Save Arduino Sensor Data to MYsql Using Processing

by Johnson Selva

Honestly its hard to store **Arduino data to MySQL** directly so that in addiction to **Arduino IDE** i used **Processing IDE** that is similar to Arduino IDE but with lot of different usage and you can able to code it in java.

Note:don't run Arduino serial monitor while running processing code because port conflict will occur as both have to use the same port

You Need:

1. Arduino Uno/Mega or clone
2. Wamp server
3. Processing IDE 2.2.1 (**don't use greater than that**)
4. BezierSQLib-0.2.0 library for processing (**Download link below**)
5. sensor (**I used LDR and LM35 to measure light and temperature**)

https://www.instructabl...

Download

Step 1: Setting Arduino

Burn the below simple demo code to arduino that will act as a sender.,

```
 void setup()
 {
  Serial.begin(9600);
}

 void loop()
{
  int i=0,j=0;
  i=analogRead(A0);
  j=analogRead(A1);
  Serial.print(i);
  Serial.print(",");
  Serial.println(i);
}
```

Step 2: Setting Up the MySQL

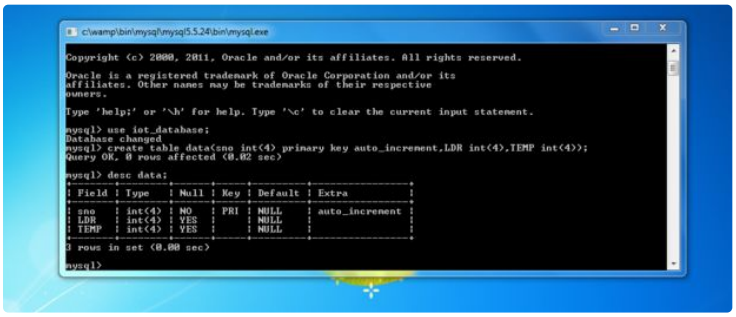
1. Install Wamp server for MySQL and configure it to store data
2. Run wamp server
3. open MySQL console
4. select database
5. Then create the table for your data

```
create table data(sno int(4) primary key auto_increment,LDR int(4),TEMP int(4));
```

use desc your_table_name to display table details

```
desc data;
```

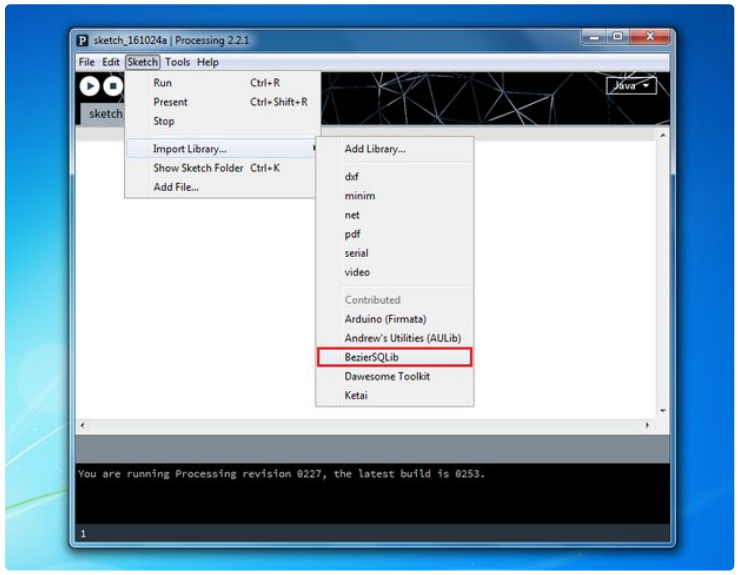
That's all for DB now we can move to processing...



Step 3: Setting Up Processing IDE

1. Download and Install the Processing IDE 2.2.1
2. Extract the above given ZIP to MyDocuments/Processing/Libraries
3. Now open processing IDE and check the library is installed correctly or not as in the above image
4. Then Copy the below code to processing and name it of your own

```
 /*
 ARDUINO TO MYSQL THROUGH PROCESSING
Read Serial messages from Arduino then write it in MySQL.
Author : J.V.JohnsonSelva  September 2016
*/

 import de.bezier.data.sql.*;  //import the MySQL library
import processing.serial.*;   //import the Serial library

 MySQL msql;      //Create MySQL Object
String[] a;
int end = 10;    // the number 10 is ASCII for linefeed (end of serial.println), later we will look for this to break up individual messages
String serial;   // declare a new string called 'serial' . A string is a sequence of characters (data type know as "char")
Serial port;     // The serial port, this is a new instance of the Serial class (an Object)

 void setup() {
  String user     = "root";
  String pass     = "";
  String database = "iot_database";

  msql = new MySQL( this, "localhost", database, user, pass );

  port = new Serial(this, Serial.list()[0], 9600); // initializing the object by assigning a port and baud rate (must match that of Arduino)
  port.clear();  // function from serial library that throws out the first reading, in case we started reading in the middle of a string from Arduino
  serial = port.readStringUntil(end); // function that reads the string from serial port until a println and then assigns string to our string variable (called 'serial')
  serial = null; // initially, the string will be null (empty)
}

 void draw()
{

  while (port.available() > 0)
  {
    //as long as there is data coming from serial port, read it and store it
    serial = port.readStringUntil(end);
  }
    if (serial != null)
    {
      //if the string is not empty, print the following
      //Note: the split function used below is not necessary if sending only a single variable. However, it is useful for parsing (separating) messages when
      //reading from multiple inputs in Arduino. Below is example code for an Arduino sketch

      a = split(serial, ',');  //a new array (called 'a') that stores values into separate cells (separated by commas specified in your Arduino program)
      println(a[0]); //print LDR value
      println(a[1]); //print LM35 value

      function();
    }
}

 void function()
{
  if ( msql.connect() )
    {
        msql.query( "insert into data(LDR,Temp)values("+a[0]+","+a[1]+")" );
    }
    else
    {
        // connection failed !
    }
    msql.close();  //Must close MySQL connection after Execution
}
```



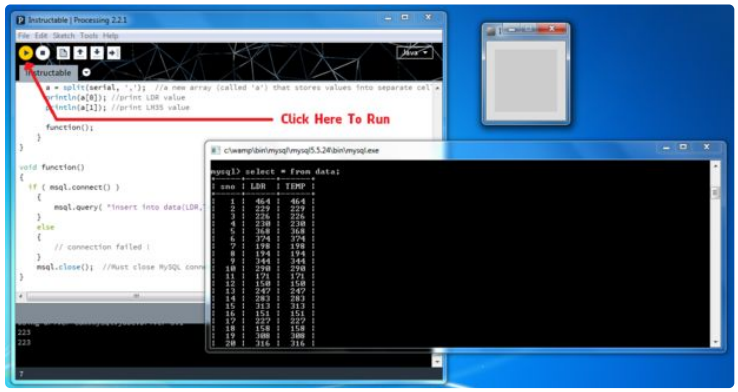
Step 4: Executing the Program.

Run the program by clicking the run button don't close the popup window closing will stop execution and below query to view stored data in MySQL...

```
select * from data;
```

To view number of data inserted use the below query..

```
select count(*) from data;
```



Step 5: Conclusion

I would like to thank you for reading my tutorial. I would appreciate if you found it useful and drop a like (favorite) or ask me anything as it keeps me motivated to do these instructables. feel free to ask any questions that you need to know...

Happy Coding Arduino...

Step 6: Support Our Channel

No library found for de.bezier.data.sql
Libraries must be installed in a folder named 'libraries' inside the 'sketchbook' folder.
javax.net.ssl.SSLException: Received fatal alert: protocol_version
at sun.security.ssl.Alerts.getSSLException(Unknown Source)
at sun.security.ssl.Alerts.getSSLException(Unknown Source)
at sun.security.ssl.SSLSocketImpl.recvAlert(Unknown Source)
at sun.security.ssl.SSLSocketImpl.readRecord(Unknown Source)
at sun.security.ssl.SSLSocketImpl.performInitialHandshake(Unknown Source)
at sun.security.ssl.SSLSocketImpl.startHandshake(Unknown Source)
at sun.security.ssl.SSLSocketImpl.startHandshake(Unknown Source)
at sun.net.www.protocol.https.HttpsClient.afterConnect(Unknown Source)
at sun.net.www.protocol.https.AbstractDelegateHttpsURLConnection.connect(Unknown Source)
at sun.net.www.protocol.https.HttpsURLConnectionImpl.connect(Unknown Source)
at processing.app.contrib.ContributionManager.download(ContributionManager.java:62)
at processing.app.contrib.ContributionManager$1.run(ContributionManager.java:123)
at java.lang.Thread.run(Unknown Source)

what is wrong am try to install lib

Thank you so much, your instruction helped so much. I spent 8 hours looking for something that works. Stumbled on your post, and it worked perfectly. Thank you so much, you helped me find the answer. THANK YOU :)

Glad you are here dude. thanks for your valuable comment.