  
    echo "password should not be empty";  
    die();  
}  
}
```

```
else {  
    echo "username should not be empty";  
    die();  
}
```

```
?>
```


Q2 How to work with authentication among users?

→ In order to gain access, users must prove to the website that are who they say they are. The I.D and key are enough to confirm the users identity; which will allow the system to authorize the user.

It's important to note that authorization on the other hand, is what dictates what users are able to see and do when they log in while authentication and authorization are often used interchangeably, the two different terms work together to create a secure log in process.

To put it simply, user authentication has three task,

1. Manage the connection between the human (user) and the website's

servers (computers)

2. Verify user identities

3. Approve (or decline) the authorizing authentication so the system can move to authorizing the users.

The process is fairly simple: users input their credential on the website login form. That information is then sent to the authentication server where the information is compared with all the users credentials on file.

Q3 Explain session with example

→ When you work with an application you open it, do some changes, and then close it. This is much like a session. The computer knows who you are. It knows when you start the application and when you end. But on the internet there is one problem the web server does not know who you are or what you do, because the HTTP address doesn't maintain state.

Session variable solves this problem by storing user information to be used across multiple pages (e.g. username, favourite color, etc). By default, session variable last until the user closes the browser.

So, session variables hold information about one single user, and are available to all pages in one application.

A session is started with the session start() function.

Date: _____
Page: _____

A variable are set with the PHP global variable: `$_SESSION`

Now, let's create a new page called "demo-session1.php". In this page we start a new php session and set some session variables:

Example

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

```
<!DOCTYPE html>  
<html>  
  <head> </head>  
  <body>  
    <?php>
```

```
    // set session variables
```

```
    $_SESSION["favcolor"] = "green";
```

```
    $_SESSION["favanimal"] = "cat";
```

```
    echo "session variables are set.";  
?>
```

```
</body>
```

```
</html>
```


Date: _____
Page: _____

Note :- The `session_start()` function must be the very first thing in your document. Before any html tags.

Q4 Explain cookies with example :

→ A cookie is often used to identify a user. A cookie is a small file that the server embeds on the user's computer. Each time the same computer requests a page with a browser, it will send the cookie too. With php you can both create and retrieve cookie value.

Create cookies with php

A cookie is created with the `setcookie()` function

Php create / retrieve a cookie

The following example create a cookie named "user" with the value "John Doe".

The cookie will expire after 30 day ($86400 * 30$) the "/" mean that cookie is available in entire website (otherwise, select the directory you prefer)

Example

<?php

\$cookie_name = "user";

\$cookie_value = "John Doe";

setcookie(\$cookie_name, \$cookie_value,
time() + (86400 * 30), "/");

?>

<html>

<body>

<?php

if (!isset(\$_COOKIE[\$cookie_name]))

{

echo "cookie named";

\$cookie_name " is not set!";

} else {

echo "cookie " . \$cookie_name . " is set! ";

echo "value is !" . \$_COOKIE[\$cookie_name];

}

?>

</body>

</html>

Q5 What is XML Parsing? Explain its usage with example.

→ XML parser is a software library or a package that provide interface for client application to work with XML document. it checks for proper format of the XML document modern day browser have built-in XML parser.

Passing a Text string

This example parses a text string into an XML Dom Object and extracts the info from it with javascript

Example :-

```
<html>
```

```
<body>
```

```
<p id = "demo"></p>
```

```
<script>
```

```
var text , parser , XML Doc ;
```

```
text = "<book store><book> " + "<title> Everyday  
Italian </title>" + "<author> Giada De Laurentis  
</author>"
```


+ "<year> 2005 </year>" + "</book> </bookstore>";

parser = new DOMParser();

XMLDoc = parser.parseFromString(text, "text/xml");

document.getElementById("demo")

innerHTML = XMLDoc.getElementsByTagName("title")

[0].childNodes[0].

nodeValue;

</script>

</body>

</html>