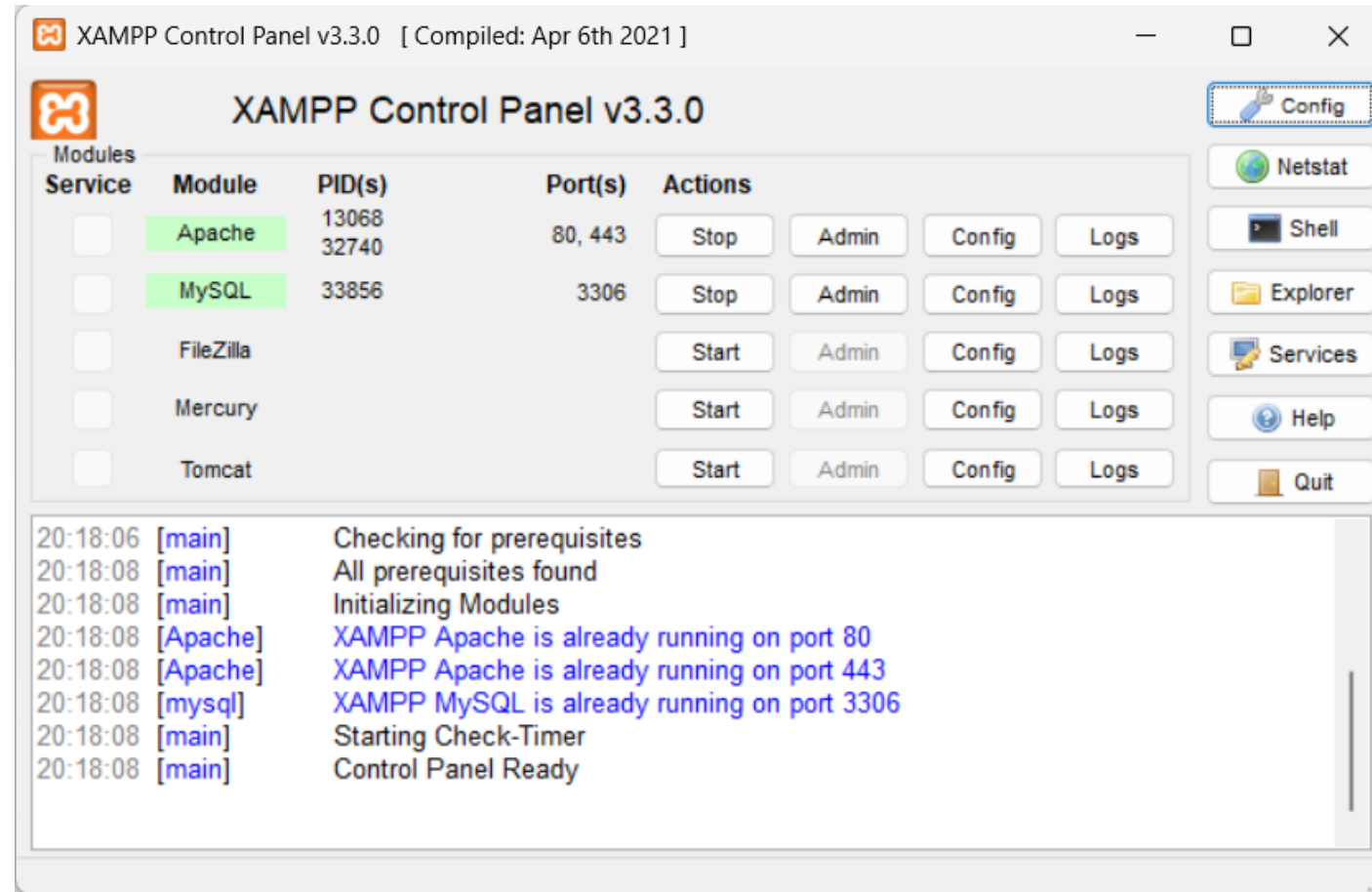


# XAMPP on Windows

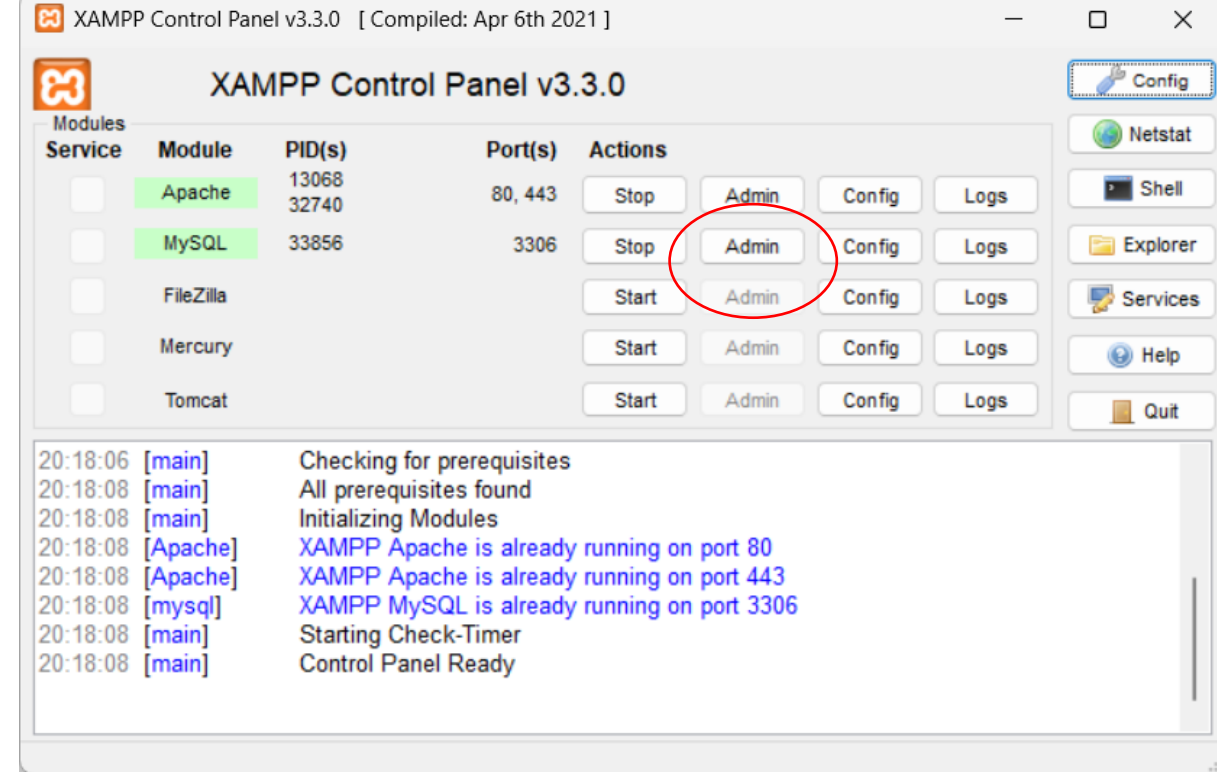
# Start the XAMP control panel

- C:\xampp→xampp-control.exe



# Launch mysql

- Click on MySQL Admin
- <http://localhost/phpmyadmin/>



# Create New database → student

The screenshot shows the phpMyAdmin web interface. On the left sidebar, the 'New' button is highlighted with a red rectangle. The main panel displays the 'Databases' tab with a 'Create database' form. The form includes a 'Database name' input field and a 'Collation' dropdown menu set to 'utf8mb4\_general\_ci'. A 'Create' button is located to the right of the form. Below the form, there are checkboxes for 'Check all' and 'Drop'. A table lists existing databases with their collations and actions.

Database	Collation	Action
<input type="checkbox"/> information_schema	utf8_general_ci	Check privileges
<input type="checkbox"/> mysql	utf8mb4_general_ci	Check privileges
<input type="checkbox"/> performance_schema	utf8_general_ci	Check privileges
<input type="checkbox"/> phpmyadmin	utf8_bin	Check privileges
<input type="checkbox"/> test	latin1_swedish_ci	Check privileges

Total: 5

# Create New table

The screenshot shows the phpMyAdmin interface. On the left sidebar, the 'student' database is selected and circled in red. The main panel shows the 'Filters' section with a search box. Below it, a table list shows 'userdetails' with 1 table and a sum. The 'Create new table' button is highlighted with a red circle. Below this button, there are input fields for 'Table name' and 'Number of columns' (set to 4), and a 'Create' button.

Recent Favorites

New  
information\_schema  
mysql  
performance\_schema  
phpmyadmin  
**student**  
New  
+ userdetails  
test

Filters

Containing the word:

Table	Action
<input type="checkbox"/> userdetails	Browse  Structure  Search  Insert  Empty
1 table	Sum

☐ Check all With selected:

Print Data dictionary

**Create new table**

Table name  Number of columns



Recent Favorites

- New
- information\_schema
- mysql
- performance\_schema
- phpmyadmin
- student
  - New
  - userdetails
- test

[Structure](#) [SQL](#) [Search](#) [Query](#) [Export](#) [Import](#) [Operations](#) [Privileges](#) [Routes](#)Table name: studentdetails Add 1 column(s) 

Name	Type	Length/Values	Default	Collation	Attributes
------	------	---------------	---------	-----------	------------

rollno	INT		None		
--------	-----	--	------	--	--

[Pick from Central Columns](#)

sname	VARCHAR	20	None		
-------	---------	----	------	--	--

[Pick from Central Columns](#)

sfathername	VARCHAR	20	None		
-------------	---------	----	------	--	--

[Pick from Central Columns](#)

scity	VARCHAR	20	None		
-------	---------	----	------	--	--

[Pick from Central Columns](#)

Table comments:

Collation:

Storage Engine:

PARTITION definition:

Partition by: 

( Expression or column list )

Partitions:

# Insert values in the table

The screenshot shows the phpMyAdmin interface for a database named 'student'. The 'Insert' tab is selected for the 'studentdetails' table. The table structure is as follows:

Column	Type	Function	Null	Value
rollno	int(11)			
sname	varchar(20)			
sfathername	varchar(20)			
scity	varchar(20)			

Below the table structure, there is a checkbox for 'Ignore' which is checked. The same table structure is repeated below the checkbox. A 'Go' button is located at the bottom right of the interface, circled in red.

# Insert values through sql query

Server: 127.0.0.1 » Database: student » Table: studentdetails "student personal details"

Browse Structure SQL Search Insert Export Import Privileges Operations Tracking Triggers

Run SQL query/queries on table student.studentdetails: ⓘ

```
1 INSERT INTO `studentdetails` (`rollno`, `sname`, `sfathername`, `scity`) VALUES ('12', 'Nihan', 'Ranjan', 'Noida');
```

SELECT \* SELECT INSERT UPDATE DELETE Clear Format Get auto-saved query


☐ Bind parameters ⓘ


Bookmark this SQL query:


Delimiter ; ☐ Show this query here again ☐ Retain query box ☐ Rollback when finished ☒ Enable foreign key checks Go





# Browse table


 Browse


 Structure


 SQL


 Search

 Insert

 Export

 Import

 Privileges


 Operations

```
SELECT * FROM `studentdetails`
```

☐ Profiling [ [Edit inline](#) ] [ [Edit](#) ] [ [Explain SQL](#) ] [ [Create PHP code](#) ] [ [Refresh](#) ]

☐ Show all | Number of rows:  Filter rows:  Sort by key:

Extra options




rollno


sname


sfathername

scity

☐

 Edit

 Copy


 Delete


12 Nihar


Ranjan

Noida

☐

 Edit


 Copy

 Delete

13 Allu


Arjun


Chennai





☐ Check all

With selected:

 Edit

 Copy

 Delete

 Export

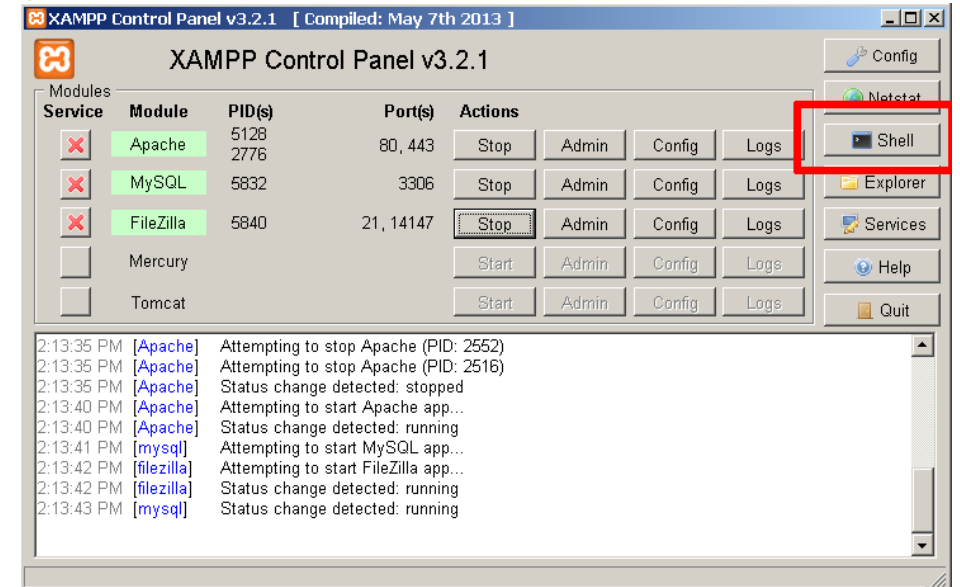
☐ Show all | Number of rows:  Filter rows:  Sort by key:

# Change the MySQL/MariaDB root password

To change the MySQL/MariaDB root password, follow these steps:

- 1.Ensure that the MySQL/MariaDB server is running.
- 2.Open your Windows command prompt by clicking the "Shell" button in the XAMPP control panel.
- 3.Use the mysqladmin command-line utility to alter the MySQL/MariaDB password, using the following syntax:

```
mysqladmin --user=root password  
"newpassword"
```



# Download mysql driver

- <https://dev.mysql.com/downloads/connector/j/>
- Extract the downloaded .zip or .tar.gz file.
- Locate the mysql-connector-java-x.x.x.jar file (where x.x.x is the version).

## MySQL Community Downloads

Connector/J


**General Availability (GA) Releases**Archives

### Connector/J 9.2.0

Select Operating System:

Platform Independent

<b>Platform Independent (Architecture Independent), Compressed TAR Archive</b> (mysql-connector-j-9.2.0.tar.gz)	9.2.0	4.3M	<a href="#">Download</a>
MD5: 7b5193cf541c99111309241886784aa7   <a href="#">Signature</a>			
<b>Platform Independent (Architecture Independent), ZIP Archive</b> (mysql-connector-j-9.2.0.zip)	9.2.0	5.1M	<a href="#">Download</a>
MD5: 1535354d37b937bf792d618f5aafb6ab   <a href="#">Signature</a>			

 We suggest that you use the [MD5 checksums](#) and [GnuPG signatures](#) to verify the integrity of the packages you download.

# Load the driver

Load the MySQL JDBC Driver

```
try {  
    Class.forName("com.mysql.cj.jdbc.Driver");  
} catch (ClassNotFoundException e) {  
    e.printStackTrace();  
}
```

# Establish connection

```
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.SQLException;

public class MySQLConnection {
    public static void main(String[] args) {
        String url = "jdbc:mysql://localhost:3306/your_database"; // Replace with your DB name
        String user = "your_username"; // MySQL username, in this case its root
        String password = "your_password"; // MySQL password, in this case ""

        try {
            Connection conn = DriverManager.getConnection(url, user, password);
            if (conn != null) {
                System.out.println("Connected to MySQL database successfully!");
            }
        } catch (SQLException e) {
            e.printStackTrace();
        }
    }
}
```

# Execute Query

```
import java.sql.*;

public class ExecuteQueryExample {

    public static void main(String[] args) {

        String url = "jdbc:mysql://localhost:3306/your_database";
        String user = "your_username";
        String password = "your_password";

        try {Connection conn = DriverManager.getConnection(url, user, password);

            Statement stmt = conn.createStatement();

            ResultSet rs = stmt.executeQuery("SELECT * FROM sdetails");

            while (rs.next()) {

                int rollno = rs.getInt("rollno");

                String name = rs.getString("sname");

                String fatherName = rs.getString("sfathername");

                String city = rs.getString("scity");

                System.out.println(rollno + " | " + name + " | " + fatherName + " | " + city);

            }

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

        }
```

# Close connection

```
conn.close();
```

```
stmt.close();
```

```
rs.close();
```

# Compile and run

- Assuming that the driver jar file is in the current folder where the .java file is placed.

```
javac -cp ".;mysql-connector-java-x.x.x.jar"  
MySQLJDBCExample.java # Windows
```

Run

```
java -cp ".;mysql-connector-java-x.x.x.jar"  
MySQLJDBCExample # Windows
```



```
30-01-2025 09:59      1,433 JosephusProblem.class
30-01-2025 09:59      1,381 JosephusProblem.java
03-02-2025 10:38        615 MatrixMultiplier.class
03-02-2025 12:48        875 MultiThreadJoinExample.class
03-02-2025 12:47      1,335 MultiThreadJoinExample.java
05-02-2025 20:57    <DIR>      mysql-connector-j-9.2.0
05-02-2025 20:57    2,609,756 mysql-connector-j-9.2.0.jar
05-02-2025 20:54    5,337,769 mysql-connector-j-9.2.0.zip
05-02-2025 21:29        2,303 MySqlConnection.class
05-02-2025 21:11        1,840 MySqlConnection.java
03-02-2025 16:15        723 NetDNS.class
03-02-2025 16:15        276 NetDNS.java
03-02-2025 16:05      1,463 NetGetIPAddress.class
03-02-2025 16:05        410 NetGetIPAddress.java
03-02-2025 10:38      1,744 ParallelMatrixMultiplication.class
03-02-2025 10:38      1,807 ParallelMatrixMultiplication.java
03-02-2025 11:55      1,617 RandomNumberWriter.class
03-02-2025 11:55        921 RandomNumberWriter.java
03-02-2025 11:55        391 random_numbers.txt
21-01-2025 12:52      1,211 ScannerTest.class
21-01-2025 12:52        372 ScannerTest.java
04-02-2025 11:01        511 statictest.class
04-02-2025 11:01        139 statictest.java
23-01-2025 15:04    <DIR>      test
20-01-2025 12:08    <DIR>      test1
03-02-2025 12:48      1,256 WorkerThread.class
      33 File(s)      7,975,562 bytes
      5 Dir(s)  38,404,747,264 bytes free
```

```
D:\javaprgs>javac -cp ".;mysql-connector-j-9.2.0.jar" MySqlConnection.java
```

```
05-02-2025 21:29      2,303 MySqlConnection.class
05-02-2025 21:11      1,840 MySqlConnection.java
03-02-2025 16:15       723 NetDNS.class
03-02-2025 16:15      276 NetDNS.java
03-02-2025 16:05     1,463 NetGetIPAddress.class
03-02-2025 16:05      410 NetGetIPAddress.java
03-02-2025 10:38     1,744 ParallelMatrixMultiplication.class
03-02-2025 10:38     1,807 ParallelMatrixMultiplication.java
03-02-2025 11:55     1,617 RandomNumberWriter.class
03-02-2025 11:55      921 RandomNumberWriter.java
03-02-2025 11:55      391 random_numbers.txt
21-01-2025 12:52     1,211 ScannerTest.class
21-01-2025 12:52      372 ScannerTest.java
04-02-2025 11:01      511 statictest.class
04-02-2025 11:01      139 statictest.java
23-01-2025 15:04      <DIR>      test
20-01-2025 12:08      <DIR>      test1
03-02-2025 12:48     1,256 WorkerThread.class
      33 File(s)      7,975,562 bytes
      5 Dir(s)  38,404,747,264 bytes free
```

```
D:\javaprgs>javac -cp ".;mysql-connector-j-9.2.0.jar" MySqlConnection.java
```

```
D:\javaprgs>java -cp ".;mysql-connector-j-9.2.0.jar" MySqlConnection
```

```
Database connected successfully!
```

```
Student Records:
```

```
12 | Nihar | Ranjan | Noida
```

```
13 | Allu | Arjun | Chennai
```

# Problem

Write a java program to

1. Fetch all the records from the employee table
  2. Search employee with empid
  3. Update specific employee details
  4. Delete a particular employee record
- Employee table has fields in MySQL
    - Empid
    - EmpName
    - EmpCity
    - EmpSalary