

Java – S09

Alin Zamfiroiu

S09 – SQLite

- ▶ Se descarca SQLite de pe site-ul oficial:
<http://sqlite.org/download.html>

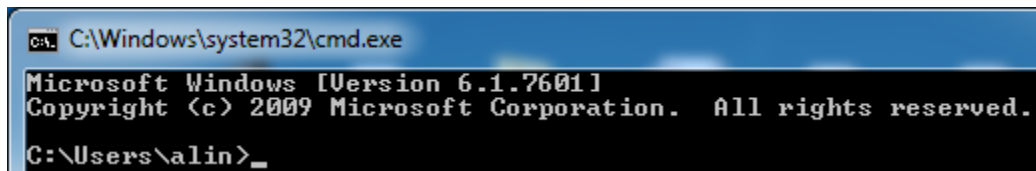
Precompiled Binaries for Windows

sqlite-dll-win32-x86-3120100.zip (422.24 KiB)	32-bit DLL (x86) for SQLite version 3.12.1. (sha1: 0d2793554207d094a0683fd4d6d712d9616fb84c)
sqlite-dll-win64-x64-3120100.zip (701.11 KiB)	64-bit DLL (x64) for SQLite version 3.12.1. (sha1: e53540d4b94dcfd9bfe94c75f60d687463957592)
sqlite-tools-win32-x86-3120100.zip (1.51 MiB)	A bundle of command-line tools for managing SQLite database files, including the sqlite3_analyzer.exe program. (sha1: 442122c559fa3480d2f60f80469d53943ecb35be)

- ▶ Se copiaza intr-un director si se memoreaza adresa acestuia.

S09 – SQLite

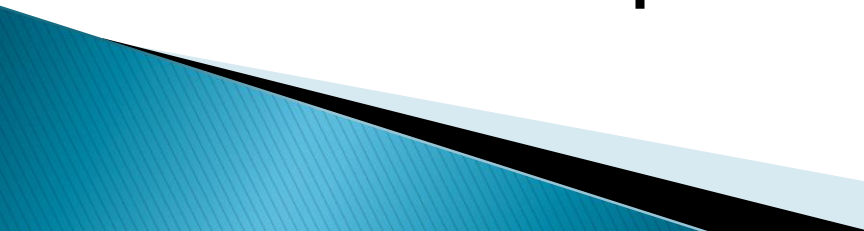
- ▶ Se deschide Command Line de la Windows.



```
C:\Windows\system32\cmd.exe
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.
C:\Users\alin>
```

- ▶ Ne mutam in directorul unde am copiat fisierul sqlite3 prin comanda `cd`.
- ▶ Pentru crearea/utilizarea unei baze de date se foloseste comanda: `sqlite3 numeDB.db`.

S09 – SQLite

- ▶ Pentru vizualizarea schemei bazei de date se foloseste comanda: “.schema”, iar pentru vizualizarea unei singure tabelle: “.schema tableName”.
 - ▶ Pentru vizualizarea tabelelor se foloseste comanda “.tables”.
 - ▶ Pentru parasirea bazei de date se foloseste comanda: “.quit” sau “.exit”.
- 

S09 – SQLite

▶ Comenzi:

- **create** table tableName(column1 type [primary key], column2 type,...);
- **insert** into tableName(column1 ,column2) values (value1 , value2);
- **select** * from tableName;

S09 – SQLite

▶ Tipuri de date:

- **TEXT**: character, varchar, varying character, nchar, native character, nvarchar, text;
- **BLOB** – BLOB, fara tip specificat;
- **INTEGER** – int, integer, tinyint, smallint, mediumint, bigint, decimal(10,5), boolean, date, datetime;
- **REAL** – real, double, double precision, float.

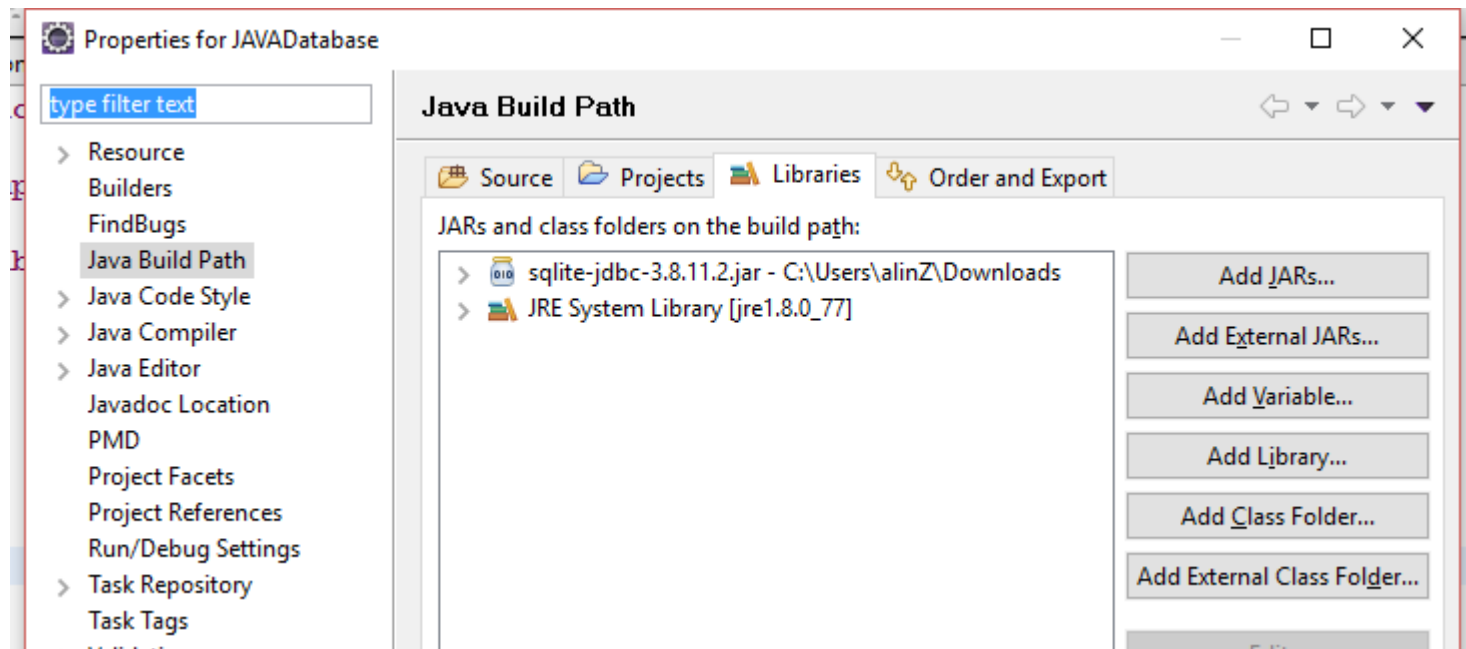
S09 – SQLite

- ▶ In cadrul proiectului Java se realizeaza o clasa care se ocupa de conexiunea la baza de date:

```
public class SQLiteConnection {  
    Connection conn;  
  
    public static Connection conectare(){  
        try{  
            Class.forName("org.sqlite.JDBC");  
            Connection conn=DriverManager.getConnection("jdbc:sqlite:C:  
            System.out.println("succes");  
            return conn;  
        }  
        catch(Exception ert){  
            System.out.println(ert.getMessage());  
            return null;  
        }  
    }  
}
```

S09 – SQLite

- ▶ Se descarca si se adauga la referinte sqlite JDBC.



S09 – SQLite

- ▶ In cadrul programului principal se creaza o conexiune si se realizeaza conectarea prin intermediul metodei din clasa creata anterior.
- ▶ Se creaza un query de interogare a bazei de date;
- ▶ Se initializeaza un obiect de tipul PreparedStatement. Conexiunea realizata are functia preparedStatement() care primeste ca parametru query-ul creat.
- ▶ Se executa query-ul. Rezultat este un obiect de tipul ResultSet

S09 – SQLite

```
public class Program {  
  
    static Connection conn=null;  
  
    public static void main(String[] args) {  
        conn=SQLiteConnection.conectare();  
        String query="select * from utilizatori where nume=?";  
        try {  
            PreparedStatement pst = conn.prepareStatement(query);  
            pst.setString(1, "utilizator");  
            ResultSet rs=pst.executeQuery();  
  
            while(rs.next()){  
                System.out.println(rs.getString("parola"));  
            }  
        } catch (SQLException e) {  
            // TODO Auto-generated catch block  
            e.printStackTrace();  
        }  
        try{  
            conn.close();  
        }catch (Exception ert){  
  
        }  
  
    }  
  
}
```

S09 – SQLite

- ▶ Sa se realizeze o interfata in JAVA FX, astfel incat cand este selectat un utilizator dintr-un combobox sa ii fie afisata parola acestuia.

