

استفاده از SQL-Server در برنامه‌ها



استاد: دکتر فرزین ذاکر

درس: آزمایشگاه پایگاه داده

صالح جعفریزاده - ۹۲۲ ۱۳۰۶۱

علی اصغر تقی زاده - ۹۲۲ ۱۳۰۵۵

روند ارائه



- مروری بر مفاهیم Database Connection & Connection String
- نصب و راه اندازی JDBC
- ساخت Connection
- Transaction
- Connection Pool
 - C3P0
 - ORM
 - مثال
 - بررسی
- Hibernate

Database Connection & Connection String



- Database Connection : ارتباط کلاینت با سرور پایگاه داده که کلاینت به وسیلهی آن دستورات را به سرور ارسال و پاسخ ها را دریافت می کند.
- Connection ها به وسیلهی driverهای عرضه شده توسط شرکتها و با استفاده از Connection String به وجود می آیند.
 - ODBC / C,C++
 - JDBC/Java
- از Connection Stringها برای آدرس دهی database یا سروری خاص و برای فراهم کردن اعتبار سنجی کاربران استفاده می شود.

نصب و راه اندازی JDBC



- دانلود JDBC مرتبط
- اضافه کردن JDBC به IntelliJ

ساخت Connection

- حالت کلی Connection String به صورت زیر است.

`jdbc:sqlserver://[serverName[\instanceName][:portNumber]][;property=value[;property=value]]`

Properties •

مثال :

- `jdbc:sqlserver://localhost;user=MyUserName;password=*****;(user name and password)`
- `jdbc:sqlserver://localhost;integratedSecurity=true;(integrated authentication)`
- `jdbc:sqlserver://localhost:1433;databaseName=AdventureWorks;integratedSecurity=true;`
- `jdbc:sqlserver://localhost;instanceName=instance1;integratedSecurity=true;(sql instances)`

• اضافه کردن `sqljdbc_auth.dll` برای `integrated authentication`

`-Djava.library.path= ~\Microsoft JDBC Driver 4.0 for SQL Server\sqljdbc_<version>\enu\auth\x64`

• ایجاد Connection String در برنامه انواع مختلف

- اتصال با استفاده از IPv6 (در وقت کلاس نیست)

Reading & Writing data



- مثال اول: Statement / ResultSet /ExecuteQuery
- مثال دوم: SQLServerPreparedStatement
- مثال سوم: ExecuteUpadte(modify database object)
- مثال چهارم: ExecuteUpadte(modify Data)
- مثال پنجم: CallableStatement(Return Status)
- مثال ششم: CallableStatement(Update Count)

Transaction



- مثال هفتم: `setAutoCommit / Commit / rollback`
- مثال هشتم: `Savepoints`
- `XA Transaction`

Connection Pool



- Database connection ها محدود هستند و وقت قابل توجهی براس ایجاد آنها صرف میشود.
- از این رو به صرفه نیست که برنامه‌ها برای هر بار استفاده از پایگاه داده یک اتصال به آن ایجاد کنند.
- Connection Pool روشی است که با استفاده از آن این مشکل حل شده است، بدین صورت که Connection های ایجاد شد در آن ذخیره می‌شوند و هر بار که برنامه‌ای خواست از آن Connection استفاده کند از Connction های موجود در Pool برمیدارد و زمانی که استفاده از آن تمام شد Connection را به Pool برمیگرداند.

Connection Pool



- برای استفاده از Connection Pool میتوان از wrapper ها استفاده کرد
- Wrapper ها ریزه‌کاری‌های لایه زیرین را از کاربر مخفی میکنند و کاربر میتواند به صورت عادی Connection ها را ایجاد کند و از آنها استفاده کند
- با استفاده از این روش بهتر است به محض اتمام کاری با Connection آنها را ببندند
- Min Pool Size
- زمانی که سیستم idle است Connection Pool شروع به ساخت Connection می‌کند تا به این مقدار برسد.
- پیش فرض: ۰
- Max Pool Size
- Acquisition Timeout
- acquireIncrement
- پیش فرض ۱

C3P0



- Jdbc تنظیمات Connection Pool را به عهده خود جاوا گذاشته با استفاده از wrapper های موجود برای java میتوان Connection Pool را تنظیم کرد.

```
ComboPooledDataSource cpds = new ComboPooledDataSource();  
cpds.setDriverClass( "org.postgresql.Driver" );  
cpds.setJdbcUrl( "jdbc:postgresql://localhost/testdb" );  
cpds.setUser( "swaldman" );  
cpds.setPassword( "test-password" );  
cpds.setMinPoolSize(5);  
cpds.setAcquireIncrement(5);  
cpds.setMaxPoolSize(20);
```

ORM



- Object-Relational mapping یک تکنیک برنامه نویسی است که برای تبدیل داده های نا سازگار بین دو نوع سیستم به کار میرود.
- کنترل داده ها در زبان های برنامه نویسی شیء‌گرا از طریق اشیاء است که اکثراً ساختار غیر ساده دارند.
- در پایگاه های داده رابطه ای تنها مقادیر ساده (اعداد، رشته و ...) را میتوان ذخیره کرد.
- چگونه اشیای تولید شده را ذخیره کنیم؟
- باید منطق اشیاء را تبدیل به فرمی تبدیل کنیم که قابلیت ذخیره شدن در پایگاه داده را دارد و ویژگی های شیء مانند روابطی که با اشیای دیگر دارد را از بین نبریم. کاری که ORM انجام میدهد.

• بدون استفاده از ORM

```
String sql = "SELECT ... FROM persons WHERE id = 10";
DbCommand cmd = new DbCommand(connection, sql);
Result res = cmd.Execute();
String name = res[0]["FIRST_NAME"];
```

• با استفاده از ORM

```
Person p = repository.GetPerson(10);
String name = p.getFirstName();
```

یا

```
Person p = Person.Get(10);
```

بررسی

مزایا

کاهش کد نویسی و کاهش باگ

معایب

بهینه نبودن پایگاه داده

چالش‌ها

ارث بری، هویت، روابط، و..

ORM	زبان برنامه نویسی
LiteSQL, ODB, QXORM	C++
Hibernate, Enterprise JavaBeans, ...	Java
NHibernate, DatabaseObjects,Net
CakePHP, CodeIgniter	PHP

Hibernate



• مزایا:

- نگاشت با استفاده از فایل ها xml
- رابط استفاده ساده
- عدم نیاز به سرور
- و ...

• پایگاه داده هایی که پشتیبانی می کند:

HSQL Database Engine, DB2/NT, MySQL, PostgreSQL, FrontBase, Oracle, Microsoft SQL Server Database, Sybase SQL Server, Informix Dynamic Server

"هیچ گرانی، بی حکمت نیست!"

پسر دکتر شریعتی به پسرش موقع خرید SQL Server

با تشکر از همراهیتان!

