

# Java & MySQL

19:43:44 18.05.2012

Dieses Beispiel ist einfach mal zum Testen gedacht. Die MySQL Datenbank wurde einfach aus einem XAMPP Paket heraus verwendet. Das diese richtige funktioniert wurde bereits mehrfach mit PHP getestet. Für PHP gibt es auch entsprechende Testskripte auf <http://www.php.net>, für Java habe ich etwas ähnliches gesucht, aber nicht gefunden. Deshalb habe ich einfach mal schnell selber eine Beispielimplementierung vorgenommen.

Bevor man loslegt braucht man unbedingt den JDBC Treiber von MySQL. Diesen sollte man mittels Google schnell finden und herunterladen können. In dem Paket ist ein Jar-File enthalten, welches als External Jar in das folgende Java Projekt eingebunden werden muss.

Die MySQL Datenbank kann einfach über XAMPP Control gestartet werden und das folgende Skript sollte den Inhalt entsprechend anzeigen. Die TODO müssen entsprechend angepasst werden.

```
[java]
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;

public class HelloMySql {

    /**
     * @param args
     * @throws ClassNotFoundException
     * @throws SQLException
     */
    public static void main(String[] args) throws ClassNotFoundException,
        SQLException {

        // JDBC MySQL Treiber einbinden
        Class.forName("com.mysql.jdbc.Driver");

        // TODO datenbank anpassen
        String url = "jdbc:mysql://localhost:3306/datenbank";

        java.sql.Connection con =
            DriverManager.getConnection(
                url, "root", "");

        System.out.println("URL: " + url);
        System.out.println("Connection: " + con);

        Statement stmt = con.createStatement();

        // TODO tabelle anpassen
        ResultSet rs = stmt.executeQuery("SELECT * FROM tabelle");
```

```
System.out.println("Display all results:");
```

```
while(rs.next()){  
    // TODO spalte anpassen  
    String str = rs.getString("spalte");  
    System.out.println(str);  
}  
}  
}  
[/java]
```