



جامعة طرابلس  
كلية تقنية المعلومات  
قسم هندسة البرمجيات



## البرمجة المرئية Visual Programming خريف 2024

المحاضرة التاسعة - الاتصال بقاعدة البيانات  
**Connection to Database**



### مواضيع المحاضرة

- ▶ توصيل JavaFX مع MySQL
- ▶ المكونات الرئيسية ل JDBC
- ▶ إنشاء تطبيق يستخدم JDBC
- ▶ الخطوات اللازمة للحصول على JDBC Connection
- ▶ إنشاء قاعدة بيانات.
- ▶ إنشاء جدول داخل قاعدة البيانات.
- ▶ إدخال بيانات داخل الجدول
- ▶ عرض البيانات من الجدول.
- ▶ تعديل البيانات داخل الجدول.
- ▶ حذف البيانات من الجدول.



## توصيل قاعدة بيانات MySQL مع JavaFX

► قبل العمل:

1- أنت بحاجة إلى تثبيت نظام إدارة قاعدة بيانات MySQL.

<http://dev.mysql.com/downloads/>

2- تنزيل برنامج تشغيل MySQL JDBC.

<http://dev.mysql.com/downloads/connector/j/>

3- إنشاء مشروع JavaFX

► 3



## أولاً: تنزيل MySQL وتثبيتها

► يوجد العديد من الحزم على موقع MySQL في قسم التنزيلات:

► <http://dev.mysql.com/downloads/mysql/>

1- MySQL Server بالنسبة إلى mysql، يمكن اختيار تثبيت واحد مما يلي:

- XAMPP (phpmyadmin) or WAMP.
- MySQL + MySQL Workbench .

2- إنشاء قاعدة البيانات.

► 4



## تحميل XAMPP عن طريق



Web Images News Explore

4,000,000 RESULTS Date Language Region

**XAMPP Installers and Downloads for Apache Friends**

<https://www.apachefriends.org/index.html>

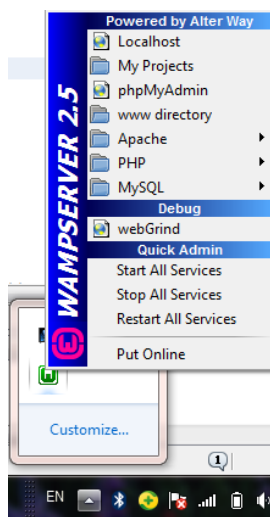
What is XAMPP? XAMPP is the most popular PHP development environment. XAMPP is a completely free, easy to install Apache distribution containing MariaDB, PHP, and Perl.



5



## أو استخدام WAMP



9



## ثانيًا: تنزيل MySQL Connector / j

▶ ثانيًا: تنزيل MySQL Connector / j

The screenshot shows the MySQL website's 'MySQL Connectors' page. The main content area lists several connectors with brief descriptions. A red arrow points to 'Connector/J', which is described as a 'Standardized database driver for Java platforms and development.' The sidebar on the left contains a list of links, with 'MySQL Connectors' being the active selection.



## ثانيًا: تنزيل MySQL Connector / j

▶ قم بتنزيل JDBC وهو اختصار Java DataBase Connectivity من الموقع

▶ <http://www.mysql.com/products/connector>.

▶ ثم حدد "Platform Independent"

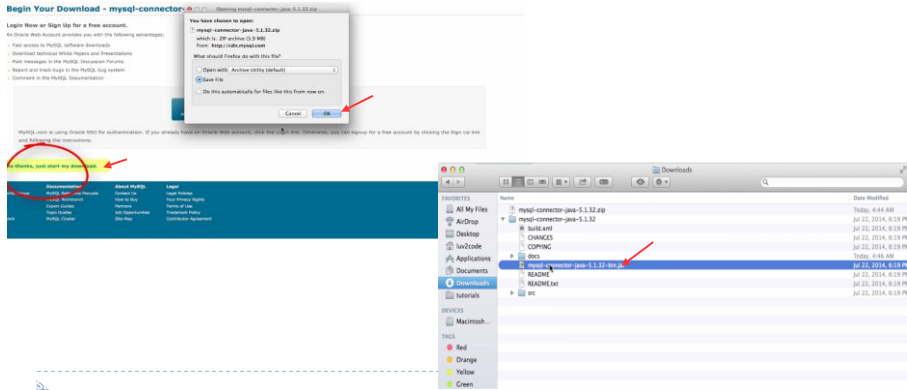
The screenshot displays the download page for MySQL Connector/J 8.0.11. It features a 'Select Operating System:' dropdown menu where 'Platform Independent' is chosen. Below this, two download links are provided: one for a compressed TAR archive and another for a ZIP archive. A red arrow points to the ZIP archive download button. At the bottom, a note advises users to use MD5 checksums and GnuPG signatures to verify the integrity of the downloaded packages.



## ثانيًا: تنزيل MySQL Connector / j

► قم بفك ضغطه. افتح المجلد الجديد الذي تم إنشاؤه. تأكد من وجود ملف `jar` في المجلد، باسم مشابه لـ

► `mysql-connector-java-5.1.34-bin.jar`

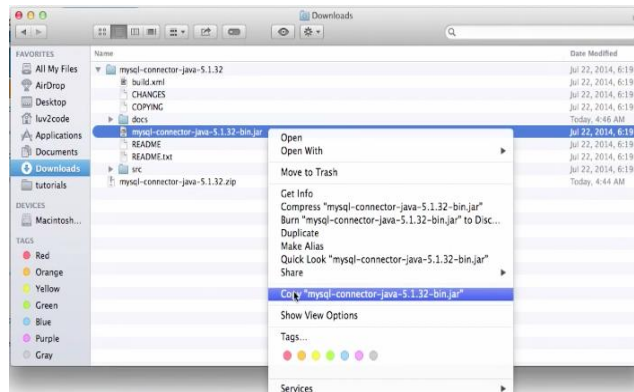


15



## ثانيًا: تنزيل MySQL Connector / j

► تحتاج الآن إلى نسخ الملف إلى دليل مشروع `netbeans`.



16

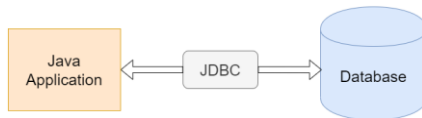
## توصيل قاعدة بيانات MySQL مع JavaFX

- ▶ تستخدم قواعد البيانات العلائقية (Relational Database Systems) في تخزين البيانات والاستفادة منها عن طريق إجراء استعلامات باستخدام لغة الاستعلام الهيكلية (SQL) للحصول على المعلومات المطلوبة.
- ▶ معظم لغات البرمجة توفر إمكانية عمل اتصال بقواعد البيانات المختلفة وإجراء استعلامات عليها باستخدام (SQL). لغة جافا تستخدم JDBC API لإنشاء اتصال بقواعد البيانات والاستعلام منها.

▶ 17

## Java DataBase Connectivity

- ▶ **JDBC** هو اختصار لـ **Java DataBase Connectivity** وهي المسئولة عن عملية الاتصال بقواعد البيانات المختلفة ولغة جافا.
- ▶ مكتبة **JDBC** تحتوي على عدد من **classes** التي تستخدم في إجراء مختلف العمليات الخاصة بالتعامل مع قواعد البيانات حيث توفر الخدمات الآتية:
  1. الاتصال بمصدر البيانات ، مثل قاعدة البيانات.
  2. إنشاء جمل SQL .
  3. تنفيذ جمل SQL .
  4. استرجاع ومعالجة النتائج التي تم تلقيها من قاعدة البيانات.



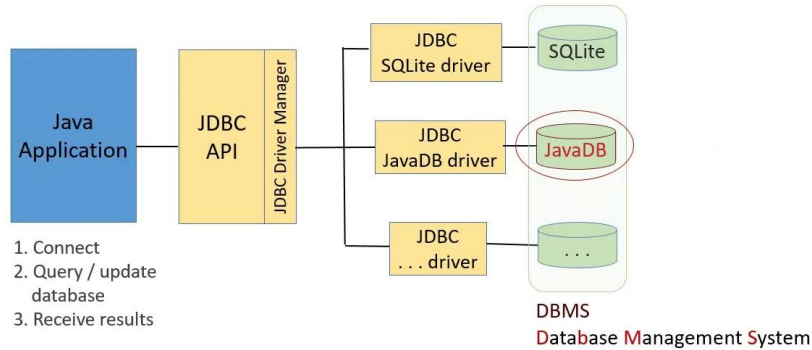
JDBC تدعم قواعد البيانات العلائقية مثل Oracle و Microsoft SQL Server وغيرها.

▶ 18



## JDBC API

### JDBC - Java Database Connectivity



19



## المكونات الرئيسية لـ JDBC

- ▶ **DriverManager** يستخدم مجموعة من database Drivers ويقوم باستعمال المناسب منها في عملية الاتصال.
- ▶ **Drivers** تقوم بعملية الاتصال بخادم قاعدة البيانات.
- ▶ **Connection** يستخدم في إجراء عملية الاتصال بقاعدة البيانات.
- ▶ **Statement** تستخدم في إنشاء جمل SQL .
- ▶ **ResultSet** يحتوي على البيانات المستخلصة من قاعدة البيانات.
- ▶ **SQLException** تقوم بالتعامل مع أي خطأ يظهر في التطبيق المستخدم لقاعدة البيانات.

20



## خطوات إنشاء تطبيق يستخدم JDBC

▶ لإنشاء تطبيق يستخدم JDBC نحتاج لتنفيذ الخطوات التالية:

1. تضمين المكتبة الخاصة ب JDBC في التطبيق.
2. تسجيل JDBC Driver .
3. فتح اتصال مع قاعدة البيانات.
4. تنفيذ جملة الإستعلام.
5. استخلاص البيانات من نتيجة استعلام قاعدة البيانات.
6. إغلاق الاتصال مع قاعدة البيانات.

▶ 21



## خطوات إنشاء تطبيق يستخدم JDBC

▶ لإنشاء تطبيق يستخدم JDBC نحتاج لتنفيذ الخطوات التالية:

1- تضمين المكتبة الخاصة ب JDBC في التطبيق

▶ **import java.sql.\*;**

2- تسجيل JDBC Driver

▶ **Class.forName("com.mysql.jdbc.Driver");**

▶ من الإصدار 4 لا تحتاج إلى استخدام Class.forName()، تقوم  
getConnection بداخل DriverManager باستدعاء driver المناسب

▶ 22





## خطوات إنشاء تطبيق يستخدم JDBC

3- فتح اتصال مع قاعدة البيانات.

```
String URL = "jdbc:mysql://hostname/databaseName";
```

```
String USER = "username";
```

```
String PASS = "password"
```

```
Connection conn = DriverManager.getConnection(URL,  
USER, PASS);
```

▶ 23



## خطوات إنشاء تطبيق يستخدم JDBC

4- تنفيذ جملة الإستعلام.

```
stmt = conn.createStatement();
```

```
String sql;
```

```
sql = "SELECT id, first, last, age FROM Employees";
```

```
ResultSet rs = stmt.executeQuery(sql);
```

▶ 24



## خطوات إنشاء تطبيق يستخدم JDBC

5- استخلاص البيانات من نتيجة استعلام قاعدة البيانات.

```
while (rs.next()) {
    //Retrieve by column name
    int id = rs.getInt("id");
    int age = rs.getInt("age");
    String first = rs.getString("first");
    String last = rs.getString("last");
}
```

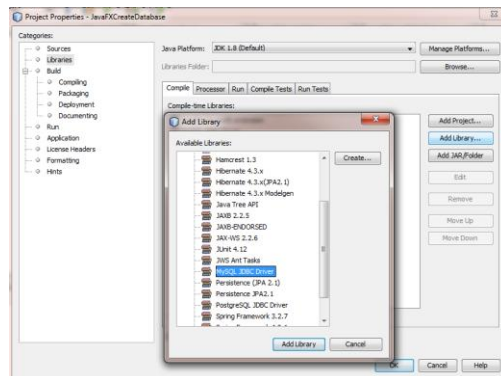
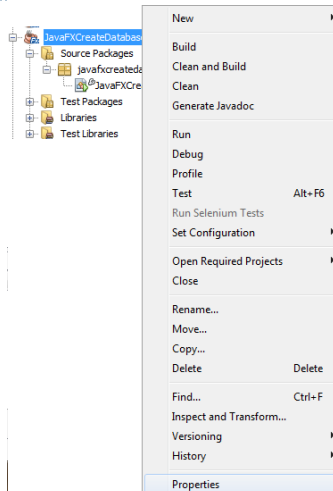
6- إغلاق الاتصال مع قاعدة البيانات.

```
rs.close();
stmt.close();
conn.close();
```

25



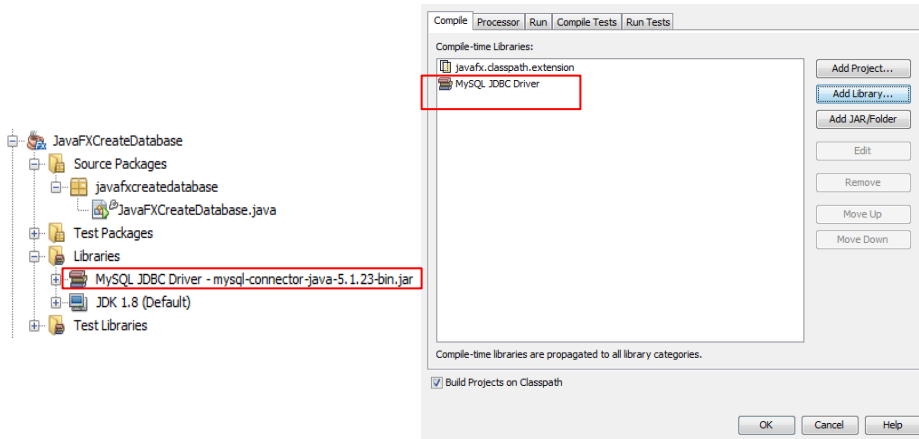
## خطوات ربط JDBC مع JAVA FX



26



## خطوات ربط JDBC مع JAVA FX

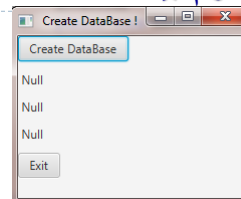


27



## مثال: برنامج يقوم بإنشاء قاعدة بيانات

```
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.SQLException;
import java.sql.Statement;
import javafx.application.Application;
import javafx.application.Platform;
import javafx.event.ActionEvent;
import javafx.event.EventHandler;
import javafx.scene.Scene;
import javafx.scene.control.Button;
import javafx.scene.control.Label;
import javafx.scene.layout.VBox;
import javafx.stage.Stage;
```



```
public class JavaFXCreateDatabase extends Application {

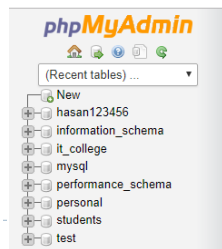
    // JDBC driver name and database URL
    static final String JDBC_DRIVER = "com.mysql.jdbc.Driver";
    static final String DB_URL = "jdbc:mysql://localhost/";
    // Database credentials
    static final String USER = "root";
    static final String PASS = "";
```

```
@Override
public void start(Stage primaryStage) {
    Button btn = new Button();
    Label l1 = new Label(" Null ");
    Label l2 = new Label(" Null ");
    Label l3 = new Label(" Null ");
    Button exit = new Button(" Exit ");

    VBox root = new VBox();
    root.setSpacing(10);

    btn.setText(" Create DataBase ");
```

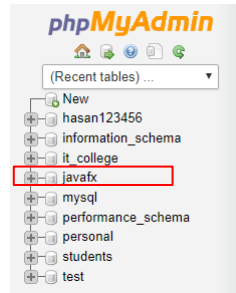
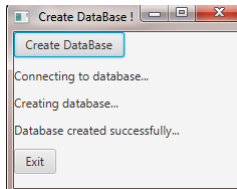
28





## مثال: برنامج يقوم بإنشاء قاعدة بيانات

```
exit.setOnAction((ActionEvent event) -> {
    Platform.exit();
}); //end setOnAction
btn.setOnAction(new EventHandler<ActionEvent>() {
    @Override
    public void handle(ActionEvent event) {
        Connection conn = null;
        Statement stmt = null;
        try {
            //STEP 2: Register JDBC driver
            Class.forName(JDBC_DRIVER);
            //STEP 3: Open a connection
            11.setText("Connecting to database...");
            conn = DriverManager.getConnection(DB_URL, USER, PASS);
            //STEP 4: Execute a query
            12.setText("Creating database...");
            stmt = conn.createStatement();
            String sql = "CREATE DATABASE JAVAFX";
            stmt.executeUpdate(sql);
            13.setText("Database created successfully...");
            //STEP 6: Close resources
            stmt.close();
            conn.close();
        } catch (ClassNotFoundException | SQLException ex) {
            //Handle errors for Class.forName and for JDBC
            System.out.println(ex);
        }
    } //end handle
}); //end setOnAction
```



```
root.getChildren().addAll(btn, 11, 12, 13, exit);
Scene scene = new Scene(root, 300, 250);
primaryStage.setTitle(" Create DataBase ! ");
primaryStage.setScene(scene);
primaryStage.show();
} //end Start

public static void main(String[] args) {
    launch(args);
} //end main
```

29



## مثال: برنامج يقوم بإنشاء جدول

```
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.SQLException;
import java.sql.Statement;
import javafx.application.Application;
import javafx.application.Platform;
import javafx.event.ActionEvent;
import javafx.event.EventHandler;
import javafx.scene.Scene;
import javafx.scene.control.Button;
import javafx.scene.control.Label;
import javafx.scene.layout.VBox;
import javafx.stage.Stage;
```

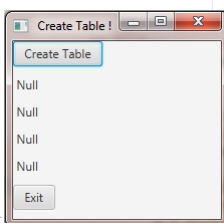
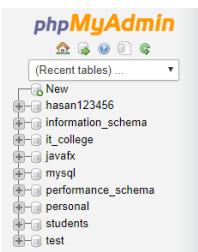
```
public class JavaFXCreateTable extends Application {
    // JDBC driver name and database URL
    static final String JDBC_DRIVER = "com.mysql.jdbc.Driver";
    static final String DB_URL = "jdbc:mysql://localhost/JAVAFX";
    // Database credentials
    static final String USER = "root";
    static final String PASS = "";

    @Override
    public void start(Stage primaryStage) {
        Button btn = new Button();
        Label 11 = new Label(" Null ");
        Label 12 = new Label(" Null ");
        Label 13 = new Label(" Null ");
        Label 14 = new Label(" Null ");
        Button exit = new Button(" Exit ");

        VBox root = new VBox();
        root.setSpacing(10);

        btn.setText(" Create Table ");

        exit.setOnAction((ActionEvent event) -> {
            Platform.exit();
        }); //end setOnAction
```





## مثال: برنامج يقوم بإنشاء جدول

```
btn.setOnAction(new EventHandler<ActionEvent>() {
    @Override
    public void handle(ActionEvent event) {
        Connection conn = null;
        Statement stmt = null;
        try {
            //STEP 2: Register JDBC driver
            Class.forName(JDBC_DRIVER);
            //STEP 3: Open a connection
            11.setText("Connecting to a selected database...");
            conn = DriverManager.getConnection(DB_URL, USER, PASS);
            12.setText("Connected database successfully...");
            //STEP 4: Execute a query
            13.setText("Creating table in given database...");
            stmt = conn.createStatement();
            String sql = "CREATE TABLE STUDENT "
                + "(id INTEGER not NULL, "
                + " first VARCHAR(255), "
                + " last VARCHAR(255), "
                + " age INTEGER, "
                + " PRIMARY KEY (id ))";
            stmt.executeUpdate(sql);
            14.setText("Created table in given database...");
            //STEP 6: Close resources
            stmt.close();
            conn.close();
        } catch (ClassNotFoundException | SQLException ex) {
```

```
        } catch (ClassNotFoundException | SQLException ex) {
            //Handle errors for Class.forName and for JDBC
            System.out.println(ex);
        } //end handle
    } //end setOnAction

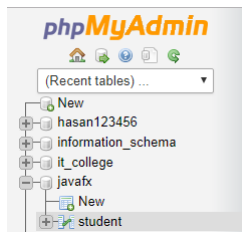
    root.getChildren().addAll(btn, 11, 12, 13, 14, exit);
    Scene scene = new Scene(root, 300, 250);
    primaryStage.setTitle(" Create Table ! ");
    primaryStage.setScene(scene);
    primaryStage.show();
} //end Start

public static void main(String[] args) {
    launch(args);
} //end main
```

31



## مثال: برنامج يقوم بإنشاء جدول



الجدول موجود بعد التنفيذ

Server: mysql wampserver » Database: javafx

Structure SQL Search Query Export Import Operations

No tables found in database.

الجدول غير موجود قبل التنفيذ

Create table

Name:  Number of columns:

---

Server: mysql wampserver » Database: javafx » Table: student

Browse Structure SQL Search Insert Export Import

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
1	id	int(11)			No	None		Change Drop
2	first	varchar(255)	latin1_swedish_ci		Yes	NULL		Change Drop
3	last	varchar(255)	latin1_swedish_ci		Yes	NULL		Change Drop
4	age	int(11)			Yes	NULL		Change Drop

32



## مثال: يقوم بإدخال البيانات في الجدول

```
public class JavaFXInsertTableData extends Application {
```

```
// JDBC driver name and database URL
static final String JDBC_DRIVER = "com.mysql.jdbc.Driver";
static final String DB_URL = "jdbc:mysql://localhost/JAVAFX";
// Database credentials
static final String USER = "root";
static final String PASS = "";
```

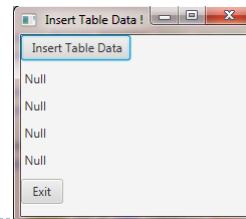
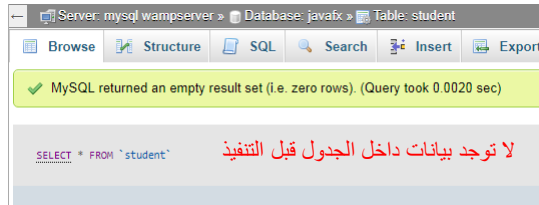
```
@Override
```

```
public void start(Stage primaryStage) {
    Button btn = new Button();
    Label l1 = new Label(" Null ");
    Label l2 = new Label(" Null ");
    Label l3 = new Label(" Null ");
    Label l4 = new Label(" Null ");
    Button exit = new Button(" Exit ");
```

```
VBox root = new VBox();
root.setSpacing(10);
```

```
btn.setText(" Insert Table Data");
```

```
exit.setOnAction( (ActionEvent event) -> {
    Platform.exit();
}); //end setOnAction
```



33



## مثال: يقوم بإدخال البيانات في الجدول

```
btn.setOnAction(new EventHandler<ActionEvent>() {
```

```
@Override
```

```
public void handle(ActionEvent event) {
```

```
    Connection conn = null;
```

```
    Statement stmt = null;
```

```
    try {
```

```
        //STEP 2: Register JDBC driver
```

```
        Class.forName(JDBC_DRIVER);
```

```
        //STEP 3: Open a connection
```

```
        l1.setText("Connecting to a selected database...");
```

```
        conn = DriverManager.getConnection(DB_URL, USER, PASS);
```

```
        l2.setText("Connected database successfully...");
```

```
        //STEP 4: Execute a query
```

```
        l3.setText("Inserting records into the table...");
```

```
        stmt = conn.createStatement();
```

```
        String sql = "INSERT INTO STUDENT VALUES (100, 'Lojin', 'Ali', 18)";
```

```
        stmt.executeUpdate(sql);
```

```
        sql = "INSERT INTO STUDENT VALUES (101, 'Heba', 'SAME', 25)";
```

```
        stmt.executeUpdate(sql);
```

```
        sql = "INSERT INTO STUDENT VALUES (102, 'Akrm', 'Sofian', 30)";
```

```
        stmt.executeUpdate(sql);
```

```
        sql = "INSERT INTO STUDENT VALUES (103, 'Mona', 'Ahmad', 28)";
```

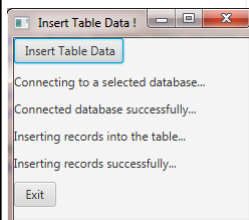
```
        l4.setText("Inserting records successfully...");
```

```
        //STEP 6: Close resources
```

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

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

```
    } catch (ClassNotFoundException | SQLException ex) {
```



34



## مثال: يقوم بإدخال البيانات في الجدول

```

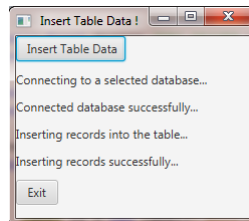
    } catch (ClassNotFoundException | SQLException ex) {
        //Handle errors for Class.forName and for JDBC
        System.out.println(ex);
    }

    //end handle
    //end setOnAction

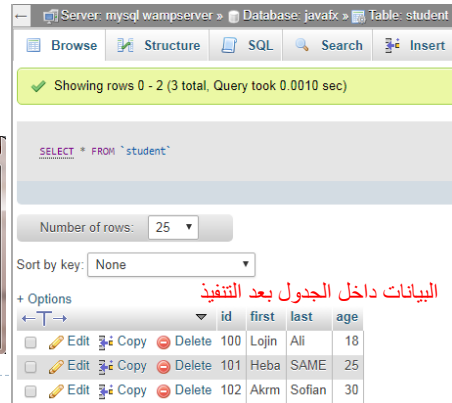
    root.getChildren().addAll(btn, l1, l2, l3, l4, exit);
    Scene scene = new Scene(root, 300, 250);
    primaryStage.setTitle(" Insert Table Data ! ");
    primaryStage.setScene(scene);
    primaryStage.show();
    //end Start

    public static void main(String[] args) {
        launch(args);
    } //end main
}

```



35



## مثال: يقوم بعرض البيانات من الجدول

```

public class JavaFXSelectTableData extends Application {

    // JDBC driver name and database URL
    static final String JDBC_DRIVER = "com.mysql.jdbc.Driver";
    static final String DB_URL = "jdbc:mysql://localhost/JAVAFX";
    // Database credentials
    static final String USER = "root";
    static final String PASS = "";

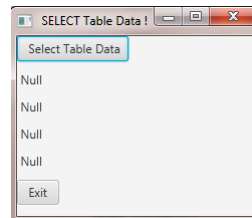
    @Override
    public void start(Stage primaryStage) {
        Button btn = new Button();
        Label l1 = new Label(" Null ");
        Label l2 = new Label(" Null ");
        Label l3 = new Label(" Null ");
        Label l4 = new Label(" Null ");
        Button exit = new Button(" Exit ");

        VBox root = new VBox();
        root.setSpacing(10);

        btn.setText(" Select Table Data");

        exit.setOnAction((ActionEvent event) -> {
            Platform.exit();
        }); //end setOnAction
    }
}

```

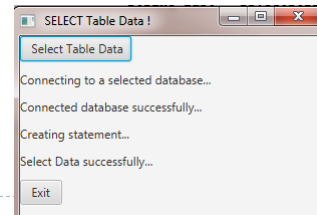


36



## مثال: يقوم بعرض البيانات من الجدول

```
btn.setOnAction(new EventHandler<ActionEvent>() {
    @Override
    public void handle(ActionEvent event) {
        Connection conn = null;
        Statement stmt = null;
        try {
            //STEP 2: Register JDBC driver
            Class.forName(JDBC_DRIVER);
            //STEP 3: Open a connection
            11.setText("Connecting to a selected database...");
            conn = DriverManager.getConnection(DB_URL, USER, PASS);
            12.setText("Connected database successfully...");
            //STEP 4: Execute a query
            13.setText("Creating statement...");
            stmt = conn.createStatement();
            String sql = "SELECT id, first, last, age FROM student";
            ResultSet rs = stmt.executeQuery(sql);
            //STEP 5: Extract data from result set
            while (rs.next()) {
                //Retrieve by column name
                int id = rs.getInt("id");
                int age = rs.getInt("age");
                String first = rs.getString("first");
                String last = rs.getString("last");
```



37

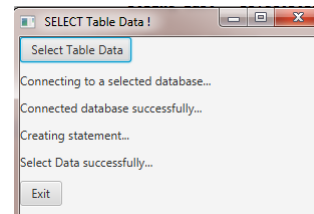


## مثال: يقوم بعرض البيانات من الجدول

```

//Display values
System.out.print("ID: " + id);
System.out.print(", Age: " + age);
System.out.print(", First: " + first);
System.out.println(", Last: " + last);
System.out.println("-----");
}
14.setText("Select Data successfully...");
//STEP 6: Close resources
rs.close();
stmt.close();
conn.close();
} catch (ClassNotFoundException | SQLException ex) {
    //Handle errors for Class.forName and for JDBC
    System.out.println(ex);
} //end handle
} //end setOnAction

root.getChildren().addAll(btn, 11, 12, 13, 14, exit);
Scene scene = new Scene(root, 300, 250);
primaryStage.setTitle(" SELECT Table Data ! ");
primaryStage.setScene(scene);
primaryStage.show();
} //end Start
```



id	first	last	age
100	Lojin	Ali	18
101	Heba	SAME	25
102	Akrm	Sofian	30

Output - JavaFXSelectTableData (jfxsa-run)

```

Executing C:\Users\hassan\Documents\NetBeans\...
ID: 100, Age: 18, First: Lojin, Last: Ali
-----
ID: 101, Age: 25, First: Heba, Last: SAME
-----
ID: 102, Age: 30, First: Akrm, Last: Sofian
-----
```

38





## مثال: يقوم بتعديل البيانات في الجدول

```
public class JavaFXUpdateTableData extends Application {
```

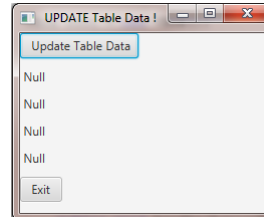
```
// JDBC driver name and database URL
static final String JDBC_DRIVER = "com.mysql.jdbc.Driver";
static final String DB_URL = "jdbc:mysql://localhost/JAVAFX";
// Database credentials
static final String USER = "root";
static final String PASS = "";
```

```
@Override
public void start(Stage primaryStage) {
    Button btn = new Button();
    Label l1 = new Label(" Null ");
    Label l2 = new Label(" Null ");
    Label l3 = new Label(" Null ");
    Label l4 = new Label(" Null ");
    Button exit = new Button(" Exit ");

    VBox root = new VBox();
    root.setSpacing(10);

    btn.setText(" Update Table Data");

    exit.setOnAction((ActionEvent event) -> {
        Platform.exit();
    }); //end setOnAction
}
```



id	first	last	age
100	Lojin	Ali	18
101	Heba	SAME	25
102	Akrm	Sofian	30

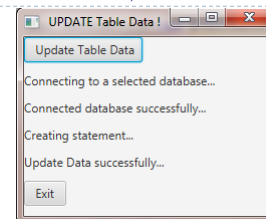
البيانات داخل الجدول قبل التعديل

39



## مثال: يقوم بتعديل البيانات في الجدول

```
btn.setOnAction(new EventHandler<ActionEvent>() {
    @Override
    public void handle(ActionEvent event) {
        Connection conn = null;
        Statement stmt = null;
        try {
            //STEP 2: Register JDBC driver
            Class.forName(JDBC_DRIVER);
            //STEP 3: Open a connection
            l1.setText("Connecting to a selected database...");
            conn = DriverManager.getConnection(DB_URL, USER, PASS);
            l2.setText("Connected database successfully...");
            //STEP 4: Execute a query
            l3.setText("Creating statement...");
            stmt = conn.createStatement();
            String sql = "UPDATE Student SET age = 27 WHERE id in (100, 101)";
            stmt.executeUpdate(sql);
            l4.setText("Update Data successfully...");
            //STEP 6: Close resources
            stmt.close();
            conn.close();
        } catch (ClassNotFoundException | SQLException ex) {
            //Handle errors for Class.forName and for JDBC
            System.out.println(ex);
        }
    }
}); //end handle
}); //end setOnAction
```



id	first	last	age
100	Lojin	Ali	27
101	Heba	SAME	27
102	Akrm	Sofian	30

البيانات داخل الجدول بعد التعديل

```
root.getChildren().addAll(btn, l1, l2, l3, l4, exit);
Scene scene = new Scene(root, 300, 250);
primaryStage.setTitle(" UPDATE Table Data ! ");
primaryStage.setScene(scene);
primaryStage.show();
} //end Start
```

```
public static void main(String[] args) {
    launch(args);
} //end main
```

40



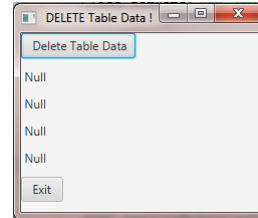
## مثال: يقوم بحذف البيانات من الجدول

```
public class JavaFXDeleteTableData extends Application {

    // JDBC driver name and database URL
    static final String JDBC_DRIVER = "com.mysql.jdbc.Driver";
    static final String DB_URL = "jdbc:mysql://localhost/JAVAFX";
    // Database credentials
    static final String USER = "root";
    static final String PASS = "";

    @Override
    public void start(Stage primaryStage) {
        Button btn = new Button();
        Label l1 = new Label(" Null ");
        Label l2 = new Label(" Null ");
        Label l3 = new Label(" Null ");
        Label l4 = new Label(" Null ");
        Button exit = new Button(" Exit ");
        VBox root = new VBox();
        root.setSpacing(10);
        btn.setText(" Delete Table Data");

        exit.setOnAction((ActionEvent event) -> {
            Platform.exit();
        }); //end setOnAction
    }
}
```



id	first	last	age
100	Lojin	Ali	27
101	Heba	SAME	27
102	Akrm	Sofian	30

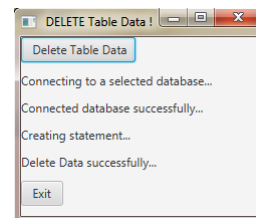
البيانات داخل الجدول قبل الحذف

41



## مثال: يقوم بحذف البيانات من الجدول

```
btn.setOnAction(new EventHandler<ActionEvent>() {
    @Override
    public void handle(ActionEvent event) {
        Connection conn = null;
        Statement stmt = null;
        try {
            //STEP 2: Register JDBC driver
            Class.forName(JDBC_DRIVER);
            //STEP 3: Open a connection
            l1.setText("Connecting to a selected database...");
            conn = DriverManager.getConnection(DB_URL, USER, PASS);
            l2.setText("Connected database successfully...");
            //STEP 4: Execute a query
            l3.setText("Creating statement...");
            stmt = conn.createStatement();
            String sql = "DELETE FROM student WHERE id = 101";
            stmt.executeUpdate(sql);
            l4.setText("Delete Data successfully...");
            //STEP 6: Close resources
            stmt.close();
            conn.close();
        } catch (ClassNotFoundException | SQLException ex) {
            //Handle errors for Class.forName and for JDBC
            System.out.println(ex);
        }
    } //end handle
}); //end setOnAction
```



id	first	last	age
100	Lojin	Ali	27
102	Akrm	Sofian	30

البيانات داخل الجدول بعد الحذف

42



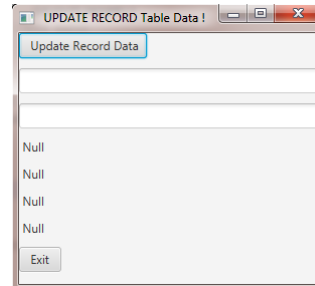
## مثال 2: يقوم بتعديل البيانات في الجدول

```
public class JavaFXUpdateRecordData extends Application {
    // JDBC driver name and database URL
    static final String JDBC_DRIVER = "com.mysql.jdbc.Driver";
    static final String DB_URL = "jdbc:mysql://localhost/JAVAFX";
    // Database credentials
    static final String USER = "root";
    static final String PASS = "";

    @Override
    public void start(Stage primaryStage) {
        Button btn = new Button();
        Label l1 = new Label(" Null ");
        Label l2 = new Label(" Null ");
        Label l3 = new Label(" Null ");
        Label l4 = new Label(" Null ");
        TextField tf = new TextField();
        TextField tf1 = new TextField();

        Button exit = new Button(" Exit ");
        VBox root = new VBox();
        root.setSpacing(10);
        btn.setText(" Update Record Data");

        exit.setOnAction((ActionEvent event) -> {
            Platform.exit();
        }); //end setOnAction
    }
}
```



id	first	last	age
100	Lojin	Ali	27
102	Akrm	Sofian	35

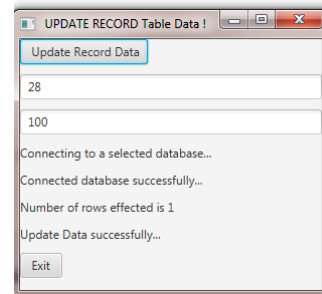
43



## مثال 2: يقوم بتعديل البيانات في الجدول

```
btn.setOnAction(new EventHandler<ActionEvent>() {
    @Override
    public void handle(ActionEvent event) {
        Connection conn = null;
        PreparedStatement pstmt = null;
        try {
            //STEP 2: Register JDBC driver
            Class.forName(JDBC_DRIVER);
            //STEP 3: Open a connection
            l1.setText("Connecting to a selected database...");
            conn = DriverManager.getConnection(DB_URL, USER, PASS);
            l2.setText("Connected database successfully...");
            //STEP 4: Execute a query
            String SQL = "Update student SET age = ? WHERE id = ?";
            pstmt = conn.prepareStatement(SQL);
            pstmt.setInt(1, Integer.parseInt(tf.getText()));
            pstmt.setInt(2, Integer.parseInt(tf1.getText()));
            int rs = pstmt.executeUpdate();
            l3.setText("Number of rows effected is " + rs);
            l4.setText("Update Data successfully...");
            //STEP 6: Close resources
            pstmt.close();
            conn.close();
        } catch (ClassNotFoundException | SQLException ex) {
            //Handle errors for Class.forName and for JDBC
            System.out.println(ex);
        } //end handle
    } //end setOnAction
});

root.getChildren().addAll(btn, tf, tf1, l1, l2, l3, l4, exit);
Scene scene = new Scene(root, 300, 250);
primaryStage.setTitle(" UPDATE RECORD Table Data ! ");
primaryStage.setScene(scene);
primaryStage.show();
} //end Start
```



	id	first	last	age
Delete	100	Lojin	Ali	28
Delete	102	Akrm	Sofian	35

44



### مثال 3: يقوم بإدخال البيانات في الجدول

```
public class JavaFXInsertDataPreparedStatement extends Application {

    // JDBC driver name and database URL
    static final String JDBC_DRIVER = "com.mysql.jdbc.Driver";
    static final String DB_URL = "jdbc:mysql://localhost/JAVAFX";
    // Database credentials
    static final String USER = "root";
    static final String PASS = "";

    @Override
    public void start(Stage primaryStage) {
        Button btn = new Button();
        Label l1 = new Label(" Null ");
        Label l2 = new Label(" Null ");
        Label l3 = new Label(" Null ");
        Label l4 = new Label(" Null ");
        TextField tf1 = new TextField();
        TextField tf2 = new TextField();
        TextField tf3 = new TextField();
        TextField tf4 = new TextField();
        Button exit = new Button(" Exit ");
        VBox root = new VBox();
        root.setSpacing(10);
        btn.setText(" Insert Record Data");
        exit.setOnAction((ActionEvent event) -> {
            Platform.exit();
        }); //end setOnAction
    }
}
```

45



### مثال 3: يقوم بإدخال البيانات في الجدول

	id	first	last	age
	100	Lojin	Ali	28
	101	Hasan	Ali	40
	102	Akrm	Sofian	35

```
btn.setOnAction(new EventHandler<ActionEvent>() {
    @Override
    public void handle(ActionEvent event) {
        Connection conn = null;
        PreparedStatement pstmt = null;
        try {
            //STEP 2: Register JDBC driver
            Class.forName(JDBC_DRIVER);
            //STEP 3: Open a connection
            l1.setText("Connecting to a selected database...");
            conn = DriverManager.getConnection(DB_URL, USER, PASS);
            l2.setText("Connected database successfully...");
            //STEP 4: Execute a query
            String SQL = "INSERT INTO STUDENT VALUES (?, ?, ?, ?)";
            pstmt = conn.prepareStatement(SQL);
            pstmt.setInt(1, Integer.parseInt(tf1.getText()));
            pstmt.setString(2, tf2.getText());
            pstmt.setString(3, tf3.getText());
            pstmt.setInt(4, Integer.valueOf(tf4.getText()));
            int rs = pstmt.executeUpdate();
            l3.setText("Number of rows effected is " + rs);
            l4.setText("Update Data successfully...");
            //STEP 6: Close resources
            pstmt.close();
            conn.close();
        } catch (ClassNotFoundException | SQLException ex) {
            //Handle errors for Class.forName and for JDBC
            System.out.println(ex);
        }
    }
});
```

46



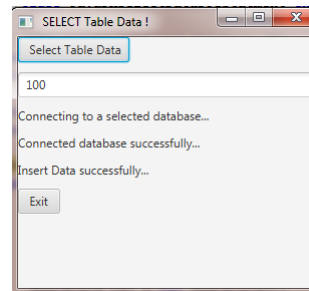
## مثال 4: يقوم بعرض البيانات في الجدول

```
public class JavaFXSelectDataPreStatement extends Application {

    // JDBC driver name and database URL
    static final String JDBC_DRIVER = "com.mysql.jdbc.Driver";
    static final String DB_URL = "jdbc:mysql://localhost/JAVAFX";
    // Database credentials
    static final String USER = "root";
    static final String PASS = "";

    @Override
    public void start(Stage primaryStage) {
        Button btn = new Button();
        Label l1 = new Label(" Null ");
        Label l2 = new Label(" Null ");
        Label l3 = new Label(" Null ");
        Label l4 = new Label(" Null ");
        TextField tf1 = new TextField();
        Button exit = new Button(" Exit ");
        VBox root = new VBox();
        root.setSpacing(10);
        btn.setText(" Select Table Data");

        exit.setOnAction((ActionEvent event) -> {
            Platform.exit();
        }); //end setOnAction
    }
}
```

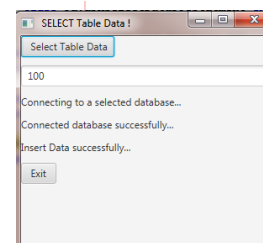


47



## مثال 4: يقوم بعرض البيانات في الجدول

```
btn.setOnAction(new EventHandler<ActionEvent>() {
    @Override
    public void handle(ActionEvent event) {
        Connection conn = null;
        PreparedStatement pstmt = null;
        try {
            //STEP 2: Register JDBC driver
            Class.forName(JDBC_DRIVER);
            //STEP 3: Open a connection
            l1.setText("Connecting to a selected database...");
            conn = DriverManager.getConnection(DB_URL, USER, PASS);
            l2.setText("Connected database successfully...");
            //STEP 4: Execute a query
            String SQL = "SELECT id, first, last, age FROM student where id = ?";
            pstmt = conn.prepareStatement(SQL);
            pstmt.setInt(1, Integer.parseInt(tf1.getText()));
            //rs = stmt.executeQuery(sql);
            ResultSet rs = pstmt.executeQuery();
            l3.setText("Insert Data successfully...");
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
});
```



48



## مثال 4: يقوم بعرض البيانات في الجدول

```
//STEP 5: Extract data from result set
while (rs.next()) {
    //Retrieve by column name
    int id = rs.getInt("id");
    int age = rs.getInt("age");
    String first = rs.getString("first");
    String last = rs.getString("last");
    //Display values
    System.out.print("ID: " + id);
    System.out.print(", Age: " + age);
    System.out.print(", First: " + first);
    System.out.println(", Last: " + last);
    System.out.println("-----");
}
14.setText("Select Data successfully...");
//STEP 6: Close resources
rs.close();
conn.close();
} catch (ClassNotFoundException | SQLException ex) {
    //Handle errors for Class.forName and for JDBC
    System.out.println(ex);
}
} //end handle
} //end setOnAction
```

ID: 100, Age: 28, First: Lojin, Last: Ali

49



## ملخص المحاضرة

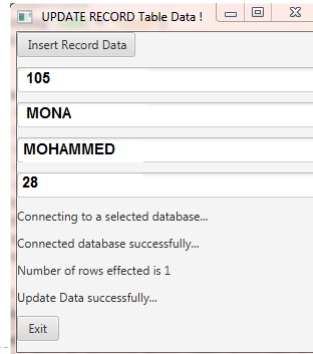
- يعتبر التعامل مع قاعدة البيانات من الأشياء الأساسية داخل لغات البرمجة، في هذه المحاضرة تم الحديث حول كيفية تحميل البرامج الخاصة بقاعدة البيانات وكيفية التعامل معها من داخل لغة JAVA FX.
- تم توضيح كيف يتم إنشاء قاعدة بيانات ثم كيف يتم إنشاء جدول داخل قاعدة البيانات، وكيف يتم إدخال بيانات داخل هذا الجدول وكيف يتم عرضها للمستخدم، ثم تعديل البيانات داخل الجدول وأخير كيف يتم حذف البيانات من الجدول.

50



## تمرين

► اكتب التعليمات البرمجية لتنفيذ الواجهة التالية لإدخال بيانات داخل قاعدة البيانات بإسم JAVAFX والجدول STUDENT.

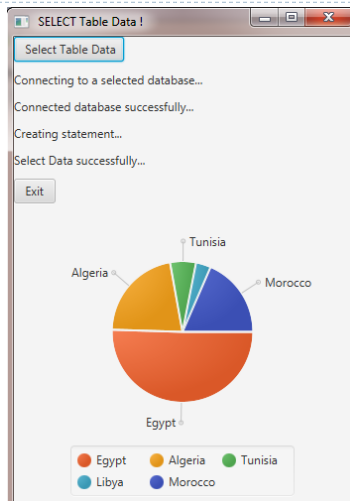


51



## تمرين

► اكتب التعليمات البرمجية لتنفيذ الواجهة التالية لعرض بيانات داخل قاعدة البيانات college والجدول country.



52

---

نهاية المحاضرة

