

JDBC Tutorial

JDBC

- This is a very simple tutorial how to use JDBC that enables communication between Java application and DBMS (We use MySQL here)
- First, you need JDBC Driver for MySQL
 - Download `mysql-connector-java-5.0.8.zip`
 - Upzip the file
 - Add `mysql-connector-java-5.0.8-bin.jar` in your CLASSPATH

Driver Loading

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

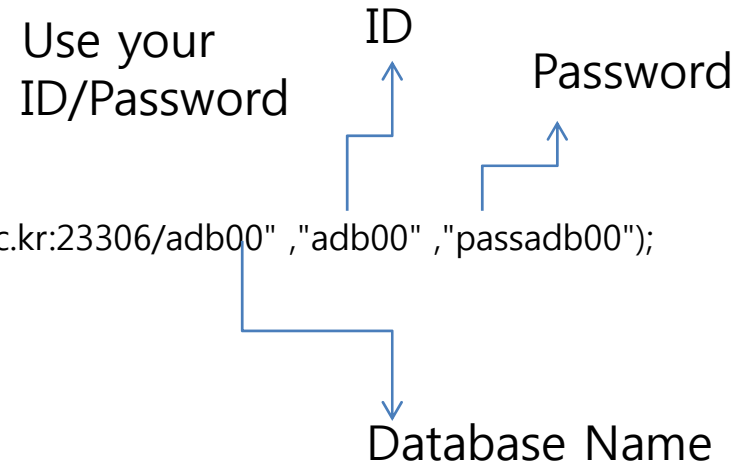
```
public class jdbcTest {  
    public static void main(String[] args) {  
        try {  
            Class.forName("com.mysql.jdbc.Driver").newInstance();  
        } catch (Exception e) {  
            ex.printStackTrace();  
        }  
    }  
}
```

If this code works
correctly without error
messages, driver has
been loaded properly

JDBC Connection

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

```
public class DBConnection {  
    public static void main(String[] args) {  
        try {  
            Class.forName("com.mysql.jdbc.Driver").newInstance();  
        } catch (Exception ex) {  
            ex.printStackTrace();  
        }  
        try{  
            Connection conn =  
                DriverManager.getConnection("jdbc:mysql://vega.snu.ac.kr:23306/adb00" , "adb00" , "passadb00");  
        } catch (SQLException ex) {  
            ex.printStackTrace();  
        }  
    }  
}
```



How to query using JDBC

Driver Loading & Connection

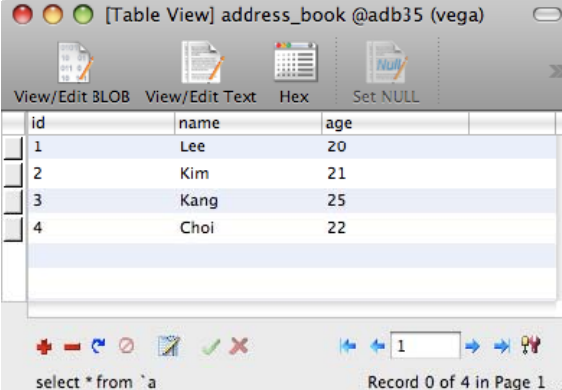
....

```
Statement stmt = null;
ResultSet rs = null;
try{
    stmt = conn.createStatement();
    rs = stmt.executeQuery("SELECT * FROM address_book");
```

```
// or alternatively, if you don't know ahead of time that
// the query will be a SELECT...
```

```
if (stmt.execute("SELECT * FROM address_book")) {
    rs = stmt.getResultSet();
}
} catch(Exception ex) {
    // handle the error
}
```

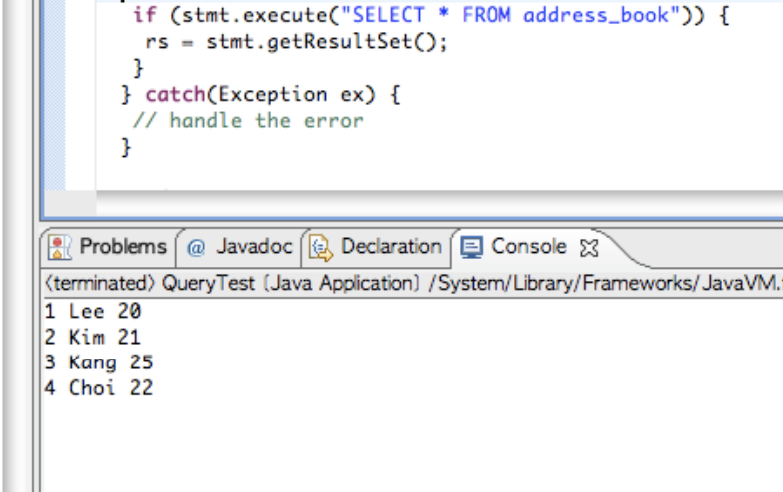
```
try{
    while (rs.next()) {
        int id = rs.getInt("id");
        String name = rs.getString("name");
        int age = rs.getInt("age");
        System.out.println(id + " " + name + " " + age);
    }
} catch(Exception ex) {
    // handle the error
}
}
```



The screenshot shows a database table view titled "[Table View] address_book @adb35 (vega)". The table has three columns: 'id', 'name', and 'age'. The data rows are:

id	name	age
1	Lee	20
2	Kim	21
3	Kang	25
4	Choi	22

At the bottom of the window, there is a toolbar with icons for adding, deleting, and refreshing records. Below the toolbar, the text "select * from `a`" and "Record 0 of 4 in Page 1" is visible.



The screenshot shows an IDE window with a code editor and a console. The code in the editor is:

```
if (stmt.execute("SELECT * FROM address_book")) {
    rs = stmt.getResultSet();
}
} catch(Exception ex) {
    // handle the error
}
```

The console output shows the results of the query:

```
<terminated> QueryTest (Java Application) /System/Library/Frameworks/JavaVM.f
1 Lee 20
2 Kim 21
3 Kang 25
4 Choi 22
```

// executeQuery() -> SELECT

// executeUpdate() -> INSERT, UPDATE, DELETE