

PHP MySQLi = PHP MySQL Improved!

The MySQLi functions allows you to access MySQL database servers.

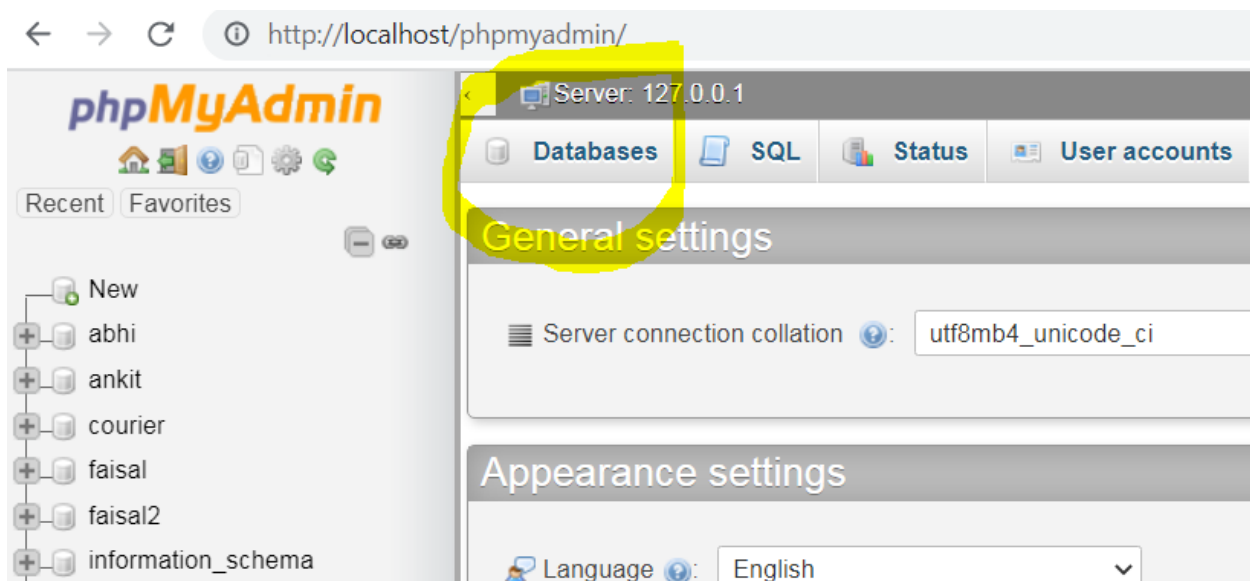
Note: The MySQLi extension is designed to work with MySQL version 4.1.13 or newer

The **MySQLi Extension** (MySQL Improved) is a [relational database](#) driver used in the [PHP programming language](#) to provide an interface with [MySQL databases](#).

How can you create database in phpmyadmin ?

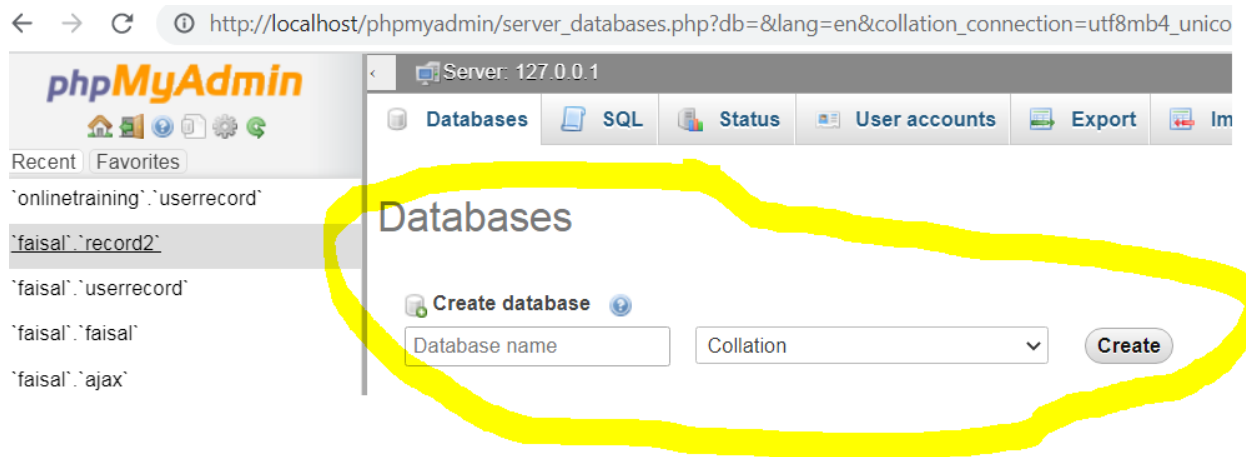
Ans:- open any browser and type

<http://localhost/phpmyadmin/>



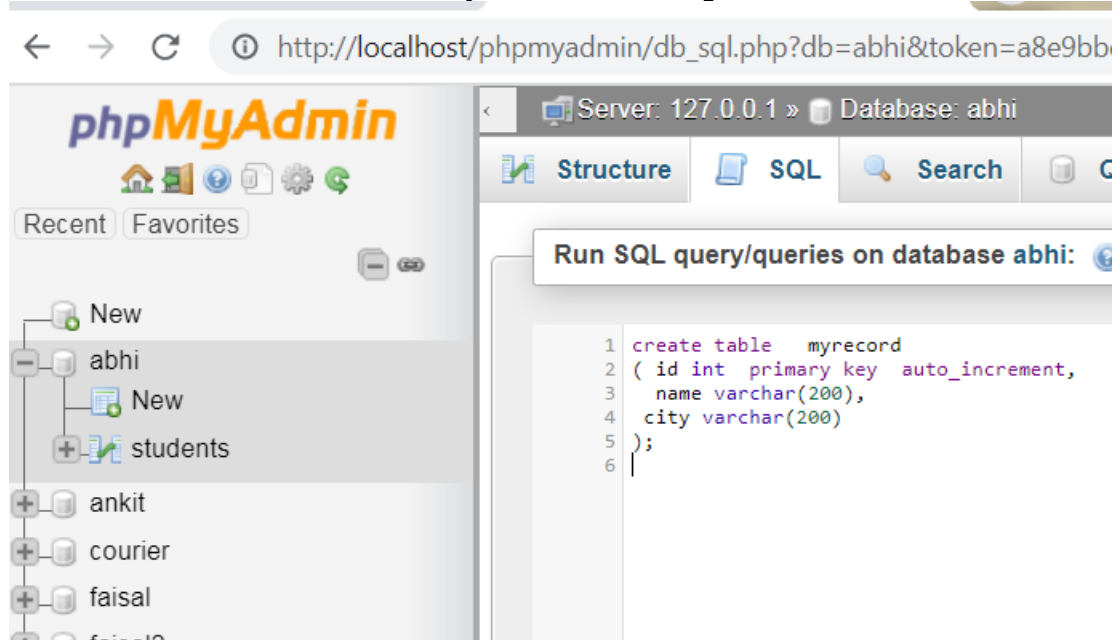
Then click on databases

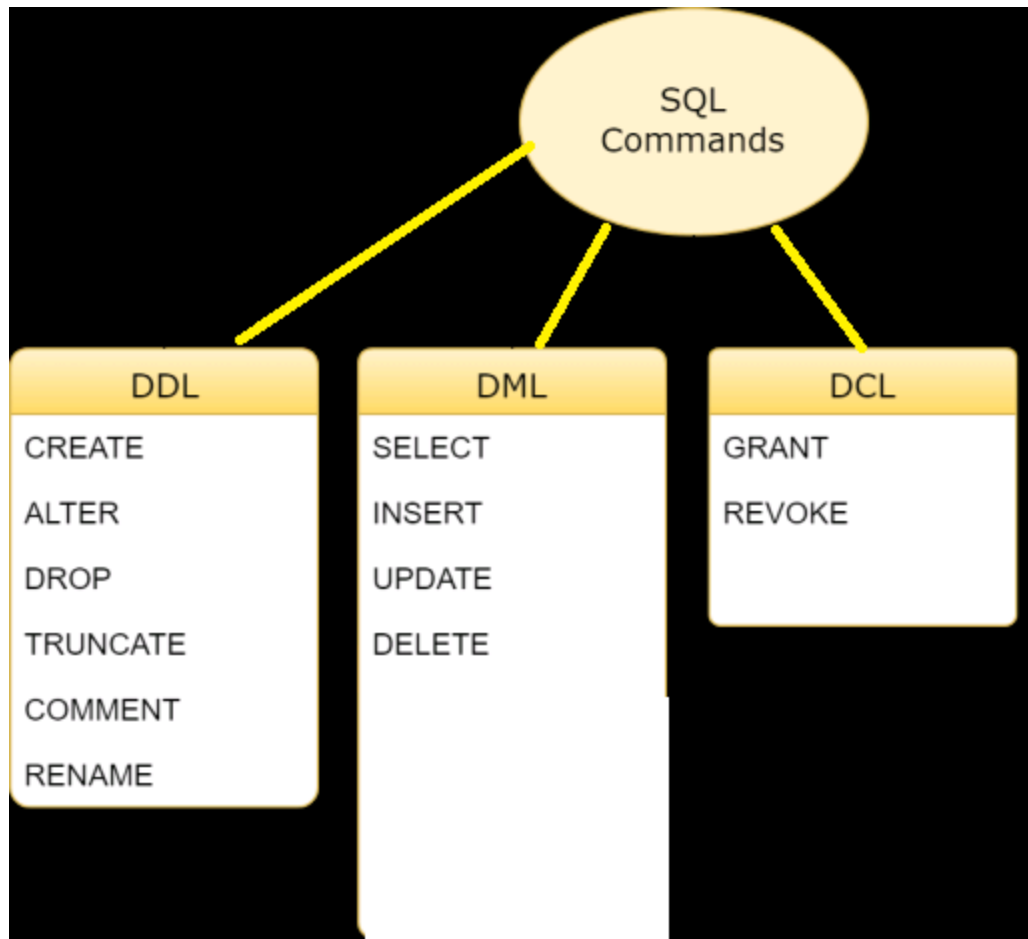
Then you will see option to enter your database name



Enter your database name and click on create.

And after that create table myrecord with sql create table command .





DDL:

DDL is short name of Data Definition Language, which deals with database schemas and descriptions, of how the data should reside in the database.

- **CREATE** – to create database and its objects like (table, index, views, store procedure, function, and triggers)
- **ALTER** – alters the structure of the existing database
- **DROP** – delete objects from the database
- **TRUNCATE** – remove all records from a table, including all spaces allocated for the records are removed
- **COMMENT** – add comments to the data dictionary
- **RENAME** – rename an object

DML:

DML is short name of Data Manipulation Language which deals with data manipulation and includes most common SQL statements such SELECT, INSERT, UPDATE, DELETE etc, and it is used to store, modify, retrieve, delete and update data in a database.

- **SELECT** – retrieve data from a database
- **INSERT** – insert data into a table
- **UPDATE** – updates existing data within a table
- **DELETE** – Delete all records from a database table

DCL:

DCL is short name of Data Control Language which includes commands such as GRANT and mostly concerned with rights, permissions and other controls of the database system.

- GRANT – allow users access privileges to database
- REVOKE – withdraw users access privileges given by using the GRANT command

run your xampp & open any browser & type "localhost/phpmyadmin"
:

(1)to create database use command:

```
SQL> create database practice;
```

(2)to create table use command:

```
SQL>create table myrecord  
  ( id int,  
    name varchar(20)  
  );
```

(3)to insert record into table:

there is two type:

type 1:

```
insert into myrecord(id,name)values('1','om');
```

type 2:

```
insert into myrecord values('2','som');
```

(4)Display record from table:

(a) how to display all record from table:

```
SQL>select * from myrecord;
```

(b)how to display particular column record:

```
SQL>select name from myrecord; //for single column
```

```
SQL>select id,name from myrecord; //for more than one column
```

(c)how to display particular rows record:

```
SQL>select * from myrecord where id='1'; // for single row
```

```
SQL>select * from myrecord where id in(1,2); //for more than one row
```

(5)how to update record of a table:

```
SQL>update myrecord set id='3',  
name='omanjali' where id='1'; //update by id
```

```
SQL>update myrecord set id='3',  
name='omanjali' where name='som'; //update by name
```

(6)how to delete record from table:

```
SQL>delete from myrecord ; //delete all record
```

SQL>delete from myrecord where id='1'; //delete by id

SQL>delete from myrecord where name='om'; //delete by name

(7)how to add new column into a table :

SQL>alter table myrecord
add city varchar(200);

(8)how to modify size or type of a column:

SQL>alter table myrecord
modify city varchar(20);

(9)how to rename a table :

SQL>rename table myrecord to mynewrecord;

(10)how to rename a column:

syntax:

```
alter  
table table_name change old_column_name new_column_name data_type  
;
```

example:

```
SQL>ALTER TABLE  
mynewrecord CHANGE name myname VARCHAR(20);
```

(11)how to change position of column into table:

SYNTAX:

```
ALTER TABLE  
tablename MODIFY COLUMN columnname datatype AFTER column;
```

example:

```
SQL>ALTER
TABLE mynewrecord MODIFY COLUMN name VARCHAR(50) A
FTER id;
```

(12)how to drop a particular column :

```
SQL>alter table mynewrecord drop name;
```

(1)create table myrecord :

```
create table myrecord
( id int primary key auto_increment,
  name varchar(200),
  city varchar(200)
);
```

(2)write code for insert.html :

```
<form action=insert.php method=post>
name <input type=text name=name>
city <input type=text name=city>
<input type="submit" name="submit" value="submit">
</form>
```

(3)now write code for insert.php file:

```
<?php
$servername = "localhost";
$username = "root";
$password = "";
$dbname = "test";

$name=$_POST['name'];
$city=$_POST['city'];

// Create connection
$conn = mysqli_connect($servername, $username, $password, $dbname);
```

```

// Check connection
if (!$conn) {
    die("Connection failed: " . mysqli_connect_error());
}

$sql = "INSERT INTO myrecord (name,city)
VALUES ('$name','$city)";

$result=mysqli_query($conn, $sql) or die(mysqli_error());

if($result)
{
    echo "record inserted";
}
else
{
    echo "record not inserted";
}
mysqli_close($conn);

?>

```

Now example for update .html:

(1)write code for update.html:

```

<form action=update.php method=post>
id <input type=text name=id>
name <input type=text name=name>
city <input type=text name=city>
<input type="submit" name="submit" value="submit">
</form>

```

(2)write code for update.php :

```

<?php
$servername = "localhost";
$username = "root";
$password = "";

```



```

$dbname = "test";
$id=$_POST['id'];
$name=$_POST['name'];
$city=$_POST['city'];

// Create connection
$conn = mysqli_connect($servername, $username, $password, $dbname);
// Check connection
if (!$conn) {
    die("Connection failed: " . mysqli_connect_error());
}

$sql = "update myrecord set name='$name',city='$city' where id='$id'";

$result=mysqli_query($conn, $sql) or die(mysqli_error());
if($result)
{

    echo "record updated";

}
else
{

    echo "record not updated";
}

mysqli_close($conn);

?>

```

now write code for search :

(1)write code for search.html :

```

<form action=search.php method=post>

name <input type=text name=name>

```

```
<input type="submit" name="submit" value="submit">
```

```
</form>
```

(2)write code for search.php :

```
<?php
```

```
$servername = "localhost";
```

```
$username = "root";
```

```
$password = "";
```

```
$dbname = "test";
```

```
$name=$_POST['name'];
```

```
// Create connection
```

```
$conn = mysqli_connect($servername, $username, $password, $dbname);
```

```
// Check connection
```

```
if (!$conn) {
```

```
    die("Connection failed: " . mysqli_connect_error());
```

```
}
```

```
$sql = "select * from myrecord where name='$name'";
```

```
$result=mysqli_query($conn, $sql) or die(mysqli_error());
```

```
if($result)
```

```
{
```

```
while($row=mysqli_fetch_array($result))
```

```
{
```

```
    echo $row['name'];
```

```
    echo $row['city'];
```

```
}
```

```
}
```

```
else
```

```
{
```

```
    echo "record not found";
```

```
}
```

```
mysqli_close($conn);
```

?>

Now write code for delete record :

(1)write code for delete.html file:

```
<form action=delete.php method=post>
id <input type=text name=id>

<input type="submit" name="submit" value="submit">
</form>
```

(2)write for delete.php file:

```
<?php
$servername = "localhost";
$username = "root";
$password = "";
$dbname = "test";
$id=$_POST['id'];

// Create connection
$conn = mysqli_connect($servername, $username, $password, $dbname);
// Check connection
if (!$conn) {
    die("Connection failed: " . mysqli_connect_error());
}

$sql = "delete from myrecord where id='$id'";

$result=mysqli_query($conn, $sql) or die(mysqli_error());
if($result)
{
    echo "record deleted";
}
else
{
```

```
echo "record not deleted";  
}
```

```
mysqli_close($conn);
```

```
?>
```