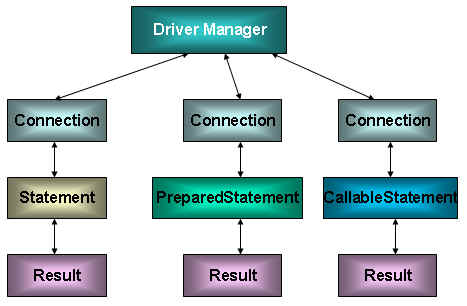


1. **JDBC API**

JDBC API是一系列的接口，它使得应用程序能够**进行数据库连接**，**执行SQL语句**，并且**得到返回结果**。



**1.1 Driver接口**

* Java.sql.Driver接口是所有JDBC驱动程序需要实现的接口。这个接口是提供给数据厂商使用的，不同的数据库厂商提供不同的实现。
* 在程序中不需要直接去访问实现了Driver接口的类，而是**由驱动程序管理器类（java.sql.DriverManager）去调用这些Driver实现**。

1. **加载与注册JDBC驱动**

* 加载JDBC驱动需调用Class类的静态方法forName()，向其传递要加载的JDBC驱动的类名。

Class.*forName*(driverClass);

* DriverManager类是驱动程序管理类，负责管理驱动程序。
* 通常不用显示调用DriverManager类的registerDriver()方法来注册驱动程序类的实例，因为Driver接口的驱动程序类都包含了静态代码块，在这个静态代码块中，会调用DriverManager.registerDriver()方法来注册自身的一个实例。

建立连接

1. **建立连接**

* 可以**调用DriverManager类的getConnection()方法建立到数据库的连接**。
* JDBC URL用于标识一个被注册的驱动程序，驱动程序管理器通过这个URL选择正确的驱动程序，从而建立到数据库的连接。
* JDBC URL的标准由三部分组成，各部分间用冒号分隔。
* **jdbc**：<**子协议**>:<**子名称**>
* **协议**：JDBC URL中的协议总是jdbc
* **子协议**：子协议用于标识一个数据库驱动程序
* **子名称**：一种标识数据库的方法。子名称可以依不同的子协议而变化，用子名称的目的是为了定位数据库提供足够的信息。
* 几种常用数据库的JDBC URL
* Oracle：jdbc:oracle:thin:@localhost:1521:test
* SQLServer：jdbc:microsoft:sqlserver//localhost:1433;DatabaseName=test
* MYSQL：jdbc:mysql://localhost:3306/text

@Test  
 **public void** testGetConnection2() **throws** Exception {  
 System.***out***.println(getConnection2());  
 }  
  
 **public** Connection getConnection2() **throws** Exception {  
 *// 1.准备连接数据库的4个字符串  
 // 1）创建properties对象* Properties properties = **new** Properties();  
  
 *// 2）获取jdbc.properties 对应的输入流* InputStream in = **this**.getClass().getClassLoader().getResourceAsStream(**"jdbc.properties"**);  
  
 *// 3）加载2）对应的输入流* properties.load(in);  
  
 *// 4）具体决定4个字符串* String driver = properties.getProperty(**"driver"**);  
 String jdbcUrl = properties.getProperty(**"jdbcUrl"**);  
 String user = properties.getProperty(**"user"**);  
 String password = properties.getProperty(**"password"**);  
  
 *// 2.加载数据库数据库驱动* Class.*forName*(driver);  
 *// 3.通过DriverManager的getConnection()方法获取数据库连接* **return** DriverManager.*getConnection*(jdbcUrl, user, password);  
 }  
  
 */\*\*  
 \* DriverManager是驱动的管理类  
 \* 1）可以通过重载的getConnection()方法获取数据库连接。较为方便  
 \* 2）可以同时管理多个驱动程序：若注册多个数据库连接，则调用getConnection()方法时传入了参数不同，即返回不同的数据库连接。  
 \*/* @Test  
 **public void** testDriverManager() **throws** Exception {  
 *// 1.准备连接数据库的4个字符串：* String driverClass = **"com.mysql.jdbc.Driver"**;String jdbcUrl = **"jdbc:mysql:///test"**;String user = **"root"**;String password = **"mysql"**; *// 2.加载数据库驱动程序（对应的Driver实现类中有注册驱动的静态代码块）* Class.*forName*(driverClass);  
  
 *// 3.通过DriverManager的getConnection()方法获取数据库连接* Connection connection = DriverManager.*getConnection*(jdbcUrl, user, password);  
  
 System.***out***.println(connection);  
 }