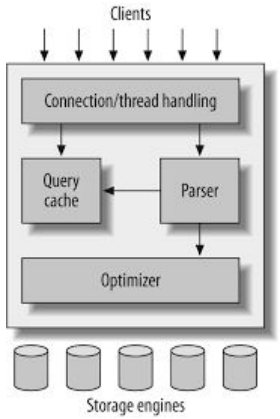
Unit - III

3.1 Introduction

MySQL is a RDBMS system. It consists of all the functions related to a database management system. It works based on client/server principle. Server consists of the database and the tools required to access the database. The client connects to the server to access the databases. This method allows more than one client to connect to the MySQL to access the database.



Server Architecture

Settingup an Account

To create a new account in MYSQL the steps given below are followed.

1. Create a databse
2. Create an User
3. Create a Table
4. Insert Values

Create a Database

In MYSQL to create a new database the create database command is used..

Syntax:

create database database_name;

example:

create database stud_db;

To display all the available database in the MYSQL server, show databses command is used.

syntax:

show databases;

This will show all the available database in the server tot he user.

From all the available database in the server, to select one particular database to work with use command is used.

syntax:

use <database name>;

Creating a user

To access the created database an user is needed. The default user available in MYSQL is “root” this used doesnot have a password. To create a new user create user command is used.

Syntax:

create user ‘username’@’localhost’ identified by ‘password’;

example:

create user ‘bala’@’localhost’ identified by ‘bala123$%^’;

The newly created uses does not have any access permission on that database. To give the required permission to the new user grant command is used.

syntax:

grant all privileges on *.* to ‘username’@’localhost’;

example:

grant all privileges on *.* to ‘bala’@’localhost’;

Create a Table

In a database to store the data the basic structure used is a table. To create a table in the database create table command is used.

syntax:

```
create table <table name>
{
 col1 datatype <key>,
 col2 datatype,
 col3 datatype
}
```

example:

```
create table stud_table
{
 roll_no INT PRIMARY KEY,
 stud_name VARCHAR(40) NOT NULL,
 total INT
}
```

Insert values

The newly created table will have only the structre and it will not contain any record. To store record in the newly created table insert command is used.

syntax:

insert into table_name(column list) values (value list);

example:

insert into stud_table(roll_no, stud_name, total) values (154101,’Anjay’, 535);

Starting, Terminating and writing your own sql program

The steps to access MYSQL database from PHP script is given below.

1. mysqli_connect command is used to make a connection from the PHP script to the MYSQL database.

2. Once the Connection is created to execute the queries in MYSQL from PHP script mysqli_query command is used.
3. After MYSQL query is executed from the PHP script the connection is closed using mysqli_close command.

example:

```
<?php
  $host = "127.0.0.1";
  $usr = "root";
  $pwd = "";
  $db = "stud_table";
  $db_con = mysqli_connect($host,$usr,$pwd,$db);
 $qry = "insert into stud_table(roll_no, stud_name, total) values (154102, 'Bala',489)";
  $res = mysqli_query($db_con, $qry);
  mysqli_close($db_con);
?>
```

MYSQL Data Types

When creating a table in MYSQL we need to give the datatype for each column that is present in the table.

Most Common Data Types

INT - Used to store integer data.
DOUBLE - used to store floating point number.
VARCHAR - used to store string data.

TIME - used to store time value in HH:MM:SS formatt.
DATE - used to store date value in YYYY-MM-DD format.
DATETIME - used to store date and time value in YYYY-MM-DD HH:MM:SS formatt.

Record selection technology

Record Selection Technology explain the syntax and procedure used in accessing and modifying the data from the database using the following commands.

- select
- update
- delete

Select Statement

To select and display a set of records from a single table or from multiple tables this statement is used.

General Form

```
select <Column name list>
from <table list>
where <conditions>
```

where clause condition types

- Relational and logical Operator
- like

- in and not in
- between and

To explain in detail about the usage of where clause condition types with example we will use stud_table that contains student details such as roll number, name and total marks

mysql> select * from stud_table;

```
+---------+-----------+-------+
| roll_no | stud_name | total |
+---------+-----------+-------+
| 154101  | Anjay     |   680 |
| 154102  | Balu      |   630 |
| 154103  | Depan     |   520 |
| 154105  | Rajan     |   620 |
+---------+-----------+-------+
```

Example: 1

In this example we will display the student details from stud_tableலில் whose total is greater than 600 and less than 700.

mysql> select * from stud_table where total>=600 and total<=700;

```
+---------+-----------+-------+
| roll_no | stud_name | total |
+---------+-----------+-------+
| 154101  | Anjay     |   680 |
| 154102  | Balu      |   630 |
| 154105  | Rajan     |   620 |
+---------+-----------+-------+
```

Example: 2

In this example select statement to get records of students with letter ‘a’ as second letter in student name is given. For this where clause with like condition is used.

mysql> select stud_name from stud_table where stud_name like '_a%';

```
+-----------+
| stud_name |
+-----------+
| Balu      |
| Rajan     |
+-----------+
```

Example: 3

This query is an example to select student records with the given two roll number using in clause.

mysql> select * from stud_table where roll_no in ('154101','154105');

```
+---------+-----------+-------+
| roll_no | stud_name | total |
+---------+-----------+-------+
| 154101  | Anjay     |   680 |
| 154105  | Rajan     |   620 |
+---------+-----------+-------+
```

Example: 4

Not In when used in the where clause will get the student records with roll number not given in the list.

mysql> select * from stud_table where roll_no not in ('154101','154105');

```
+---------+-----------+-------+
| roll_no | stud_name | total |
```

```
+---------+-----------+-------+
| 154102  | Balu      |   630 |
| 154103  | Depan     |   520 |
+---------+-----------+-------+
```

Example: 5

In this example a value rage is given for the marks and the student records that satisfy that given condition is selected. For this between and statement is used in where clause.

```
mysql> select * from stud_table where total between 650 and 700;
+---------+-----------+-------+
| roll_no | stud_name | total |
+---------+-----------+-------+
| 154101  | Anjay     |   680 |
+---------+-----------+-------+
```

Update Statement

Update statement is used to modify the values that are already present in the table. when where clause is omitted in this statement then all the records in the table will be updated.

General Form

update <table>
set < col1 = value1, col2 = value2>
where <selection condition>

Example

To add 20 marks to the students who have mark less than 600 in student table this query is used.

```
mysql> update stud_table set total = total+20 where total<=600;
Query OK, 1 row affected

mysql> select * from stud_table;
+---------+-----------+-------+
| roll_no | stud_name | total |
+---------+-----------+-------+
| 154101  | Anjay     |   680 |
| 154102  | Balu      |   630 |
| 154103  | Depan     |   540 |
| 154105  | Rajan     |   620 |
+---------+-----------+-------+
```

Delete Statement

To delete a record from the table delete statement is used. When delete statement is given without where clause all the records from the table will be deleted.

General Form

delete from <table> where <selection Condition>

Example

This example give a query that will delete all the student record from the Student Table for the students whose name ends with ‘ab’.

```
mysql> delete from stud_table where stud_name like '%an';
Query OK, 2 rows affected

mysql> select * from stud_table;
+---------+-----------+-------+
| roll_no | stud_name | total |
+---------+-----------+-------+
| 154101  | Anjay     |   680 |
| 154102  | Balu      |   630 |
+---------+-----------+-------+
```

Working with strings

The list of string manipulation function in MYSQL is given below. Theses functions are used to make changes in the data that are stored as varchar datatype.

- length
- reverse
- case change
- string join
- string replace
- repeat
- substring

length

To get the length of a given string length function is used.

General Form

```
select length (<string value>)
mysql> select length('Tamil Nadu');
+----------------------+
| length('Tamil Nadu') |
+----------------------+
|                   10 |
+----------------------+
```

reverse

This function is used to display the given string in reverse order.

General Form

```
select reverse(<string value>)

mysql> select reverse('Tamil Nadu');
+-----------------------+
| reverse('Tamil Nadu') |
+-----------------------+
| udaN limaT            |
+-----------------------+
```

case change

To change the case of a string from lower case to upper case or from upper case to lower case the functions ucase and lcase are used.

General Form

```
select ucase(<string value>);
select lcase(<string value>);
```

```
mysql> select ucase('welcome');
+------------------+
| ucase('welcome') |
+------------------+
| WELCOME      |
+------------------+
mysql> select lcase('WELCOME');
+------------------+
| lcase('WELCOME') |
+------------------+
| welcome          |
+------------------+
```

String Join

Concat function is used to join more than one different strings into a single string.

General Form

select concat(string1, string2, string3,....)

```
mysql> select concat('Welcome',' ','Home');
+------------------------------+
| concat('Welcome',' ','Home') |
+------------------------------+
| Welcome Home             |
+------------------------------+
```

Repeat

Repeat function is used to display a string for a given n number of times repeatedly.

General Form

select repeat(<string>,count)

```
mysql> select repeat('*',5);
+---------------+
| repeat('*',5) |
+---------------+
| *****     |
+---------------+
```

Substring

This function is used to get the substring from a string. For this the starting position is given from which the substring has to be extracted.

General Form

select substring(<string Value>, position)

position value 1ல் இருந்து ஆரம்பிக்கும்.

```
mysql> select substring('Tamil Nadu',5);
+---------------------------+
| substring('Tamil Nadu',5) |
+---------------------------+
| l Nadu                    |
+---------------------------+
```

Date Time Functions

Date and Time are important datatype in MYSQL. The functions used to manipulate date and time value are given below

- now
- curdate
- curtime
- datediff
- date_add
- date_sub
- dayname
- monthname
- hour
- minute

now

This function returns the current date and time from the mysql server.

```
mysql> select now();
+---------------------+
| now()               |
+---------------------+
| 2018-02-16 12:04:12 |
+---------------------+
```

curdate

This function is used to return only the current date from mysql server.

```
mysql> select curdate();
+------------+
| curdate()  |
+------------+
| 2018-02-16 |
+------------+
```

curtime

This function is used to display the current time from mysql server.

```
mysql> select curtime();
+-----------+
| curtime() |
+-----------+
| 12:07:47  |
+-----------+
```

datediff

This calculates the difference between two date values and returns the number of days as result.

General Form

select datediff(date1, date2);

```
mysql> select datediff(now(),'2018-02-06');
+------------------------------+
```

| datediff(now(),'2018-02-06') |

+------------------------------+

| 10 |

+------------------------------+

date_add

This function is used to add a value to the given date.

General Form

Select date_add(date value, INTERRVAL value <TYPE>)

In this the TYPE is DAY, MONTH or YEAR.

mysql> select date_add(curdate(),INTERVAL 4 DAY);

+-----------------------------------+

| date_add(curdate(),INTERVAL 4 DAY) |

+-----------------------------------+

| 2018-02-20 |

+-----------------------------------+

date_sub

This function is used to subtract a value from the given date.

General Form

Select date_sub(date value, INTERRVAL value <TYPE>)

In this the TYPE is DAY, MONTH or YEAR.

mysql> select date_sub(curdate(),INTERVAL 4 YEAR);

+-----------------------------------+

| date_sub(curdate(),INTERVAL 4 YEAR) |

+-----------------------------------+

| 2014-02-16 |

+-----------------------------------+

dayname

This function is used to display the day name of the given date.

mysql> select dayname(now());

+----------------+

| dayname(now()) |

+----------------+

| Friday |

+----------------+

monthname

This function is used to display the month name of the given date.

mysql> select monthname(now());

+------------------+

| monthname(now()) |

+------------------+

| February |

+------------------+

hour

This function displays only the hour part from the given time.

mysql> select hour(curtime());

+-----------------+

| hour(curtime()) |

+-----------------+

```
|          12 |
+-----------------+
```

minute

This function is used to get the minute part from the given time.

```
mysql> select minute(curtime());
+-------------------+
| minute(curtime()) |
+-------------------+
|          22 |
+-------------------+
```

Sorting Query Result Module

The records obtained from the select statement are sorted using the order by command.

General Form

```
select <column>
from <table>
where <condition>
order by <column Name> asc/desc
```

asc-sort in ascending order

desc- sort in decending order.

```
mysql> select * from stud_table order by total desc;
+---------+-----------+-------+
| roll_no | stud_name | total |
+---------+-----------+-------+
| 154101  | Anjay     |   680 |
| 154102  | Balu      |   630 |
+---------+-----------+-------+
```

Generating Summaries

Summaries are used to get an overall view about the data present in the database.

functions

- count, distinct, group by
- sum & avg
- min & max

```
mysql> select * from stud_asign;
+------+--------+------+
| roll | asign  | mark |
+------+--------+------+
| 101  | asign1 |   19 |
| 101  | asign2 |   20 |
| 102  | asign2 |   20 |
| 103  | asign1 |   18 |
| 103  | asign2 |   20 |
+------+--------+------+
```

count, distinct, group by

This function is used to get the number of values present in a column.

```
mysql> select count(roll) from stud_asign;
```

```
+-------------+
| count(roll) |
+-------------+
|           5 |
+-------------+
```

To get the number of values that does not repeat in a column distinct function is used.

```
mysql> select count(distinct roll) from stud_asign;
+----------------------+
| count(distinct roll) |
+----------------------+
|                    3 |
+----------------------+
```

To group the records in a table based on a column the group by command is used.

```
mysql> select roll, count(asign) from stud_asign group by(roll);
+------+--------------+
| roll | count(asign) |
+------+--------------+
| 101  |            2 |
| 102  |            1 |
| 103  |            2 |
+------+--------------+
```

sum & avg

To add all the values in a column sum function is used.

```
mysql> select roll, sum(mark) from stud_asign group by roll;
+------+-----------+
| roll | sum(mark) |
+------+-----------+
| 101  |        39 |
| 102  |        20 |
| 103  |        38 |
+------+-----------+
```

To calculate the average of all the values present in a column avg function is used.

```
mysql> select roll, avg(mark) from stud_asign group by roll;
+------+-----------+
| roll | avg(mark) |
+------+-----------+
| 101  |   19.5000 |
| 102  |   20.0000 |
| 103  |   19.0000 |
+------+-----------+
```

min & max

To get the smallest and largest value from a column min and max functions are used.

```
mysql> select min(mark) from stud_asign;
+-----------+
| min(mark) |
```

```
+-----------+
|        18 |
+-----------+
```

```
mysql> select max(mark) from stud_asign;
+-----------+
| max(mark) |
+-----------+
|        20 |
+-----------+
```

Working with metadata

Metadata is data about data. It consists of information regarding the data present in the database.

Three Types of metadata

- query
- database
- MYSQL server

Query

To get the number of rows affected from a query execution in MYSQL php uses the following function.

syntax

```
$count = mysqli_affected_rows(Connection Object);
```

Database

To get the list of database availabe in MYSQL server php uses the following function.

syntax

```
$db_list = mysql_list_dbs(connection object);
```

MYSQL Server

To get the information about the databse and users present in MYSQL server from php the following queries are executed.

To get the list of databse in the server - select database();

To get the list of users in the database - select user();

Working with MYSQL Sequence

Sequence is used to make a column to automatically increase it value in a table. This is given when the table is created using the create command.

syntax

AUTO_INCREMENT

example

```
create table bill(
bill_no int AUTO_INCREMENT PRIMARY KEY,
prod vacate(30),
```

qty int,

price double);

AUTO_INCREMENT column must be a PRIMARY KEY.

MYSQL and WEB

- Speed - As MYSQL runs on a server it works consistently and fast.
- Cost - MYSQL is distributed under Open Source Software GNU license so there is no cost associated with using it in our application.
- Ease of use - MYSQL provides both GUI based tools and command prompt based access to manage the database stored in it.
- Query Language Support - MYSQL database supports almost all the query available in Query Language.
- Capability - As MYSQL database is based on the client/server architecture it allows more than one user to make connection to it and work with it.
- Connectivity - MYSQL database allows multiple diverse application to access its database. (EX) MS-Access, PHP, Java.
- Portability - MYSQL database is not bound to any particular Operating System. It can work on any Operating System.

PHP and SQL database:-PHP and LDAP

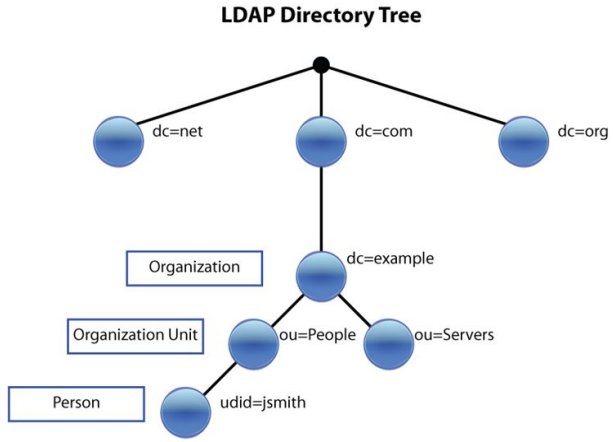
LDAP (Lightweight Directory Access Protocol). This is used to search and get information from the server. This is mainly used in large organisations to store the address details of its employees. This allows the data to be effectively searched and located.

Root - starting point

Domain Component - next leaf

Organizational Unit - next Group

Unique Identifier - for identification



Accessing LDAP using PHP

The steps used to access LDAP from PHP are.

1. Connect to LDAP server
2. Bind to the server
3. Search Server
4. Get Result

Connect to LDAP server

To access the data stored in the LDAP server first we need to connect to the server.

syntax

con_obj = ldap_connect(servername)

Bind to the server

Once the connetion is made to the Server. The program has to bind to a particular domain using the domain name and its corresponding password. For this ldap_bind syntax is used.

syntax

ldap_bind(con_obj, domain, password)

Search Server

Once the client logins to the server to search the server ldap_search syntax is used. In this the connection object and the branch name in which the actual search is to be made and the name which is to be searched are given as filter parameter.

syntax

result=ldap_search(con_obj, branch, filter)

Get Result

Search resultஇன் valueகளை பிரித்துஎடுக்க lapd_get_entries syntax உபயோகிக்கப்படுகிறது.

syntax

ldap_get_entries(con_obj,result)

Example

```
<html>
<head>
   <title>LDAP Search</title>
</head>
<body>
<?php
   $ldap_dn = "cn=read-only-admin,dc=example,dc=com";
   $ldap_password = "password";

   $ldap_con = ldap_connect("ldap.forumsys.com");

  ldap_set_option($ldap_con, LDAP_OPT_PROTOCOL_VERSION, 3);

   if(ldap_bind($ldap_con, $ldap_dn, $ldap_password)) {

    $filter = "(cn=Albert Einstein)";
    $result = ldap_search($ldap_con,"dc=example,dc=com",$filter) or
exit("Unable to search");
    $entries = ldap_get_entries($ldap_con, $result);

          print "<pre>";
          print_r ($entries);
          print "</pre>";
   } else {
          echo "Invalid user/pass or other errors!";
   }
?>
</body>
```

```
</html>
```

Sending and Receiving email

To send email from PHP mail() function is used. This function requires a web server to send email it will not send email when execute from localhost. This function requires a minimum of four parameters. This function will return true value if the email is sent properly.

syntax

mail(to,from, message, header)

example

```
<html>
  <head>
        <title>Sending HTML email using PHP</title>
  </head>
  <body>
   <?php
    $to = "autpc123@gmail.com";
    $subject = "Testing email function";

    $message = "<b>Sample mail Form Server.</b>";
    $message .= "<h1>This is headline.</h1>";

    $header = "From:dns@audns.co.in \r\n";
    $header .= "Cc:dns@audns.co.in \r\n";
    $header .= "Content-type: text/html\r\n";

    $retval = mail ($to,$subject,$message,$header);

    if( $retval == true ) {
      echo "Message sent to $to from dns@audns.co.in successfully...";
    }else {
      echo "Message could not be sent...";
    }
  ?>
  </body>
```



```
</html>
```

To read email using PHP IMAP(Internet Message Access Protocol) is used. For this to work we need to get the setting from the email servers, for example gmails imap server setting are server name is imap.gmail.com and port number is 993. To make a connection to the gmail using the imap settings the mail id's username and password is to be given correctly. To read the email first imap_open() funtion is used to open the connection, it has three parameters server host details, username and password.

syntax

mail_obj = imap_open(host,username,password)

Once the email is openned imap_search() function is used to get the emails from the mail id using mail_obj and search folder as the parameter.

syntax

result = imap_search(mail_obj , 'ALL')

The obtained email are sorted using the rsort function.

syntax

rsort(result)

To display the details regarding each email imap_fetch_overview() function is used. Using the result of this function the details of the email such as subject, from and date detail can be obtained.

syntax

imap_fetch_overview(mail_obj, indv_email)

After completing the reading operation the email connection is closed using the imap_close() function.

example

```
<html>
<head>
   <title>
        IMAP get emails
   </title>
</head>
<body>
<?php

$hostname = '{imap.gmail.com:993/imap/ssl}INBOX';
$username = user@gmail.com';
$password = ‘password’;

$inbox = imap_open($hostname,$username,$password) or die('Cannot connect to Gmail: ' . imap_last_error());

$emails = imap_search($inbox,'ALL');
if($emails) {
   $output = '';
   rsort($emails);
   foreach($emails as $email_number) {
        $overview = imap_fetch_overview($inbox,$email_number,0);
        $output.= '<div>';
        $output.= '<span class="subject">'.$overview[0]->subject.
                  '</span><br> ';
        $output.= '<span class="from">'.$overview[0]->from.'</span>';
        $output.= '<span class="date">on '.$overview[0]->date.
                 '</span>';
        $output.= '</div><br>';
   }

   echo $output;
}

imap_close($inbox);
?>
</body>
</html>
```

PHP Database Connectivity

To connect from PHP to MYSQL databse mysqli_connect function is used. In this function four parameters, host address, username, password and database name are given. To obtain the error in connection if any mysqli_connect_errno() function is used\.

example

```
<html>
<title>
   DBConnection
</title>
<body>
<?php
 $host = "127.0.0.1";
 $usr = "root";
 $pwd ="";
 $db = "stud_db";
 $con = mysqli_connect($host,$usr,$pwd,$db);

 if (mysqli_connect_errno())
   echo "<br>Connection Failed";
 else
   echo "<br>connection Successful";
?>
</body>
</html>
```

Retriving data from MYSQL

To retrieve data from MYSQL database first we need to connect to the database. Once the connection is established with the MYSQL server from php, using the connection object and the query to be executed as parameter mysqli_query function is used to retrieve the data.

syntax

result = mysqli_query(con_obj,query)

To get the query result first mysqli_num_rows function is used this will return the number of rows obtained from executing the query.

syntax

mysqli_num_rows(result)

To get the rows from the result one by one mysqli_fetch_row function is used. In this the column values are obtained using the number index.

syntax

mysqli_fetch_row(result)

Example

```
<html>
<title>
   DBRetrival
</title>
<body>
<?php
 $host = "127.0.0.1";
 $usr = "root";
 $pwd ="";
 $db = "stud_db";
 $con = mysqli_connect($host,$usr,$pwd,$db);;
 if (mysqli_connect_errno())
   echo "Connection Failed";
 else
 {
        $qry = "select roll_no, stud_name from stud_table";
        $res = mysqli_query($con,$qry);
        if($res==TRUE)
        {
        echo "numrows = ".mysqli_num_rows($res)."<br><br>";
        echo "<table><tr><th>Roll NO</th><th>Name</th></tr>";
        while($row = mysqli_fetch_row($res))
        {
                echo "<tr><td>$row[0]</td><td>$row[1]</td></tr>";
        }
   }
        else
                echo "Error in query";
 }
?>
</body>
</html>
```

Manipulating Data in MYSQL using PHP

To manipulate data in MYSQL database first a connection is made to the MYSQL server from php. Then the data manipulation query is built as a string variable. Using the connection object and the query is executed using mysqli_query function.

syntax

mysqli_query(con_obj,query)

To check if the query is executed correctly and data is manipulated in the database mysqli_affected_rows function is used.

syntax

mysqli_affected_rows(con_obj)

Example

```
<html>
<title>
   DB Manipulate Data
</title>
<body>
<?php
 $host = "127.0.0.1";
 $usr = "root";
 $pwd ="";
 $db = "stud_db";
 $con = mysqli_connect($host,$usr,$pwd,$db);;
 if (mysqli_connect_errno())
   echo "Connection Failed";
 else
 {
   $qry = "update stud_table set stud_name = 'Depan' where roll_no = 154103";
   $res = mysqli_query($con,$qry);
  $rows = mysqli_affected_rows($con);
   if($rows==1)
   {
    echo "Data Updated Successfully<br>";
   }
   else
    echo "Error in query";
 }
?>
</body>
</html>
```

Review Qustions

Part - A

1. Draw the MYSQL server Architecture.
2. What is the syntax used to create user in MYSQL?
3. Write about any two MYSQL Data Types.
4. Write the select query with between and where clause.
5. Write about any two string functions in MYSQL.
6. Write about any two Data Time functions in MYSQL.
7. Write about sum and avg functions in MYSQL.
8. Expand LDAP.
9. Exapnd IMAP.

Part - B

1. Write the steps in settingup an account in MYSQL.
2. Write a program to connect to MYSQL.
3. Explain about update and delete statements.
4. Explain about sorting query result module.
5. Write short note on working with MYSQL sequence.
6. Write about MYSQL and WEB.
7. How email is sent using PHP? Explain.

Part - C

1. Explain in detail about PHP database Connectivity.
2. Write about Retriving data from MYSQL.
3. Explain in detail about Record Selection Technology.

4. How sending and receiving emails is performed in MYSQL?
5. Write about PHP and LDAP.
6. Explain generating summaries.