

**Guide for Oracle Connection with PHP**

**Eastern Illinois University**

**TEC- 5970A**

**PHP Development with Oracle**

**(Shyam Gurram, Graduate Assistant, Summer 2015)**

Note: This document was a result of hard work by Shyam Gurram, a graduate assistant in the School of Technology, during Summer 2015.

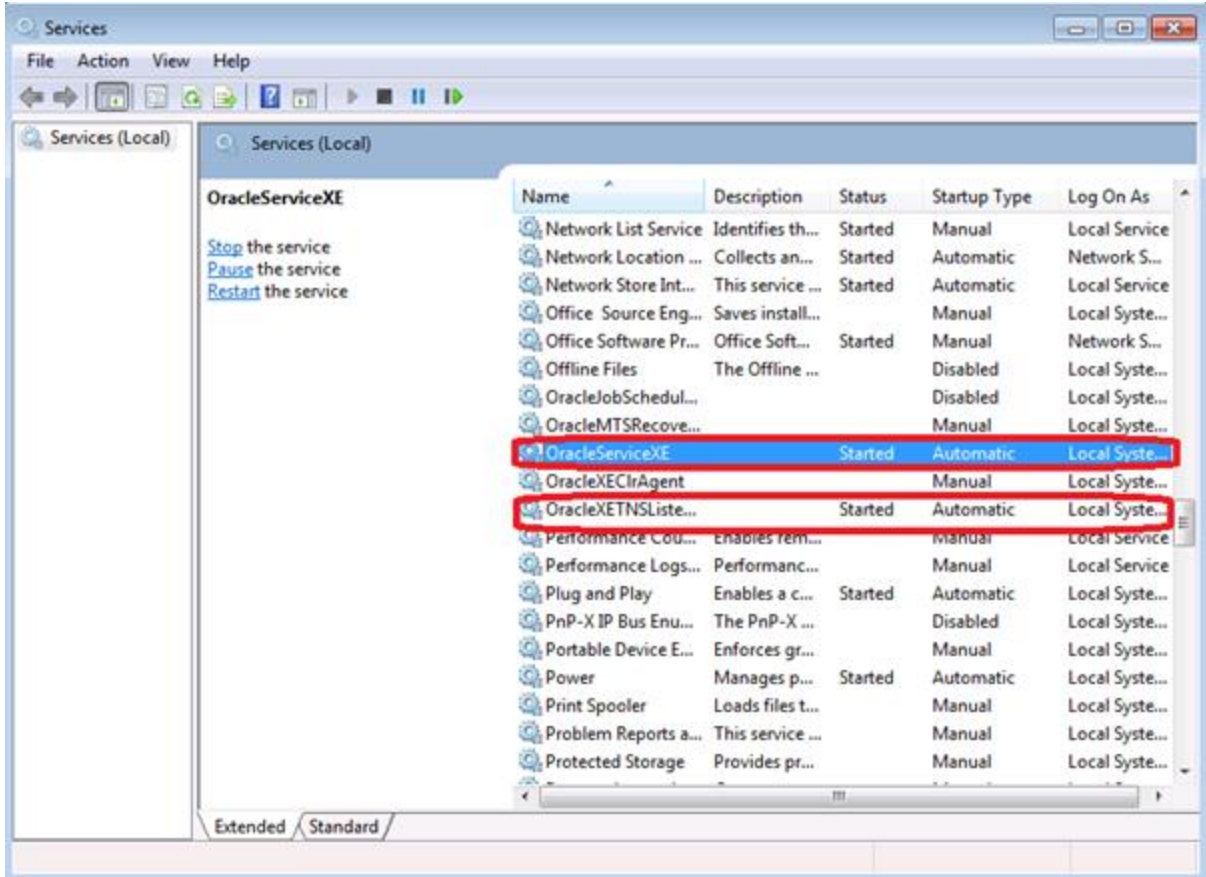
## Contents

How to Connect to Oracle Database as SYSTEM, HR User:.....	4
How to Unlock Oracle Database Account: .....	5
Query a Table in HR Account: .....	6
How to Run a SQL Script?.....	7
TNS NAMES.ORA File: .....	7
Connection to Oracle Database through PHP:.....	10
Writing a Sample PHP Program Connecting to the Database: .....	12
Sample Program of PHP: .....	14
Note:.....	16
How to Execute PHP Program on Zend Server: .....	16

Before connecting to a database we need to check the Oracle database connections in services. To check that first we need to open the services that is shown in the below figure.



You need to open the services and check the OracleServiceXE in start status, if not we need to start. And also we need to check the OracleXETNSListener.



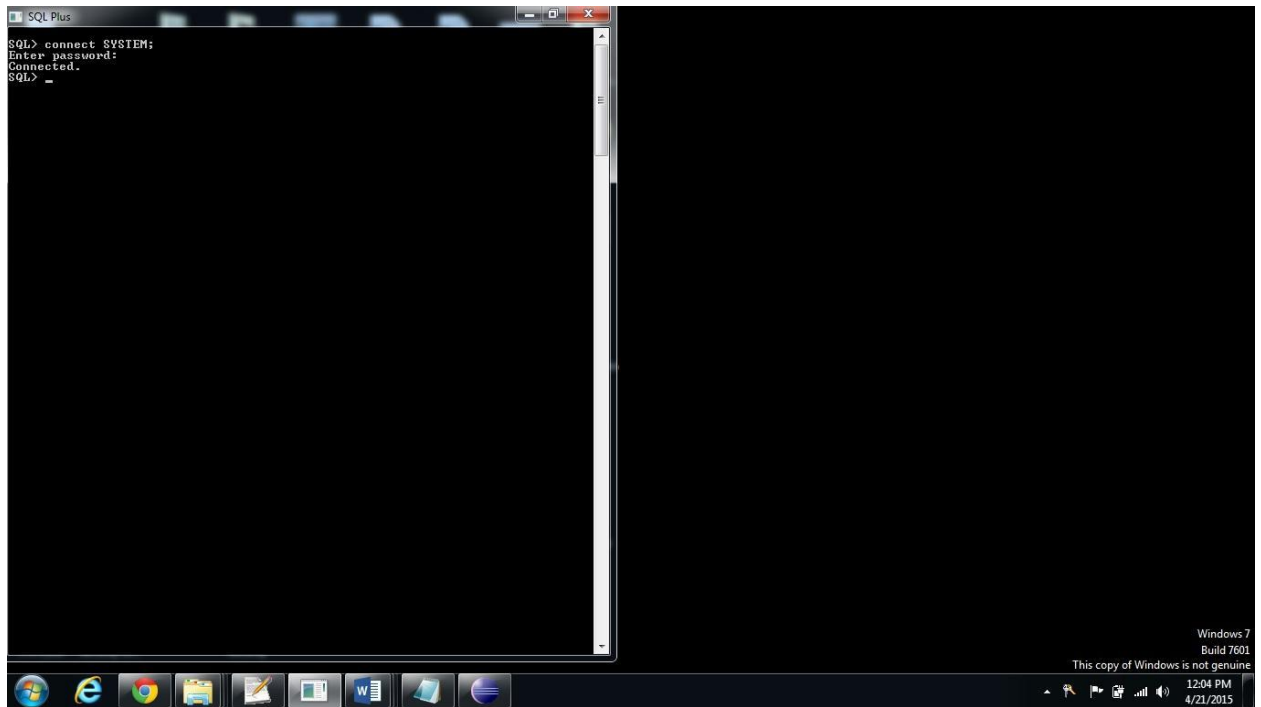
How to Connect to Oracle Database as SYSTEM, HR User:

1. Open the “SQL Command Line” (or SQL Plus) and type the following command.

```
SQL> Connect SYSTEM
```

2. You will be asked to enter a password where you need to enter SYSTEM User password given at that time of installation

Enter Password as “admin” (Your password may be different depending upon what password you have given the SYSTEM user);



**Figure 1.1 Connecting to SYSTEM User.**

### How to Unlock Oracle Database Account:

1. Open the SQL Command and give type the following command.
2. You will be asked to enter a password where you need to enter SYSTEM User password given at that time of installation
3. If the database account got locked there are some commands to unlock it.

```
SQL> ALTER USER <user_name> ACCOUNT UNLOCK;
```

For example,

```
SQL> ALTER USER HR ACCOUNT UNLOCK;
```

4. If you like to change the password, you may use the following SQL command:

```
SQL> ALTER USER <user_name> IDENTIFIED BY <password>
```

For example,

```
SQL> ALTER USER HR IDENTIFIED BY admin;
```

Now your HR Password is **admin**.

### Query a Table in HR Account:

1. Using HR credentials connect to DB, as below:

```
SQL>conn HR;
```

Then, enter the password for HR user.

2. If you want to query a sample table of Employees then give the following command.

```
SQL> SELECT * FROM EMPLOYEES;
```

3. If you need specific details then you need to give the condition using where clause.

```
SQL> SELECT * FROM EMPLOYEES WHERE FIRST_NAME  
='Neena';
```

## How to Run a SQL Script?

If we have a SQL script file named as **sample.sql** which is saved in C drive then use the following command to get your scripts run

```
@c:\sample.sql;
```

### TNS NAMES.ORA File:

The tnsnames.ora file is a configuration file that contains network service names mapped to connect descriptors for the local naming method, or net service names mapped to listener protocol addresses.

Depending upon your installation and version, the tnsnames.ora file may be located in different location. For example, in my installation, the file exists in the following folder:

```
c:\oracle\app\oracle\product\11.2.0\server\network\ADMIN
```

You may open the file using text editor such as notepad. For example, you may see the following from the tnsnames.ora file:

```
XE =
```

```
(DESCRIPTION =
```

```
(ADDRESS = (PROTOCOL = TCP)(HOST = Thota-PC)(PORT = 1521))
```

```
(CONNECT_DATA =
```

```
(SERVER = DEDICATED)
(SERVICE_NAME = XE)
)
```

Note: Host May vary from system to system in my system host is my personal computer name.

### TNSPING:

When a database is installed successfully on your machine and set up is made ready then comes the task of checking whether we are able to connect to DB or not. At this point comes the use of TNSPing. This is used to check whether the listener for a Service can be reached successfully or not.

Following command can be used to test the connectivity

Syntax is **“Tnsping net\_service\_name count”**

In the preceding command, the following arguments are used:

- *net\_service\_name* must exist in tnsnames.ora file or the name service in use, such as NIS.
- *count* determines how many times the program attempts to reach the server.

This argument is optional.



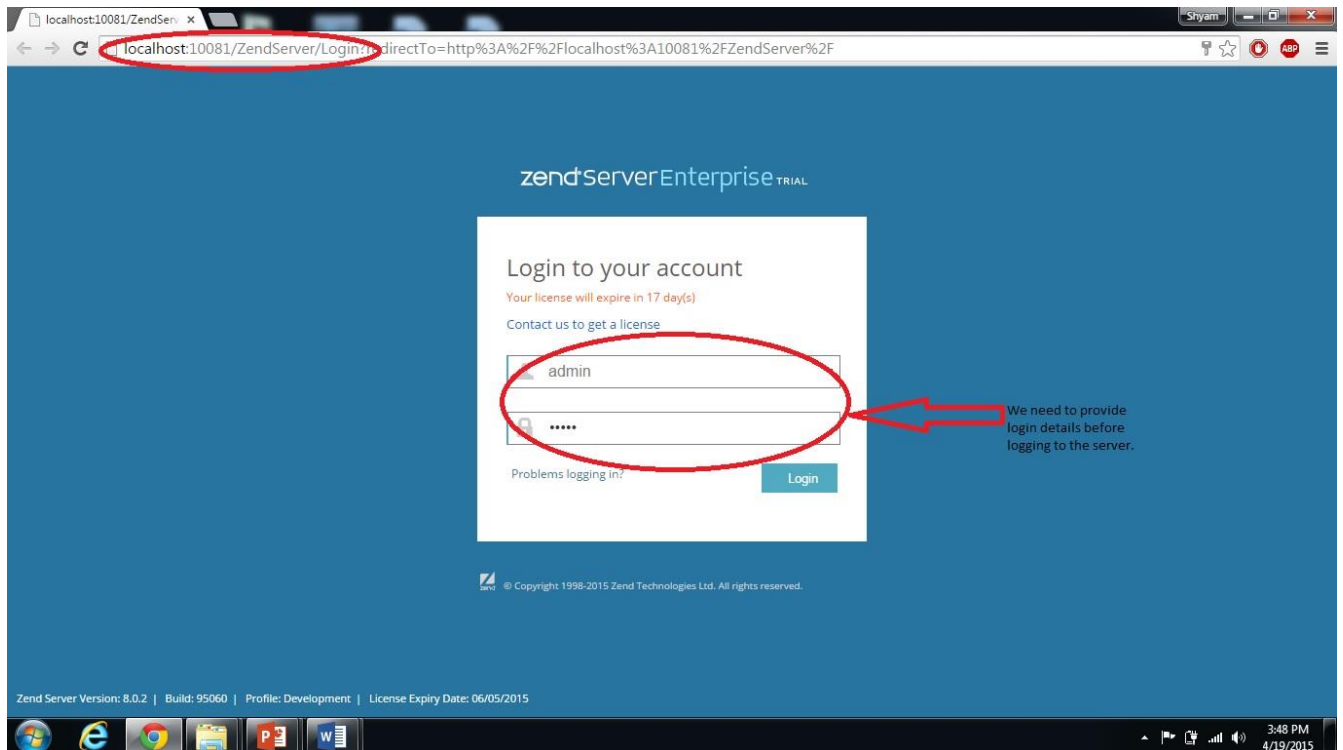
For example, the following works:

**C:> tnsping localhost count**

If the connection is successful it will show how much time it took to connect to DB else it will show an appropriate message for the failure of connection.

## Connection to Oracle Database through PHP:

- A. Before connecting to the database we need to perform some configuration changes in Zend Server.
- B. Firstly we need to click on the Zend server shortcut which was saved on desktop. Zend server will be opened in default browser, we need to give log in credentials (the log in details provided during installation) before logging on to the Zend server.



**Figure 1.5 providing the login credentials.**

- A. Once you log in, click the configurations tab, under that click on PHP link as shown in the below figure. Under PHP we have an **OCI8**

connection. We need to switch ON all the connections under OCI8. At the same time we need to switch OFF other connections not related to Oracle application such as MYSQL.

The screenshot shows the Zend Server Enterprise trial interface. The top navigation bar includes 'Overview', 'Applications', 'Configurations', and 'Administration'. The 'Configurations' tab is active, and the 'PHP' component is selected. Below the navigation bar, there is a table of components with columns for Name, Status, Version, Description, and Messages. The 'oci8' component is highlighted, and its status is 'Loaded'. Below the table, the 'Directives' section is visible, showing various configuration options for OCI8. The 'oci8.events', 'oci8.old\_oci\_close\_semantics', and 'oci8.privileged\_connect' options are set to 'OFF', while 'oci8.default\_prefetch' is set to '100', 'oci8.max\_persistent' is set to '-1', and 'oci8.persistent\_timeout' is set to '-1' seconds. The 'oci8.ping\_interval' is set to '60' seconds, and 'oci8.statement\_cache\_size' is set to '20'.

Name	Status	Version	Description	Messages
mysql	Loaded	0.1	MySQL Improved	
mysqlnd	Built-in	mysqlnd 5...	MySQL Native Driver	
oci8	Loaded	2.0.8	Oracle Call Interface	

Directives

Supports binding of PHP variables to Oracle placeholders, has full LOB, FILE and ROWID support, and allows you to use user-supplied define variables. Allows access to Oracle7, 8 and 9 databases. Note: This can only be enabled if the Oracle Instant Client is installed. See <http://www.php.net/manual/en/function.oci-connect.php> for information on using the `oci_connect` function to connect to your Oracle database. See <http://www.oracle.com/technology/tech/php/pdf/underground-php-oracle-manual.pdf> for more on the OCI Libraries and Oracle Extensions.

**oci8.connection\_class**  
This user defined text is used by Oracle 11g Database Resident Connection Pooling (DRCP) connections to sub-partition to connection pool.

**oci8.events**  
Allows PHP to be notified of database Fast Application Notification (FAN) events. The database must be configured to post FAN events.

**oci8.old\_oci\_close\_semantics**  
This option controls `oci_close()` behaviour.

**oci8.ping\_interval**  
The length of time that must pass before issuing a ping during `oci_pconnect()`.

**oci8.statement\_cache\_size**  
This option enables statement caching, and specifies how many statements to cache.

**oci8.default\_prefetch**  
This option enables statement prefetching and sets the default number of rows that will be fetched automatically after statement execution.

**oci8.max\_persistent**  
The maximum number of persistent OCI8 connections per process.

**oci8.persistent\_timeout**  
The maximum length of time that a given process is allowed to maintain an idle persistent connection.

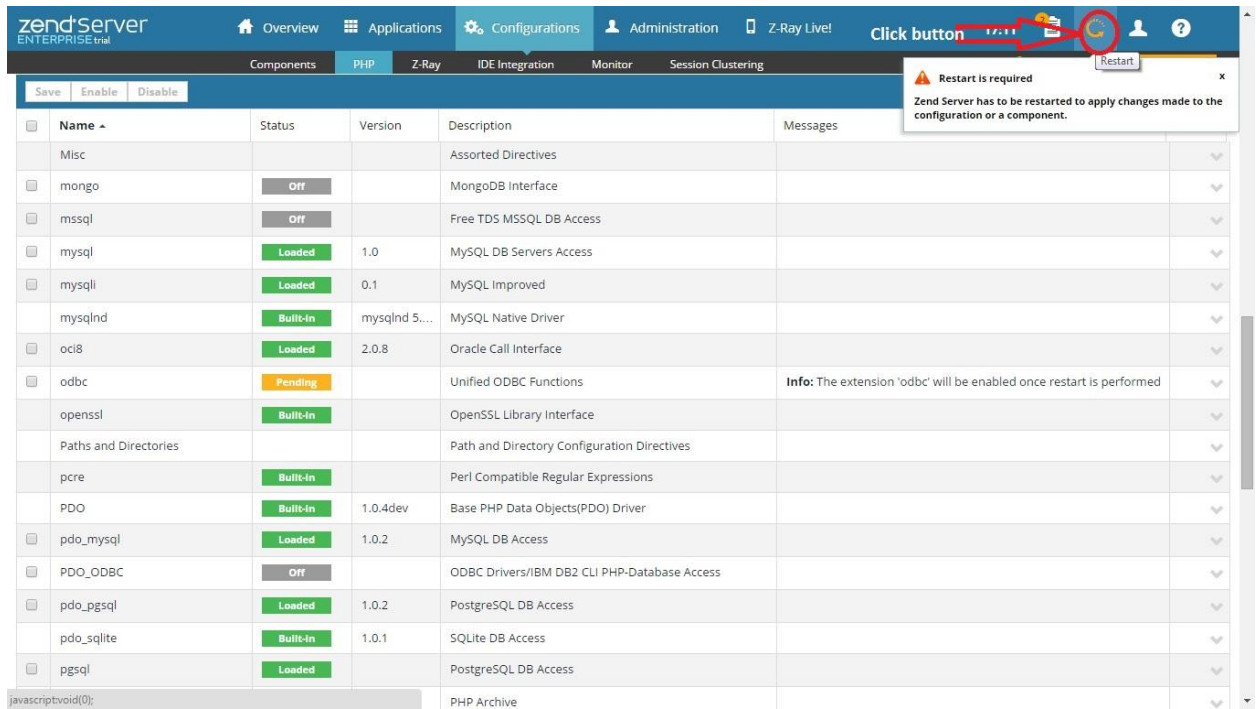
**oci8.privileged\_connect**  
This option enables privileged connections using external credentials.

oci8.connection_class				
oci8.events	OFF			
oci8.old_oci_close_semantics	OFF			
oci8.ping_interval	60	seconds		
oci8.statement_cache_size	20			
oci8.default_prefetch	100			
oci8.max_persistent	-1			
oci8.persistent_timeout	-1	seconds		
oci8.privileged_connect	OFF			

localhost:10081/ZendServer/SessionClustering/

**Figure 1.6 Enabling the Oracle Connections in the server.**

B. After pooling the connections, in order to reflect these we need to restart the server which is shown in the below figure.



**Figure 1.7 Restarting the server after all changes.**

## Writing a Sample PHP Program Connecting to the Database:

Before writing a PHP program, we need to know about some methods which will connect to the database.

### **OCI\_CONNECT:**

This method has three parameters:

1. Username
2. Password
3. Connection\_String.

This method will connect to the Oracle Database (Oracle 11g XE).

Username refers to Oracle database username (for example, HR). Password

refers to the password given for the associated Username.

Connection\_string refers to Oracle DB hostname which is present in your oracle database.

Path of your oracle DB hostname:

**c:\oracle\app\oracle\product\11.2.0\server\network\ADMIN**

In the above folder, there is a file called TNSnames.ora file. Under the line of XE you can find your database hostname. For example, the following exists in tnsnames.ora file:

XE =

(DESCRIPTION =

(ADDRESS = (PROTOCOL = TCP)(HOST = Thota-PC)(PORT = 1521))

(CONNECT\_DATA =

(SERVER = DEDICATED)

(SERVICE\_NAME = XE)

)

Where host name is “Thota-PC” and database service\_name is “XE.” Thus, connection\_string should be “Thota-PC/XE”

### **OCI\_PARSE:**

This method prepares an Oracle statement for execution.

**OCI\_EXECUTE:**

This method executes a statement. After execution, statements like `INSERT` will have data committed to the database by default. For statements like `SELECT`, execution performs the logic of the query.

**OCI\_FETCH\_OBJECT:**

This method returns the next row from a query as an object. Each attribute of the object corresponds to a column of the row. This function is typically called in a loop until it returns `false`, indicating no more rows exist.

**OCI\_CLOSE:**

This method will closes an Oracle Connection. It is recommended to close connections that are no longer needed because this makes database resources available for other users.

### Sample Program of PHP:

The Sample Program for Database Connection:

```
<html>
<head>
<style type="text/css">
table { border-collapse: collapse; }
td { border: solid 1px black; padding: 3px; }
```

```
</style>
</head>
<body>
<h2>Employees</h2>
<?php
$db = oci_connect('HR', 'zend1234', 'User-PC/XE');
if (!$db) {
    trigger_error('Unable to connect to database', E_USER_ERROR);
}
$sql = 'SELECT FIRST_NAME, LAST_NAME, HIRE_DATE FROM
EMPLOYEES';
$stmt = oci_parse($db, $sql);
oci_execute($stmt);
$count = 0;
echo '<table>';
while (($row = oci_fetch_object($stmt)) != false) {
    echo '<tr>';
    echo '<td>' . htmlentities($row->FIRST_NAME) . '</td>';
    echo '<td>' . htmlentities($row->LAST_NAME) . '</td>';
    echo '<td>' . htmlentities($row->HIRE_DATE) . '</td>';
    echo '</tr>';
    $count++;
}
echo '</table><br/>';
echo $count . ' record(s) found.';
oci_free_statement($stmt);
oci_close($db);
```

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

**Note:**

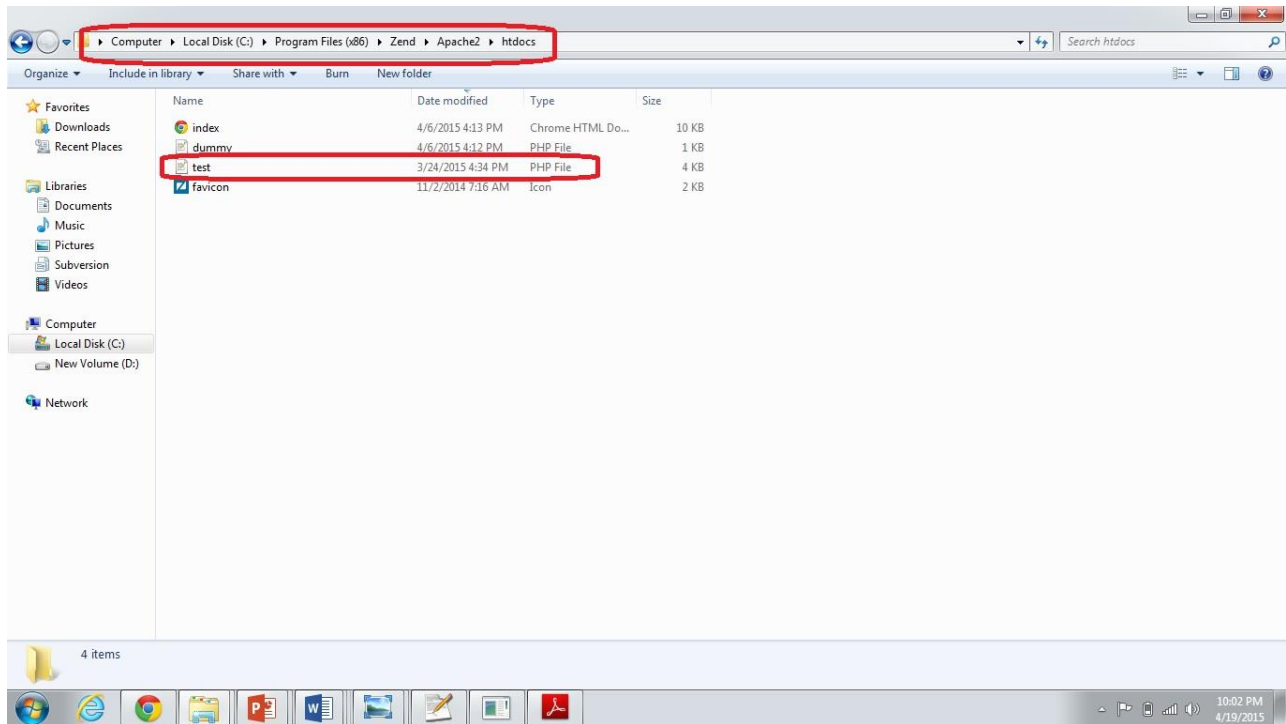
1. In the above program we used a method **OCI\_CONNECT** there we are using my system database name and password and the hostname.
2. Before executing this code in your system, ensure that Employees table created in your database and there are some rows in that table.

**How to Execute PHP Program on Zend Server:**

1. We may use **Eclipse PDT**, notepad++ or other text editors to write or develop your PHP program.
2. If we use an Eclipse PDT tool then debugging would be an easy process.
3. After writing a PHP program we need to save the file under htdocs folder in Apache. The path is shown in below.

**C:\Program Files (x86)\Zend\Apache2\htdocs.** Under **htdocs** we need to save the file. The below figure will give a brief explanation.





4. For example the above sample PHP file is saved with the name of test.PHP. In order to execute this file on your default browser, type ***localhost/test.php*** and click on enter then we can see the output.