Core Java

17. Database Programming with JDBC

Database

- Traditionally referring to relational databases
- VS Flat Files: concurrency, reliability, scalability, accessibility and availability
- A variety of Databases: Oracle, Microsoft SQL Server, Sybase, IBM DB2, MySQL, PostgreSQL
- Uses a table (relational model)

Table

- Tables refer to entities as well as relations
- Columns are attributes
- Rows are instances
- Joining relation tables to make sense
- Types and Constraints on attributes
- Constraints on instances
- Structured Query Language: Declarative

SQL

- SQL is a functional, declarative programming language
- Tells database to select some data from it, or update/insert/delete new data into it
- Used extensively in JDBC, as opposed to keying in objects (Berkeley DB Java Edition, Persistence APIs like JPA)
- Makes it simpler for the programmer if he knows the database

JDBC

- Java Database Connectivity
- Separates Interface (Vendor-neutral) from the underlying Implementation
- Version 4.0 (backward compatible)
- JDBC drivers are the underlying implementations
- They all use *DriverManager.registerDriver()* in static blocks to register themselves

Basic Steps

- Register driver with *Class.forName()*
- Each driver tells the *DriverManager* if it can service a particular type of URL or not
- Create a *Connection* object with a URL and appropriate *Properties*
- Create a *Statement/PreparedStatement/CallableStatement*
- Execute: *getUpdateCount()* or *ResultSet*

ResultSet

- Cursor pointing to a particular row
- Can sometimes only go *next()*
- Updatable, very useful
- Can get meta-information with *ResultSetMetaData*
- Primarily for select statements; Usually used iterate over the *ResultSet*

Connection Pooling

- *Connection*s need to be pooled for high availability
- Time involved in having to use *DriverManager.getConnection* everytime
- Control over number of *Connection* objects
- Use *DataSource* provided with server or application
- References should be *null*–cleaned properly

Embedded JDBC

- How to ship applications with a mini-database
- Example: A Contact Book application
- Embedded databases: Only limited accessibility and availability
- Derby (Cloudscape), HSQLDB
- Written in Java, very portable, free!
- A galore of features, tiny footprint

Demonstration

- Compile and Execute a few programs

Questions?