

اتصال جافا بقواعد البيانات

مقدمة:

توفر لغة الجافا كمثيلاتها من لغات البرمجة عدة طرق لتخزين المعلومات على ملفات مخزنة في الذاكرة الثانوية مثل: الملفات التسلسلية **Sequential files** وملفات الوصول العشوائي **Random access file** و على الرغم من فائدة هاتين الطريقتين في حفظ البيانات، إلا أنهما لا تهتمان بالإمكانيات الكافية للاستعلام عن البيانات بشكل مناسب. إن أنظمة قواعد البيانات لا توفر فقط القدرة على معالجة الملفات، ولكنها تنظم البيانات بطريقة تسمح لها بتنفيذ عمليات استعلام معقدة. إن أشهر أنواع قواعد البيانات المستخدمة حالياً هي قواعد البيانات العلائقية **Relational Database System** وفي هذا النوع من قواعد البيانات تستخدم لغة تسمى لغة الاستعلام الهيكلية **Structured Query Language (SQL)** والتي تستخدم للاستعلام عن البيانات التي تحقق شرطاً معيناً في قاعدة البيانات. وللإستفادة من قواعد البيانات هذه فإن معظم لغات البرمجة ومنها جافا توفر إمكانية عمل اتصال بقواعد البيانات المختلفة وكتابة جمل **SQL** للاستعلام عن البيانات. ومن أشهر قواعد البيانات المستخدمة ما يلي: **Microsoft Access, Sybase, Oracle, Informix, Microsoft SQL Server** وغيرها. في هذا البحث سنتعرف على كيفية كتابة برنامج يتصل بقاعدة بيانات **Microsoft Access** آخذين بعين الاعتبار إن لدى القارئ المهاماً بلغة **SQL** ومفاهيم قاعدة البيانات.

لكي تتمكن من ربط كود الجافا بقاعدة بيانات اكسس هناك طريقتين

الطريقة الأولى

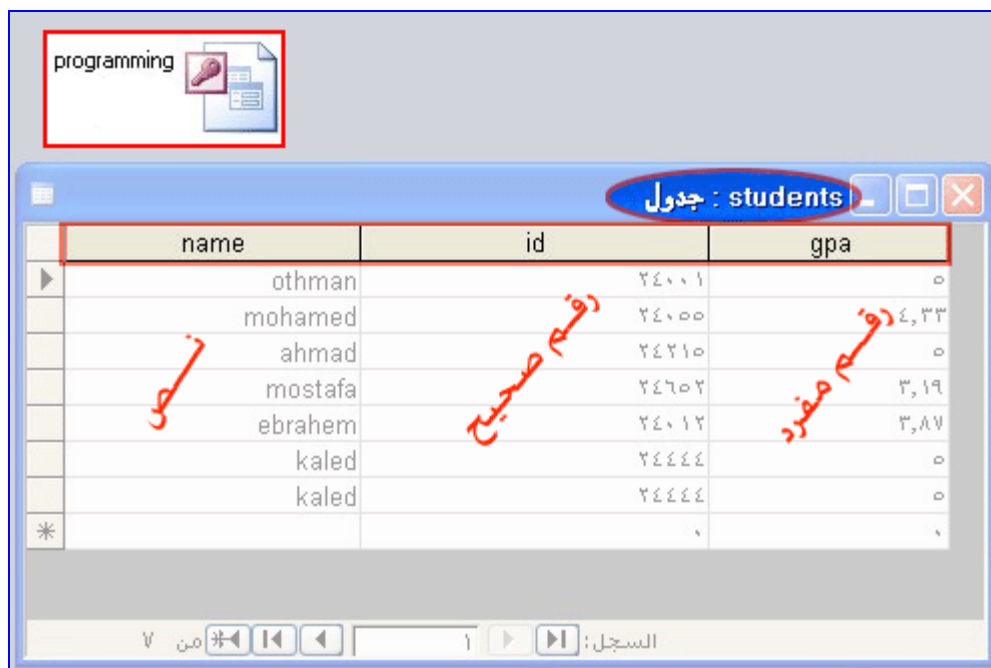
وهي الأفضل لقواعد البيانات الصغيرة، حيث أن المطلوب فقط هو إرفاق ملف قاعدة البيانات وملفات الجافا في مجلد واحد. وهذه الطريقة هي ما سنستخدمها في مثالنا الحالي.

الطريقة الثانية

وهي تسجيل قاعدة البيانات كمصدر بيانات في مصدر قواعد البيانات المفتوح **ODBC** ، و سنتناول كيفية التسجيل، وذكر الفرق الوحيد بين الطريقتين في نهاية البحث.

مثال توضيحي:

في البداية سنقوم بإنشاء قاعدة بيانات باسم **programming**
ثم ننشئ جدول في هذه القاعدة باسم **students**
ويتكون هذا الجدول من ٣ أعمدة:
العمود الأول باسم **name** ونوع البيانات نص.
العمود الثاني باسم **id** ونوع البيانات رقم (صحيح).
العمود الثالث باسم **gpa** ونوع البيانات رقم (مفرد).



name	id	gpa
othman	٢٤٠٠١	٥
mohamed	٢٤٠٥٥	٤,٣٣
ahmad	٢٤٢١٥	٥
mostafa	٢٤٦٥٢	٣,١٩
ebrahem	٢٤٠١٢	٣,٨٧
kaled	٢٤٤٤٤	٥
kaled	٢٤٤٤٤	٥
*		

في الخطوة اللاحقة سنقوم بإجراء عمليات **الاستعلام** على قاعدة البيانات
و **إضافة** بيانات و **تعديل** و **حذف** سجل ومن ثم **إغلاق** هذه القاعدة.

صفء الاتصال

```
1 import java.sql.*;
2 class ConnectDB
3 {
4     Connection conn;
5     Statement stat;
6     public ConnectDB() throws ClassNotFoundException, SQLException
7     {
8         Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
9         conn=DriverManager.getConnection("jdbc:odbc:Driver="+
10             "{Microsoft Access Driver (*.mdb)} ; DBQ=programming.mdb");
11         stat = conn.createStatement();
12     }
13
14     public void select() throws SQLException
15     {
16         String str = "SELECT * FROM students";
17         ResultSet rs = stat.executeQuery( str);
18         while( rs.next())
19         {
20             System.out.print( rs.getString(1)+"\t\t");
21             System.out.print( rs.getInt(2)  +"\t\t");
22             System.out.print( rs.getDouble(3));
23             System.out.println();
24         }
25     }
26
27     public void insert(String _name, int _id, double _gpa)
28         throws SQLException
29     {
30         String str= "INSERT INTO students (name, id, gpa)"+
31             " values( '"+_name+"', '"+_id+", '"+_gpa+"')";
32         stat.executeUpdate(str);
33     }
34
35     public void update( String _name, double _gpa)
36         throws SQLException
37     {
38         String str = "UPDATE students SET gpa="+_gpa+
39             " WHERE name='"+_name+"'";
40         stat.executeUpdate(str);
41     }
42
43     public void delete( String _name) throws SQLException
44     {
45         String str="DELETE FROM students WHERE name='"+_name+"' ";
46         stat.executeUpdate(str);
47     }
48
49     public void close() throws SQLException
50     {
51         stat.close();
52         conn.close();
53     }
54 }
```

صنف التنفيذ

```
1 class TestDB{
2
3     public static void main( String [] args){
4
5         try{
6             ConnectDB db = new ConnectDB();
7
8             db.select();
9
10            db.insert("kalefah", 24111, 3.55);
11
12            db.update("ahmad", 5);
13
14            db.delete("naser");
15
16            db.close();
17        }
18        catch( Exception e){ //ClassNotFoundException, SQLException
19            e.printStackTrace();
20        }
21    }
22 }
```

في الصفحة التالية خطوات إنشاء صنف يتصل بقاعدة البيانات وستتضمن
توضيح لصنف الاتصال السابق

خطوات إنشاء صنف يتصل بقاعدة البيانات

١- استيراد حزمة `sql` من مكتبة الجافا

```
3 import java.sql.*;
```

٢ - تعريف كائنين

أحدهما من نوع `Connection` ، وظيفته إجراء الاتصال بين جافا وقاعدة البيانات ،
و الآخر كائن من النوع `Statement` ، وظيفته إرسال جمل `SQL` لقاعدة البيانات.

```
7 Connection conn;  
8 Statement stat;
```

٣ - إنشاء منشأ (مشيد) لصنف يحتوي على أوامر التشغيل و الاتصال بقاعدة البيانات:

تتكون أوامر التشغيل و الاتصال من ثلاث خطوات وهي :

الاول: استدعاء الطريقة `forName` لتحميل تعريف الفصيلة `Class` التي تحتوي على **مشغل** قاعدة البيانات ،
و تحميل هذه الفصيلة قد يتسبب في حدوث استثناء في حالة أن الفصيلة المطلوبة لا يمكن تحميلها
وهذا الاستثناء هو `java.lang.ClassNotFoundException`.

الثاني: استدعاء الطريقة `getConnection` الموجودة في الفصيلة `DriverManager` من أجل **الاتصال**
بقاعدة البيانات ، و في حالة عدم القدرة على الاتصال بقاعدة البيانات سيحدث استثناء ،
وهذا الاستثناء هو `java.sql.SQLException`.

الثالث: استدعاء الطريقة `createStatement` لإيجاد كائن `Statement` والذي سيستخدم
للاستعلام من قاعدة البيانات.

```
10 public ConnectDB () throws ClassNotFoundException, SQLException  
11 {  
12     Class.forName ( " sun.jdbc.odbc.JdbcOdbcDriver" ) ;  
13  
14     conn=DriverManager.getConnection("jdbc:odbc:Driver="+  
15         "{Microsoft Access Driver (*.mdb)} ; DBQ=programming.mdb" );  
16  
17     stat = conn.createStatement ();  
18 }
```

ملاحظة: يجب أن تكون قاعدة البيانات و ملفات الجافا في مجلد واحد. وإلا يجب عليك تحديد مسار قاعدة البيانات.

إن جميع جمل SQL قد تتسبب في حدوث استثناء من النوع `java.sql.SQLException` لذلك سنعالج هذا الأمر بإضافة `throws SQLException` قبل بداية كل طريقة تحتوي على أحد أوامر SQL.

في مثالنا هذا اسمينا الطرق المستخدمة بـ `select, insert, update, delete` وهذه المهسميات اختيارية.

أولاً: الاستعلام

خطوات تنفيذ استعلام من قاعدة البيانات و عرض نتائج هذا الاستعلام:

١. حفظ جملة الاستعلام في متغير من النوع `String`.
٢. تنفيذ الاستعلام من خلال استدعاء الطريقة `executeQuery` الموجود في كائن الصنف `Statement`.
٣. إنشاء كائن من النوع `ResultSet` وإسناد إليه البيانات الناتجة من عملية الاستعلام.

وظيفة هذا الكائن هي الوصول إلى الصفوف الناتجة عن الاستعلام.

ويحتوي أيضا على الطريقة `next` التي تحرك المؤشر إلى السجل التالي.

ويحتوي أيضا على الطرق `getString` و `getInt` و `getDouble` ... الخ، حيث يتم توزيع هذه الطرق على حسب نوع محتويات الأعمدة في قاعدة البيانات و رقم العمود، حيث يبدأ الترقيم من الواحد.

```
20 public void select() throws SQLException
21 {
22     String str = "SELECT * FROM students";
23     ResultSet rs = stat.executeQuery( str);
24     while( rs.next())
25     {
26         System.out.print( rs.getString(1)+"\t\t");
27         System.out.print( rs.getInt(2)+"\t\t");
28         System.out.print( rs.getDouble(3));
29         System.out.println();
30     }
31 }
```

ثانياً: الإضافة

لإضافة سجل إلى قاعدة بيانات يتوجب عليك القيام بخطوتين فقط

١. حفظ جملة **SQL** لإضافة سجل في متغير نصي.
٢. إرسال هذا المتغير لقاعدة البيانات، باستدعاء طريقة التعديل **executeUpdate** الموجودة في كائن **Statement**.

في هذا المثال أنشأنا لعملية الإضافة طريقة وأسميناها (**insert**) تستقبل ثلاث متغيرات تكون متجانسة مع المتغيرات في أعمدة جدول قاعدة البيانات،
و كما هو معلوم أن أمر **SQL** لإضافة سجل في جدول **students** هو كالتالي:

```
INSERT INTO students (name, id, gpa) values ('_name', _id, _gpa);
```

علما بأن الفاصلة المنقوطة من أمر الإضافة ليست ذات أهمية في شفرة الجافا.
تستقبل هذه الطريقة (**insert**) البيانات الجديدة، ثم تقوم بحفظها في المتغير **str** لتكوين أمر الإضافة.
بعد ذلك يرسل المتغير **str** إلى قاعدة البيانات لإجراء عملية الإضافة.

```
33 public void insert(String _name, int _id, double _gpa)
34     throws SQLException
35 {
36     String str= "INSERT INTO students (name, id, gpa)"+
37                 " values( '"+_name+"', '"+_id+"', '"+_gpa+"')";
38
39     stat.executeUpdate(str);
40 }
```

ثالثاً: التعديل

لا تختلف خطوات التعديل عن خطوات الإضافة، فقط استبدل أمر الإضافة بأمر تعديل.

١. حفظ جملة **SQL** للتعديل على سجل في متغير نصي.

٢. إرسال هذا المتغير لقاعدة البيانات، باستدعاء طريقة التعديل **executeUpdate** الموجودة في كائن **Statement**.

جملة التعديل على المعدل التراكمي **gpa** لـ أحد الطلاب سيكون كالتالي:

```
UPDATE students SET gpa=_gpa WHERE name='_name'
```

وكما ذكرنا سابقاً يمكنك الاستغناء عن إضافة الفاصلة المنقوطة.

```
44 public void update( String _name, double _gpa)
45                     throws SQLException
46 {
47     String str = "UPDATE students SET gpa="+_gpa+
48                 " WHERE name='"+_name+"'";
49
50     stat.executeUpdate(str);
51 }
```

رابعاً: الحذف

كذلك لا يوجد اختلاف في عملية الحذف عن العمليات السابقة.
سنكتفي بكتابة أمر الحذف و طريقة الحذف في المثال هذا.

أمر إجراء عملية حذف حقل (صف) احد الطلاب هو كالتالي:

```
DELETE FROM students WHERE name='_name'
```

```
57 public void delete( String _name) throws SQLException
58 {
59     String str="DELETE FROM students WHERE name = '"+_name+"' ";
60
61     stat.executeUpdate(str);
62 }
```

هـ - إغلاق قاعدة البيانات

- إغلاق كائن Statement .
- إغلاق كائن Connection .

في هذا المثال أنشأنا طريقة باسم **close** تمكنا من غلق الكائنين بمجرد استدعائهما.

```
67 public void close() throws SQLException
68 {
69     stat.close();
70     conn.close();
71 }
```

عندما تكون قاعدة البيانات مسجلة كمصدر بيانات مفتوح، فينبغي عليه

استبدال الأمر التالي:

```
14 conn=DriverManager.getConnection("jdbc:odbc:Driver="+
15 "{Microsoft Access Driver (*.mdb)} ; DBQ=programming.mdb");
```

بأحد هذين الأمرين

١. إذا كانت قاعدة البيانات لا تحتوي على كلمة مرور

```
14 conn=DriverManager.getConnection("jdbc:odbc:programming");
```

٢. إذا كانت قاعدة البيانات تحتوي على اسم مستخدم وكلمة مرور:

```
14 conn=DriverManager.getConnection("jdbc:odbc:programming",
15 "scott","tiger");
16 // user ,passowrd;
```

تسجيل قاعدة بيانات كمصدر بيانات في مصدر قواعد البيانات المفتوح ODBC

للقيام بهذا العمل علينا إتباع الخطوات التالية :

١. اذهب إلى لوحة التحكم **☛** أدوات إدارية **☛** مصادر البيانات ODBC.
٢. انقر على زر **Add**.
٣. حدد نوع قاعدة البيانات، ولتكن كما في هذا المثال **Microsoft Access Driver (*.mdb)**.
٤. انقر على زر **finish**.
٥. اكتب اسم قاعدة البيانات.
٦. انقر على زر تحديد.
٧. حدد موقع قاعدة البيانات.
٨. أضغط على قاعدة البيانات عند ظهورها في الجهة اليمنى من صندوق الحوار.
٩. انقر الزر موافق.
١٠. في حالة رغبتك لإنشاء اسم مستخدم و كلمة مرور لقاعدة البيانات.
 - i. انقر على زر خيارات متقدمة.
 - ii. ادخل اسم تسجيل الدخول.
 - iii. و كلمة المرور.
 - iv. ثم انقر على الزر موافق .
 ١١. ثم انقر على الزر موافق **☛** موافق.

للاستزادة

راجع برمجة ٣

إعداد

عثمان الفحطاني

محمد العزمان

بإشراف الأستاذ محمد عاشور