

第 2 章 客户端工具

Oracle 和 SQL Server 附带的客户端工具用于操作和管理数据库。依界面类型分为命令行和图形界面两种。专业技术人员、特别是数据库管理员一般倾向于使用命令行工具。安装数据库服务器软件时，客户端工具一般会默认安装。精通某种数据库产品，除了对其系统原理深刻理解外，还意味着精通其各种客户端工具的使用方法。

本章主要包括：

- 客户端工具概述
- 启动数据库服务
- 命令行工具使用方法
- 图形界面工具使用方法

2.1 概述

Oracle 和 SQL Server 各自有一套工具集，分别包括命令行和图形界面两种类型，命令行工具也称为字符界面工具。本节对两个产品的主要客户端工具做一总体介绍。除了这些产品自带的工具外，Oracle 常用的工具还包括第三方的 TOAD 和 PL/SQL Developer，两者都运行于 Windows 系统。

2.1.1 Oracle 的客户端工具

Oracle 数据库的客户端工具包括：

- SQL*Plus
- SQL Developer
- Oracle Enterprise Manager Database Express
- Oracle Enterprise Manager Cloud Control

SQL*Plus 为命令行工具，是 Oracle 数据库的主要客户端工具，用于执行 SQL 命令、编写存储过程，以及其他各种数据库管理任务。SQL*Plus 在 1985 年出现在 Oracle 5 中，以替换之前的 UFI(User Friendly Interface)。

SQL Developer 是用 Java 语言编写的图形界面工具，可以看作是图形版本的 SQL*Plus，2006 年出现在 Oracle 10g 中，面向对象主要为数据库应用开发人员，用于浏览、创建、修改或删除各种数据库对象，执行 SQL 命令或脚本文件，编辑和调试 PL/SQL 程序，导出数据，生成报表等常见任务。除 Oracle 外，SQL Developer 也可以操作其他常见数据库，如 MySQL、SQL Server、Access、Sybase、DB2 等。

Oracle Enterprise Manager Database Express(简称 EM Express)及 Oracle Enterprise Manager Cloud Control 使用浏览器管理数据库，主要面向数据库管理员使用，这两个工具替换了之前版本的 Oracle Enterprise Manager Database Control。EM Express 包含了 Cloud control 中的关键性能监控页面和基本管理页面，用法较为简单，不需要过多配置，本书只介绍其使用方法。

2.1.2 SQL Server 的客户端工具

SQL Server 的客户端工具包括：

- sqlcmd
- SQL Server Management Studio

sqlcmd 是命令行工具，对应于 Oracle 的 SQL*Plus，在 SQL Server 2005 加入，替换之前的 isql 和 osql 工具。

SQL Server Management Studio 是图形界面管理工具，简称为 SSMS，功能强大，简单易用。SSMS 可以看作是 Oracle 的 OEM 与 SQL Developer 两者功能的结合。此工具在 SQL Server 2005 之前称为 Enterprise Manager。

虽然现在的 Oracle Enterprise Manager 与 SSMS 在界面及功能方面有很大区别，但 Oracle 推出 Enterprise Manager 的初衷即是对其之模仿，以争夺 SQL Server 用户，Oracle 9i 推出的初始版本与 SSMS 非常相似，Oracle Enterprise Manager 的名称也直接取自当时的 SQL Server。

2.2 启动数据库服务

服务器与客户端的关系类似于百货商场与其顾客的关系。顾客能在商场买到商品，首先要保证商场已经开门营业。与 Access 等文件型 PC 数据库不同，Oracle 和 SQL Server 都是大型数据库，要先启动数据库服务器，才能使用。

2.2.1 启动 Oracle 数据库服务

在 Windows 系统，Oracle 的每个数据库对应一个 Windows 服务，名称为 OracleServiceSID，这里的 SID 为相应数据库实例名称。启动 Oracle 数据库服务，包括两个步骤：启动 Windows 的 OracleServiceSID 服务及启动数据库。

启动 OracleServiceSID 服务，可使用 net start 命令，也可使用图形界面的 Windows 服务管理器。下面使用 net start 命令启动 OracleServiceLAW 服务，law 是数据库的对应实例名称。

```
C:\>net start oracleserviceLAW
OracleServiceLAW 服务正在启动 .....
OracleServiceLAW 服务已经启动成功。
```

若使用 Windows 服务管理器，则如下图所示：

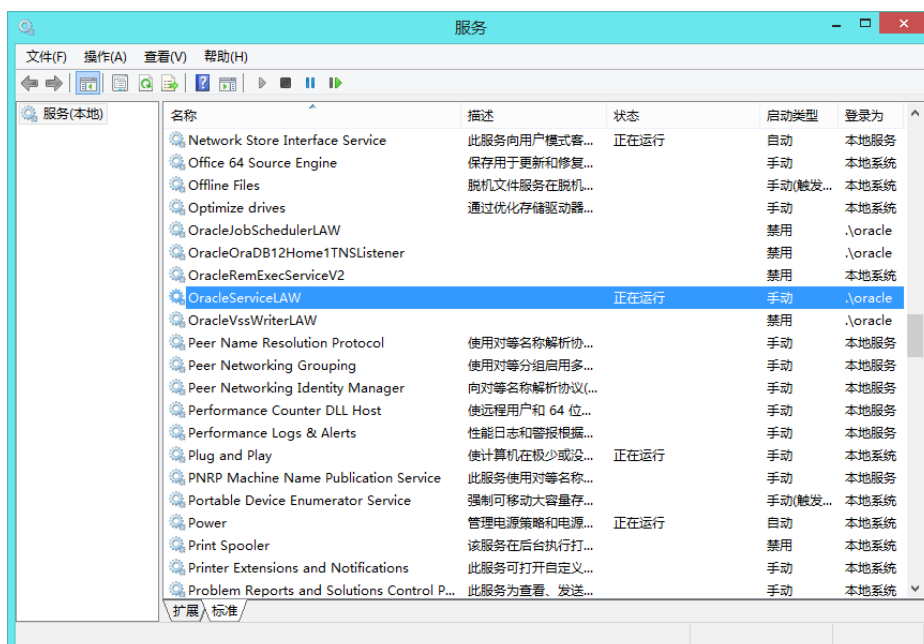


图2-1 启动Oracle数据库服务

启动数据库时，需要以 sys 用户登录：

```
C:\>sqlplus / as sysdba
SQL*Plus: Release 12.1.0.1.0 Production on 星期三 12月 21 20:51:21 2015
Copyright (c) 1982, 2013, Oracle. All rights reserved.
已连接到空闲例程。
```

登录后，执行 startup 命令启动数据库：

```
SQL> startup
ORACLE 例程已经启动。
Total System Global Area 1670221824 bytes
Fixed Size 2403352 bytes
Variable Size 1006633960 bytes
Database Buffers 654311424 bytes
Redo Buffers 6873088 bytes
数据库装载完毕。
数据库已经打开。
```

在 Windows 系统，若使用 dbca 工具创建数据库，则默认会设置数据库随其服务自动启动，一般不需要上面启动数据库这个步骤。在 UNIX/Linux 系统，只需启动数据库，不存在与 Windows 系统的 OracleServiceSID 服务对应的服务。

2.2.2 启动 SQL Server 数据库服务

使用 SQL Server，只需启动 SQL Server 服务器即可，不需要再另外启动数据库。启动服务后，服务器上的数据库就都可以使用了。

启动 SQL Server 数据库服务的方法与启动 Oracle 数据库服务相似，也是使用 net start 命令和服务管理器。

下面执行 net start 命令启动 SQL Server 数据库服务器，这里假定采用默认实例安装，其对应的 Windows 服务名称为 mssqlserver。

```
C:\>net start mssqlserver
SQL Server (MSSQLSERVER) 服务正在启动 .
SQL Server (MSSQLSERVER) 服务已经启动成功。
```

SQL Server 服务器实例上可以创建多个数据库，启动 mssqlserver 服务后，这些数据库就都可以使用了，而不用再次启动数据库，这是与 Oracle 数据库的不同之处，每个 Oracle 数据库在操作之前都需要单独启动，而且在 Windows 系统上，每个数据库都对应操作系统上一个单独的服务。

使用 Windows 的服务管理器启动 SQL Server 服务器如下图所示：

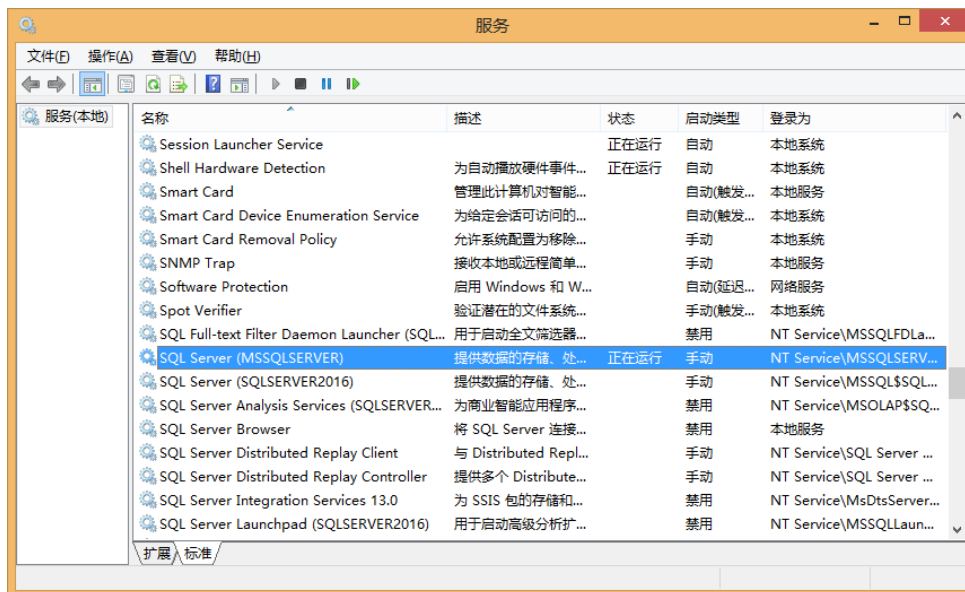


图2-2 启动SQL Server服务

2.3 命令行工具的使用方法

命令行工具以交互方式执行命令，用户可以根据执行结果决定下一条执行什么命令，执行过程中不涉及使用鼠标、点击菜单等操作，效率高、速度快，适合系统管理或数据库管理工作。另外，命令行工具一般总是可用的，而图形工具却不一定，比如使用基于命令行的SSH 工具远程连接至服务器进行系统管理的情况。从事数据库相关专业技术性工作，要熟练掌握常见的命令行工具。本节介绍两种产品命令行工具的使用方法和常见操作。

2.3.1 启动客户端工具并连接至数据库

启动 Oracle 的 SQL*Plus，只需在 Windows 的 cmd 工具中输入以下命令：

```
C:\>sqlplus /nolog
SQL*Plus: Release 12.1.0.1.0 Production on 星期三 12月 23 20:51:21 2015
Copyright (c) 1982, 2013, Oracle. All rights reserved.
SQL>
```

在 SQL 提示符后输入 connect 命令(可简写为 conn)，并输入数据库用户及其口令,以“/”符号分隔，以连接到数据库，这里连接的数据库是客户端工具所在服务器上的本地数据库，然后可以在 SQL 提示符后输入 SQL 命令：

```
SQL> connect scott/tiger
已连接。
SQL>
```

也可以在启动 SQL*Plus 时输入用户名及其口令：

```
C:\>sqlplus scott/tiger
SQL*Plus: Release 12.1.0.1.0 Production on 星期三 12月 23 20:56:49 2015
Copyright (c) 1982, 2013, Oracle. All rights reserved.
```

```
上次成功登录时间: 星期二 12月 22 2015 22:16:33 +08:00
```

```
连接到:
Oracle Database 12c Enterprise Edition Release 12.1.0.1.0 - 64bit Production
With the Partitioning, OLAP, Advanced Analytics and Real Application Testing options
SQL>
```

启动 SQL Server 的 sqlcmd 并连接到服务器，只需在命令提示符中，输入 sqlcmd 回车，以操作系统验证方式连接本地服务器，如果出现“1>”提示符，则说明正常连接：

```
C:\>sqlcmd
1>
```

能够使用操作系统验证方式连接服务器的前提是当前登录 Windows 的帐号已经被映射为 SQL Server 登录帐号，或者当前登录 Windows 的帐号是安装 SQL Server 的 Windows 帐号（这个帐号会自动映射为 SQL Server 帐号，而且具备 SQL Server 服务器管理员权限），这种方式类似于在 SQL*Plus 中以 sys 用户的操作系统验证方式连接数据库，即：

```
C:\> sqlplus / as sysdba
```

若要以 SQL Server 验证方式连接服务器，可以使用 -U 参数指定服务器登录帐号名称，使用 -P 参数指定口令，如下面命令所示：

```
C:\>sqlcmd -U law -P testpassword
1>
```

连接远程服务器或本地服务器的非默认实例，则要附加 -S 参数：

```
C:\>sqlcmd -S law_x240\instance1 -U law -P testpassword
```

x240 是服务器的机器名称，instance1 是非默认实例的实例名称。

若连接远程默认实例，在 -S 后面指定远程 SQL Server 服务器的机器名称即可。

若省略 -S 参数，则连接本地服务器的默认实例，如果省略 -U 参数，则默认使用 Windows 验证方式连接服务器。

以 SQL*Plus 连接远程 Oracle 数据库服务器，相对 SQL Server 来说要复杂一些，需要在客户端配置网络连接，在服务器端除了启动数据库外，还要执行下面命令启动监听器：

```
C:\>lsnrctl start
```

然后使用下面的简易方式连接至数据库，简易方式不需要在客户端配置网络连接：

```
C:\>sqlplus scott/tiger@law_x240:1521/law
```

law_x240 是要连接的服务器名称，1521 为端口号，law 为数据库名称。

若端口号为默认的 1521，则可以将其省略，只需指定服务器和数据库名称：

```
C:\>sqlplus scott/tiger@law_x240/law
```

也可以使用本地网络服务名连接远程数据库，如：

```
C:\> sqlplus scott/tiger@test
```

test 是在客户端机器上配置的本地网络服务名。

SQL*Plus 和 sqlcmd 都是使用 exit 或 quit 命令退出连接。

2.3.2 查看软件版本

Oracle 使用 v\$version 数据字典视图查看软件版本：

```
SQL> select banner from v$version;
```

BANNER

```
Oracle Database 12c Enterprise Edition Release 12.1.0.1.0 - 64bit Production
PL/SQL Release 12.1.0.1.0 - Production
CORE 12.1.0.1.0 Production
TNS for 64-bit Windows: Version 12.1.0.1.0 - Production
NLSRTL Version 12.1.0.1.0 - Production
```

也可以使用 SQL*Plus 的 show release 命令以简洁方式列出版本号：

```
SQL> show release
release 1201000100
```

SQL Server 使用 @@version 全局变量查看软件版本：

```
1> select @@version
2> go

-----
Microsoft SQL Server 2016 (RTM) - 13.0.1601.5 (X64)
Apr 29 2016 23:23:58
Copyright (c) Microsoft Corporation
Developer Edition (64-bit) on Windows 8.1 Enterprise 6.3 <X64> (Build 9600: )
```

2.3.3 切换用户

SQL*Plus 使用 connect 命令(可简写为 conn)切换用户。切换用户后，之前的用户自动退出连接。查看当前用户名称，可执行 show user 命令。

```
SQL> conn scott/tiger
已连接。
SQL> show user
USER 为 "SCOTT"
```

与 SQL*Plus 类似，在 sqlcmd 中也可以使用 connect 令切换用户，但要加上冒号。另外要使用 -U 及 -P 参数分别指定登录名及口令。

```
1> :connect law_x240 -U user1 -P testpassword
Sqlcmd: 已成功连接到服务器 "law_x240"。
```

若使用操作系统验证，则不必附加登录名及口令。

```
1> :connect law_x240
Sqlcmd: 已成功连接到服务器 "law_x240"。
```

切换完成后，SQL Server 自动关闭之前的用户连接。

其中的 law_x240 是连接的数据库服务器的机器名称，若为默认实例，则可以如同示例只输入机器名称。若是非默认实例，可以在机器名称后附加实例名称，如：law_x240\instance1。

要查询当前登录账号及数据库用户名称，则可以分别使用下面命令：

```
1> print system_user
2> go
law_x240\Administrator
1> print user
2> go
```

其中第一个命令输出当前服务器登录名，第二个命令输出当前数据库用户。

2.3.4 切换数据库

Oracle 的一个数据库对应一个服务，切换数据库，也就是切换数据库服务。

本地服务器如果创建了多个数据库，可以通过设置 ORACLE_SID 环境变量切换 SQL*Plus 默认连接的数据库，如下所示：

```
C:\>set oracle_sid=law
C:\>sqlplus scott/tiger
SQL*Plus: Release 12.1.0.1.0 Production on 星期三 12月 23 21:32:50 2015
Copyright (c) 1982, 2013, Oracle. All rights reserved.

上次成功登录时间: 星期三 12月 23 2015 20:56:49 +08:00

连接到:
Oracle Database 12c Enterprise Edition Release 12.1.0.1.0 - 64bit Production
With the Partitioning, OLAP, Advanced Analytics and Real Application Testing options
```

SQL Server 服务器实例上可以创建多个数据库，连接到服务器后，可以随时使用 use 命令切换数据库，前提是当前服务器登录帐号已经映射为这多个数据库中的用户，如下面命令所示：

```
C:\>sqlcmd
1> use law
2> go
已将数据库上下文更改为 'law'。
```

2.3.5 查看当前连接的数据库名称

SQL*Plus 中查看当前连接的数据库名称，需要以 system 或 sys 用户连接数据库，执行下面命令：

```
SQL> conn system/oracle
已连接。
SQL> show parameter db_name
```

NAME	TYPE	VALUE
db_name	string	law

SQL Server 中的 db_name()函数可返回当前数据库名称：

```
1> print db_name()
2> go
law
```

2.3.6 修改密码

SQL*Plus 使用 password 命令修改密码，任何用户可以修改自己的密码，管理员可以修改其他用户的密码。

下面是 scott 用户修改自己的密码，新旧密码显示时，不会显示在屏幕上：

```
SQL> conn scott/tiger
已连接。
SQL> password
更改 SCOTT 的口令
旧口令:
新口令:
重新键入新口令:
```

口令已更改

下面命令是 system 用户修改 scott 用户的密码:

```
SQL> conn system/oracle
已连接。
SQL> password scott
更改 scott 的口令
新口令:
重新键入新口令:
口令已更改
```

sqlcmd 在启动时使用 -z 参数指定当前帐号的新密码, 下面命令把 sa 的口令改为 sasa:

```
C:\>sqlcmd -U sa -P sa -z sasa
```

2.3.7 执行 SQL 命令

Oracle 的 SQL*Plus 中, 使用分号或斜杠来执行 SQL 语句, 使用斜杠时, 要另起新行。

```
SQL> conn scott/tiger
已连接。
SQL> select * from dept;
```

DEPTNO	DNAME	LOC
10	ACCOUNTING	NEW YORK
20	RESEARCH	DALLAS
30	SALES	CHICAGO
40	OPERATIONS	BOSTON

```
SQL> select * from dept
2 /
```

DEPTNO	DNAME	LOC
10	ACCOUNTING	NEW YORK
20	RESEARCH	DALLAS
30	SALES	CHICAGO
40	OPERATIONS	BOSTON

SQL Server 的 sqlcmd 中, 输入 “go” 来执行 SQL 命令。

下面我们执行几个简单的 SQL 命令。

首先执行 create database 命令在服务器上创建一个新的数据库 testDB:

```
C:\>sqlcmd -E
1> create database testDB
2> go
```

使用 use 命令连接到新建的 testDB 数据库:

```
1> use testDB
2> go
```

在 testDB 数据库上新建表 t:

```
1> create table t(a int, b int)
2> go
```

添加记录:

```
1> insert into t values(1, 10), (2, 20), (3, 30)
2> go
```

然后执行查询：

```
1> select * from t
2> go
a          b
-----
          1          10
          2          20
          3          30
```

2.3.8 设置客户端界面每行容纳的字符数

设置 SQL*Plus 每行容纳的字符数量，可以在启动 SQL*Plus 后，使用 `set linesize` 命令，查询其值，则可以使用 `show` 命令：

```
SQL> show linesize
linesize 80
SQL> set linesize 300
SQL> show linesize
linesize 300
```

除 `linesize` 外，SQL*Plus 还有很多其他环境变量，用 `show all` 命令可以显示这些变量的当前值，下面其显示结果的前几项：

```
SQL> show all
appinfo 为 OFF 并且已设置为 "SQL*Plus"
arraysize 15
autocommit OFF
autoprint OFF
autorecovery OFF
autotrace OFF
blockterminator "." (hex 2e)
btitle OFF 为下一条 SELECT 语句的前几个字符
cmdsep OFF
.....
```

若要以上变量值永久生效，可以将其存入 `%ORACLE_HOME%\sqlplus\admin\glogin.sql` 文件，SQL*Plus 启动时，会先读取这个文件，设置其环境变量。

设置 `sqlcmd` 每行容纳的字符数量，只能在启动 `sqlcmd` 时以 `-w` 参数指定。

```
C:\>sqlcmd -w 300
```

2.3.9 修改执行过的 SQL 命令以重新执行

使用上、下方向键可以把执行过的命令回显，然后使用左、右方向键移动光标至合适位置修改特定内容。除此方法外，SQL*Plus 及 `sqlcmd` 中都可以使用 `ed` 命令在编辑器中修改执行过的上一条 SQL 命令。

SQL*Plus 不需任何配置即可使用 `ed` 命令在记事本中修改之前执行的 SQL 语句，缓冲区的 SQL 命令会存入当前目录下的 `afiedt.buf` 文件，SQL*Plus 之前的客户端工具为 UFI 的升级版 AFI，当时使用 VAX/VMS 系统，编辑器名称为 EDT，`afiedt` 的叫法即来源于此。

`sqlcmd` 也可以把执行过的命令显示在记事本中修改，但要设置 `sqlcmdeditor` 环境变量。

右键单击“我的电脑”图标，弹出菜单中选择“属性”，在属性对话框中选择“高级”，然后点击“环境变量”按钮，在“系统变量”部分点击“新建”按钮，“变量名”中填入

“sqlcmdeditor”，“变量值”填入“notepad”。完成以上设置后，重启 cmd 工具使其读取新的环境变量值即可生效。

2.3.10 执行 SQL 脚本文件

在记事本中编辑以下内容，保存为 sql 脚本文件，文件名称为 e:\test.sql。

```
select * from dept;
```

然后可以在 SQL*Plus 中使用 start 执行以上脚本：

```
SQL> conn scott/tiger
已连接。
SQL> start e:\test.sql
```

DEPTNO	DNAME	LOC
10	ACCOUNTING	NEW YORK
20	RESEARCH	DALLAS
30	SALES	CHICAGO
40	OPERATIONS	BOSTON

以上的 start 关键字也可以替换为 “@”。

SQL Server 的 sqlcmd 使用:r 执行 SQL 脚本，如果脚本文件最后没有 go 命令，则要单独输入 go，如下面命令所示：

```
C:\>sqlcmd -d law
1> :r e:\test.sql
2> go
```

DEPTNO	DNAME	LOC
10	ACCOUNTING	NEW YORK
20	RESEARCH	DALLAS
30	SALES	CHICAGO
40	OPERATIONS	BOSTON

启动 sqlcmd 时也可以使用-i 参数执行 sql 脚本文件，-d 参数用于指定数据库名称：

```
C:\Windows\system32>sqlcmd -d law -i e:\test.sql
```

DEPTNO	DNAME	LOC
10	ACCOUNTING	NEW YORK
20	RESEARCH	DALLAS
30	SALES	CHICAGO
40	OPERATIONS	BOSTON

2.3.11 查询结果保存为文件

SQL*Plus 使用 spool 命令实现，下面命令把 spool 开始直到 spool off 之间的内容都导出到 c:\result.txt 文件，在当前客户端也会显示查询结果：

```
SQL> spool c:\result.txt
SQL> select * from dept;
```

DEPTNO	DNAME	LOC
10	ACCOUNTING	NEW YORK
20	RESEARCH	DALLAS
30	SALES	CHICAGO
40	OPERATIONS	BOSTON

```
SQL> spool off
```

sqlcmd 使用:out 命令指定输出结果保存的文件名称, 客户端不会显示查询结果:

```
C:\>sqlcmd -d law
1> :out e:\out.txt
1> select * from dept
2> go
1> exit
```

执行 type 命令查看输出文件内容。

```
C:\>type e:\out.txt
DEPTNO DNAME          LOC
-----
10 ACCOUNTING        NEW YORK
20 RESEARCH           DALLAS
30 SALES              CHICAGO
40 OPERATIONS         BOSTON
```

也可以在启动 sqlcmd 时, 以-i 参数指定 SQL 脚本文件, 以-o 参数指定输出文件:

```
C:\>sqlcmd -d law -i e:\test.sql -o e:\out.txt
C:\>type e:\out.txt
DEPTNO DNAME          LOC
-----
10 ACCOUNTING        NEW YORK
20 RESEARCH           DALLAS
30 SALES              CHICAGO
40 OPERATIONS         BOSTON
```

2.3.12 设置查询结果的列宽

在 SQL*Plus 中, 可以使用 `column column_name format an` 命令设置字符串列的显示宽度, 以避免显示结果因为换行而显得凌乱, 其中的 `column` 及 `format` 关键字可分别简写为 `col` 及 `for`, `a` 后面的数字 `n` 用于指定字符数量。

如查询 `file_id` 为 1 的数据文件名称, 在未设置 `file_name` 的列宽时, 显示结果如下:

```
SQL> conn system/oracle
已连接。
SQL> select file_name from dba_data_files where file_id=1;

FILE_NAME
-----
E:\ORACLE\ORADATA\LAW11G\SYSTEM01.DBF
```

如果设置其列宽为 40 个字符, 则结果如下:

```
SQL> col file_name for a40
SQL> select file_name from dba_data_files where file_id=1;

FILE_NAME
-----
E:\ORACLE\ORADATA\LAW11G\SYSTEM01.DBF
```

sqlcmd 不能设置指定列的宽度, 只可以在启动时通过 `-Y` 及 `-y` 参数设置字符串列的宽度。
`-y` 参数设置下列类型列的宽度:

- `varchar(max)`

- nvarchar(max)
- varbinary(max)

varchar(max), nvarchar(max), varbinary(max)是 SQL Server 2005 的新数据类型, 用于替换 text, ntext, image 这三种数据类型, 其范围都是最大 $2^{31}-1$ 个字节。

-Y 用于设置下列类型列的宽度:

- char(n)
- nchar(n)
- varchar(n)
- nvarchar(n)
- sql_variant

设置这两个参数后的实际效果, 我们可以做以下测试:

```
1> create table t(a varchar(max), b varchar(30))
2> go
```

设置-Y 参数为 10, 影响 b 列的显示宽度:

```
C:\>sqlcmd -Y 10
1> select * from t
2> go
a                                     b
-----
(0 行受影响)
```

设置-y 参数为 10, 则可以影响 a 列的显示宽度:

```
C:\>sqlcmd -y 10
1> select * from t
2> go
a      b
-----
(0 行受影响)
```

2.3.13 查询表的结构, describe 命令

SQL*Plus 可以执行 describe 命令查询表的结构, 非常方便:

```
SQL> conn scott/tiger
已连接。
SQL> describe dept
 名称                                     是否为空? 类型
-----
DEPTNO                                NOT NULL   NUMBER(2)
DNAME                                  VCHAR2(14)
LOC                                    VCHAR2(13)
```

sqlcmd 不支持这样的用法, 我们编写一个 describe 存储过程, 完成同样的功能。

```
create proc dbo.describe
@table_name Nvarchar(100)
as
select c.name as column_name,
case t.name
when 'numeric' then
t.name+'('+cast(c.precision as varchar(5))+','+cast(c.scale as varchar(5))+')'
```

```

when 'char' then
    t.name+'('+cast(c.max_length as varchar(5))+')'
when 'varchar' then
    t.name+'('+cast(c.max_length as varchar(5))+')'
when 'nvarchar' then
    t.name+'('+cast(c.max_length as varchar(5))+')'
else t.name
end as type,
case c.is_nullable
when 1 then ' '
    else 'NOT NULL'
end as nullable
from sys.all_columns as c, sys.types as t
where c.system_type_id=t.system_type_id
and t.name<>'sysname' and c.object_id=object_id(@table_name)

```

其用法如下：describe 'schema_name.table_name'

若表属于用户的默认架构，可省略表名之前的架构名称，下面是执行这个存储过程的示例。查看 dept 表的结构：

```

1> describe dept
2> go
column_name          type                nullable
-----
DEPTNO              numeric(2,0)        NOT NULL
DNAME               varchar(14)
LOC                varchar(13)

```

查看目录视图 sys.schemas 的结构：

```

1> describe 'sys.schemas'
2> go
column_name          type                nullable
-----
schema_id            int                 NOT NULL
principal_id         int
name                 nvarchar(256)      NOT NULL

```

2.4 图形工具使用方法

相比命令行工具，图形界面工具对初学者来说更容易入门。微软一直重视图形界面工具的开发。为了提高市场占有率，Oracle 也在图形工具方面投入了越来越多的资源。本节介绍两种产品的图形工具和使用方法。

2.4.1 Oracle 的图形工具

Oracle database 12c 的图形界面管理工具有面向开发者的 SQL Developer 和面向数据库管理者的 Enterprise Manager。

在 Oracle 程序组，点击“应用程序开发”中的 SQL Developer 即可启动 SQL Developer。

首次启动，需设置 java.exe 所在路径，一般为：%ORACLE_HOME%\jdk\bin。这里的 ORACLE_HOME 是安装 Oracle 数据库软件时，在操作系统设置的环境变量，一般为：x:\app\user_name\product\12.1.0\dbhome_1\jdk\bin，x 是分区符号，user_name 是执行软件安装任务的操作系统用户名称。

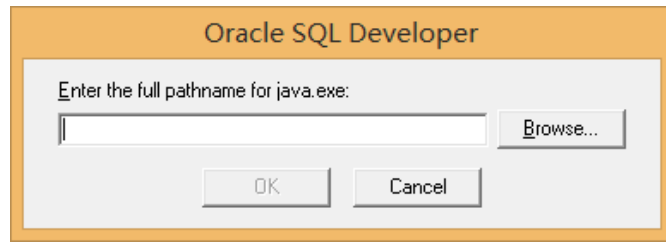


图2-3 启动SQL Developer

启动后，右键点击“连接”，然后选择“创建本地连接”，如下图所示：



图2-4 创建本地连接

数据库中未被锁住的用户都会建立一个独立的连接，如下图所示：

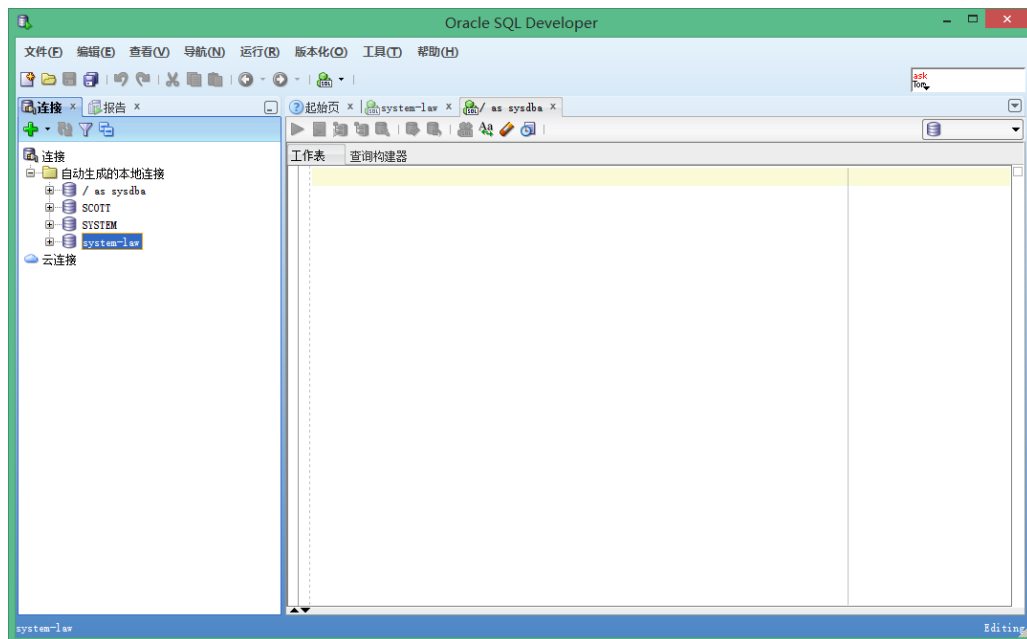


图2-5 建立数据库连接后的SQL Developer主界面

点击连接名称之前的“+”号，会弹出以下对话框要求输入对应用户的口令：



图2-6 输入用户口令

输入口令后，点击“确定”按钮，左侧用户名之前的“+”会展开，列出当前连接用户下的所有数据库对象。在右侧输入 SQL 命令，点击编辑窗口上方的第一个三角形按钮，即可执行。如果编辑窗口中有多条 SQL 命令，则可以先选中要执行的命令，然后点击执行按钮，如下图所示：

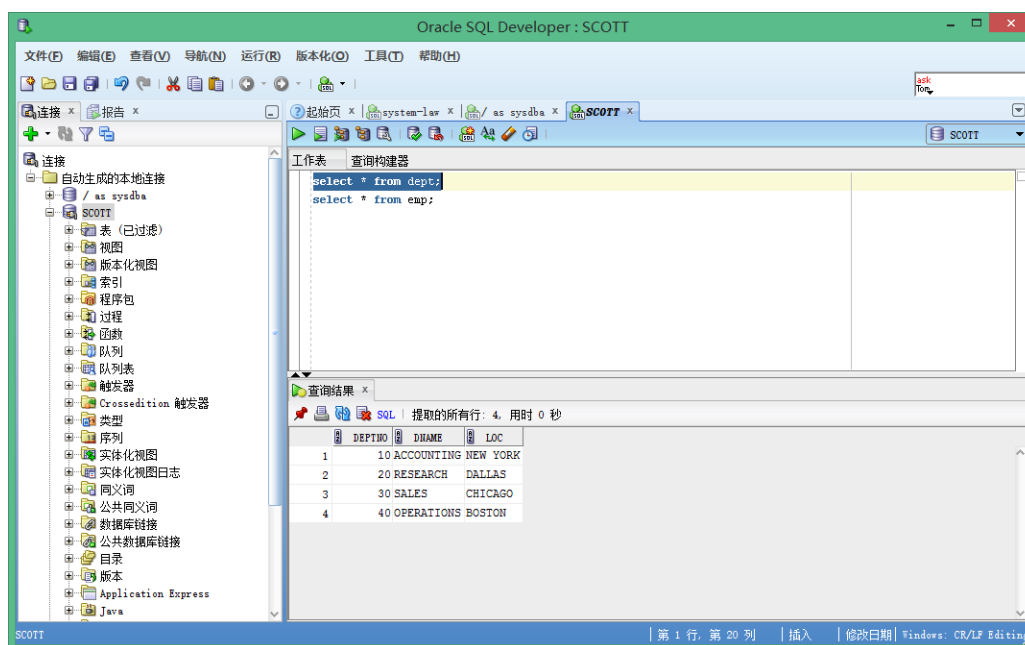


图2-7 在SQL Developer中执行SQL语句

在左侧树形结构中，单击某个数据库对象，如 dept 表，则会在右侧主窗口中显示其各种属性信息，如下图所示：

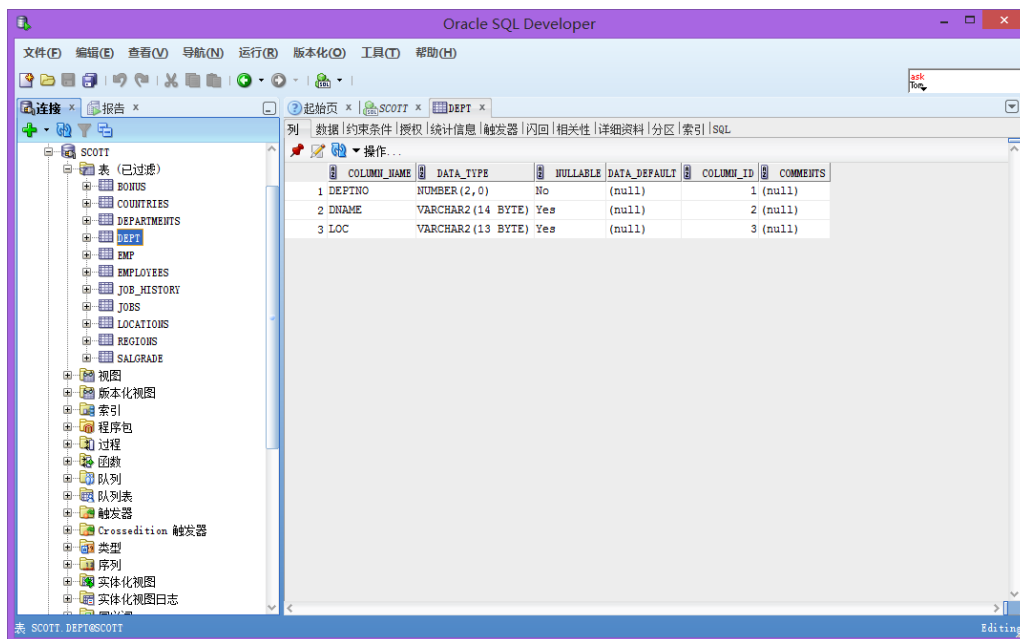


图2-8 查看表的结构

在左侧窗口中，可以右键点击相应对象，在弹出菜单中选择合适的内容执行对相关数据库对象的操作。如对表来说，如执行“编辑”命令，可以在新的对话框中修改表的结构，如下图所示：

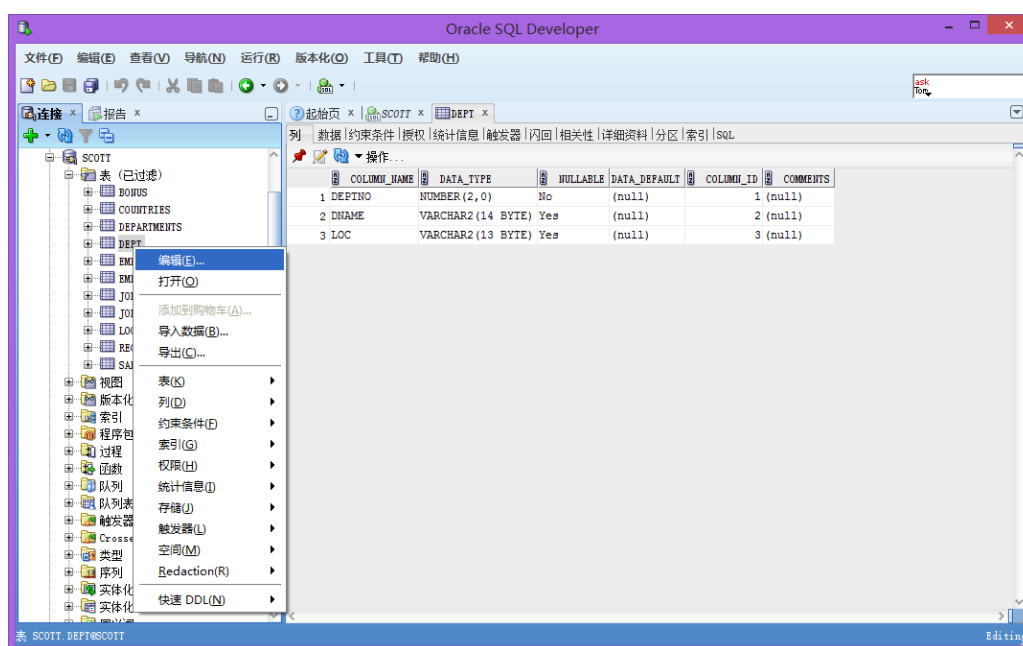


图2-9 表的操作菜单

下图是编辑表的对话框：

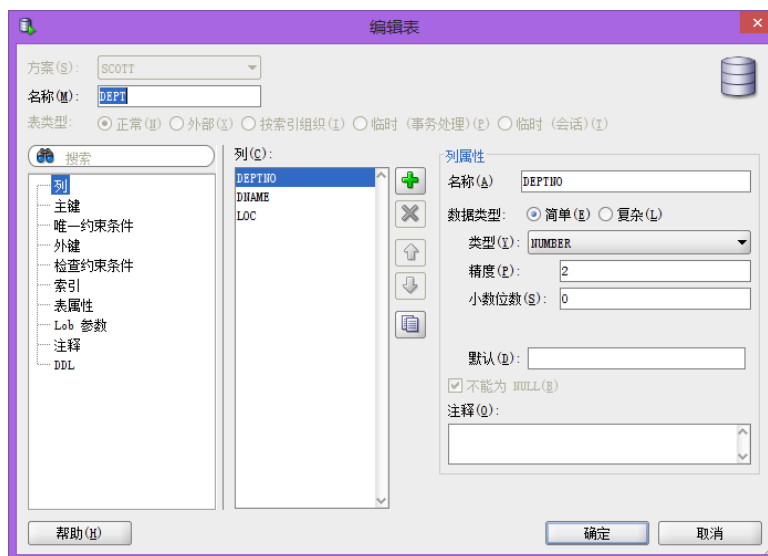


图2-10 修改表的结构

如果点击左侧窗口的 Reports 部分，则可以浏览选定用户有权限查看的关于数据库的更多信息，如下图显示的是 scott 用户可以查看的字典视图：

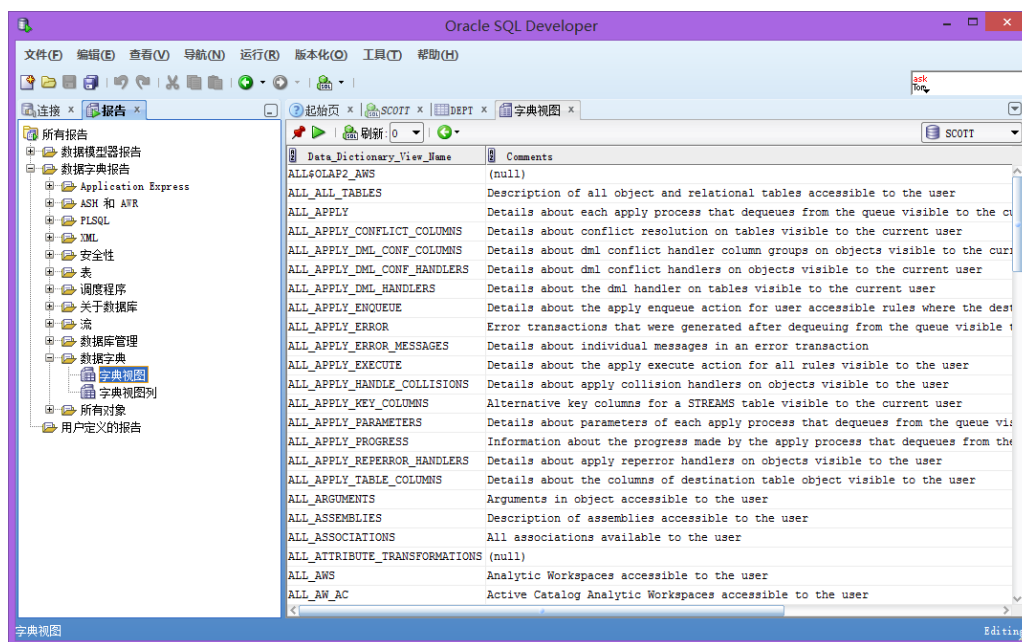


图2-11 浏览更多信息

下面使用 SQL Developer 建立到 SQL Server 数据库的连接。

用 SQL Developer 连接 SQL Server，需要安装 SQL Server JDBC 驱动程序 jtds，其下载网址为：<http://sourceforge.net/projects/jtds/files/jtds/>，选择下载页面中的 1.2 版本，得到 jtds-1.2-dist.zip 压缩包，解压到任意一个目录即可。

点击在 SQL Developer 的“工具”菜单，选择“首选项..”命令，如下图所示：

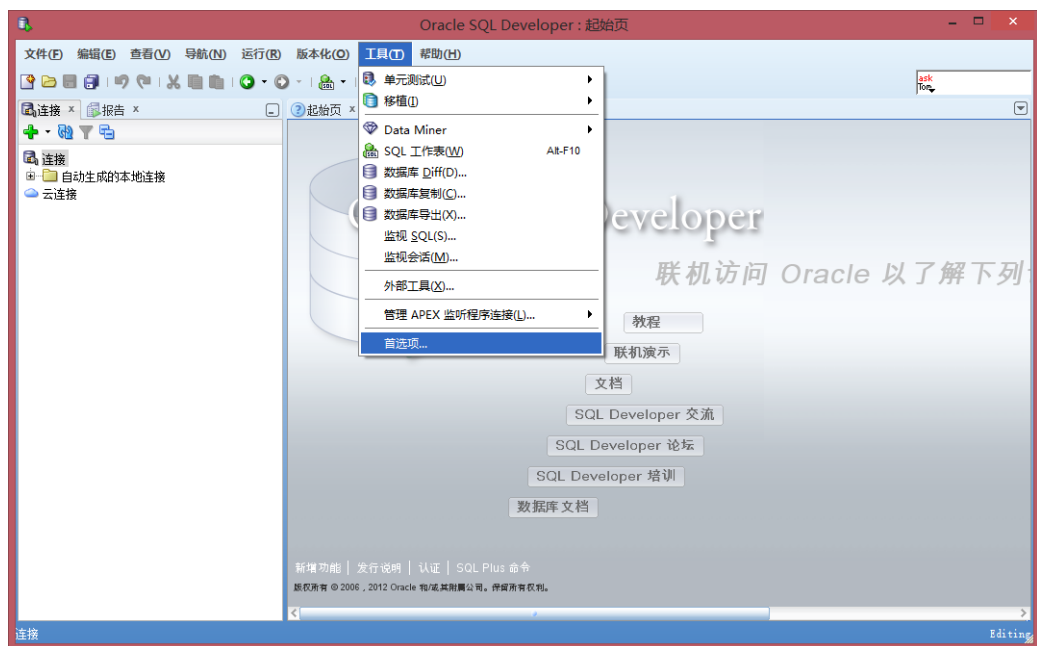


图2-12 添加第三方JDBC驱动程序

在“数据库”部分，选择“第三方 JDBC 驱动程序”，如下图所示：

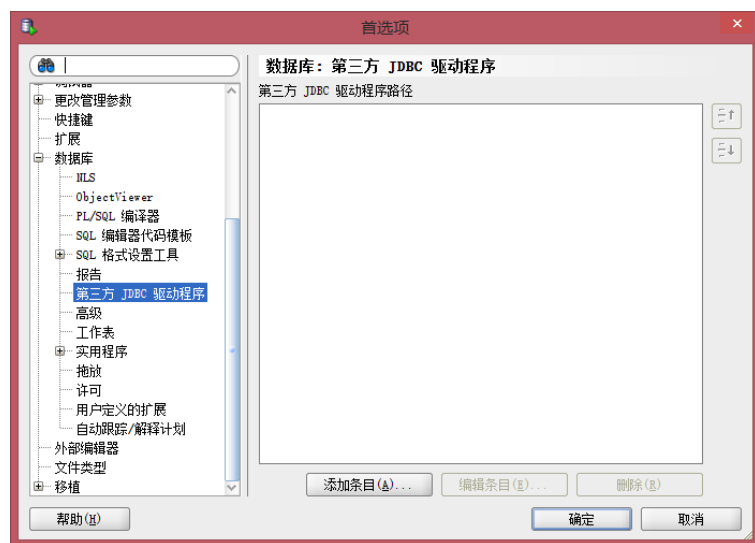


图2-13 添加第三方JDBC驱动程序

点击右侧的“添加条目(A)...”按钮，找到解压 jtds-1.2-dist.zip 后的目录，选择其中的 jtds-1.2-dist.jar，点击“选择”按钮，如下图所示：

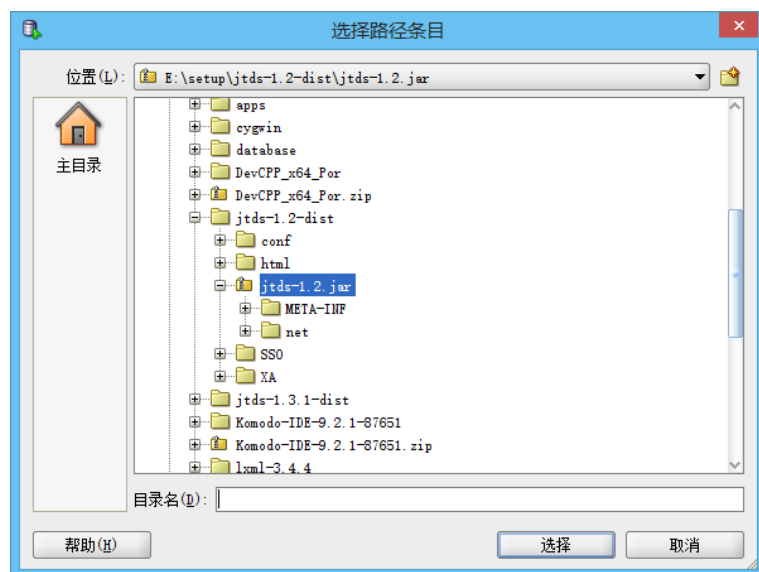


图2-14 选择JDBC驱动程序

第三方 JDBC 驱动程序路径窗口会加入所选目录，点击“确定”按钮，如下图所示：

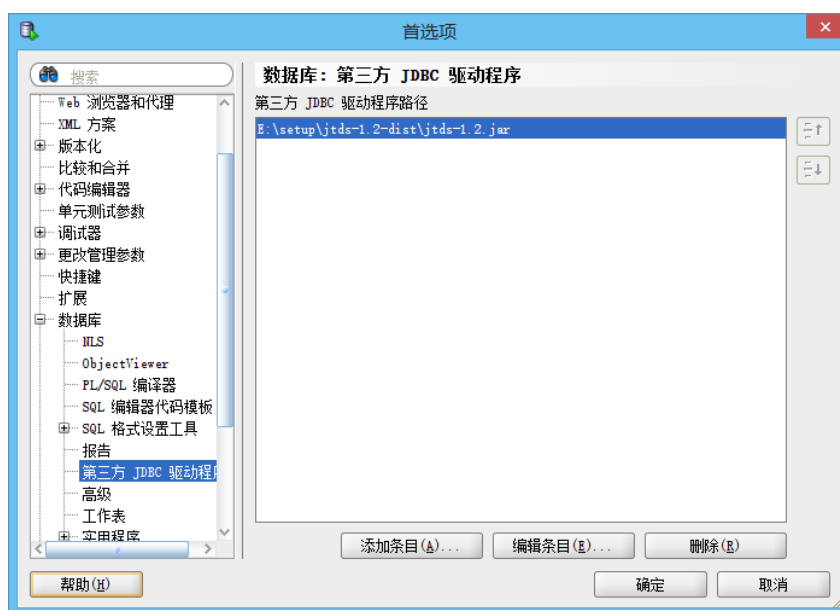


图2-15 添加第三方JDBC驱动程序

新建连接,在连接设置对话框中会显示 SQL Server 和 Sybase 选项卡,选择“SQL Server”,如下图所示,以 sa 用户连接服务器(需先设置 sa 的口令,并开启服务器端的 SQL Server 验证方式):



图2-16 配置SQL Server连接属性

在 SQL Developer 的左侧树形结构中会显示 SQL Server 服务器中的各个数据库, 在右侧 SQL 编辑器窗口可以执行 SQL 命令。执行 SQL 时, 要指定表所属的数据库及架构名称。

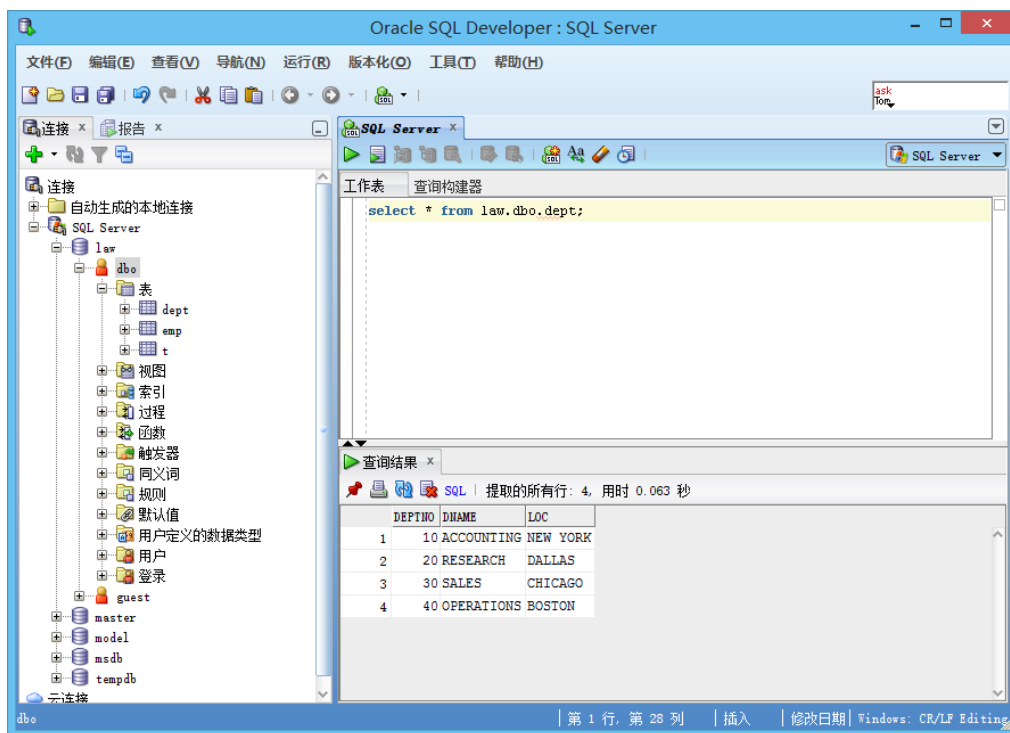


图2-17 在SQL Developer中操作SQL Server

Oracle 12c 的 Enterprise Manager Database Express 也是通过 web 页面对数据库进行管理的客户端工具。要使用此工具, 需在 dbca 建库时, 勾选“Enterprise Manager Database Express”。

服务器端启动监听器和数据库后, 在浏览器的地址栏输入: https://server_name:5500/em, 即可访问服务器, server_name 替换为服务器所在的机器名称, 如果连接本地服务器, 则可以使用 localhost。

第一次访问时, 浏览器认为此连接为非私密连接而报错, 点击“依然前往”即可。

下面是连接服务器后的登录界面, 用户名输入 sys 或 system, 并输入相应口令。

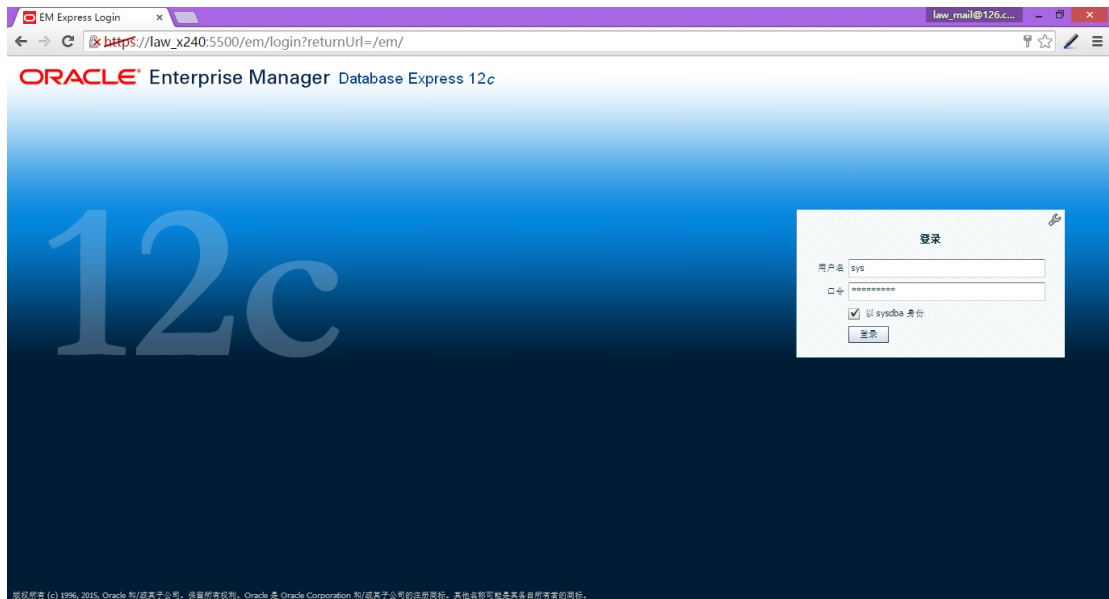


图2-18 Enterprise Manager Database Express登录界面

下面是登录后的主操作界面：

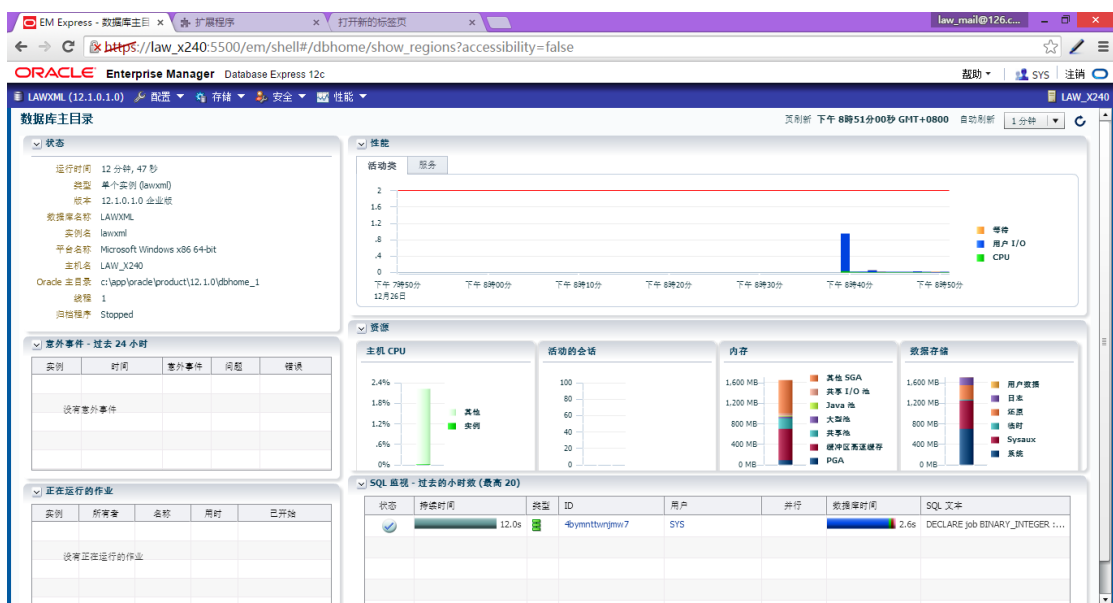


图2-19 Enterprise Manager Database Express主界面

2.4.2 SQL Server 的 Management Studio

在 SQL Server 程序组中选择 SQL Server Management Studio，启动 SSMS。
若连接本地服务器，Server Name 输入句点 “.” 即可，其连接设置窗口如下：



图2-20 填写Management Studio登录信息

在身份验证部分选择 Windows 身份验证，或选择 SQL Server 身份验证，输入相应服务器登录帐号及口令，连接服务器后，打开如下主界面：

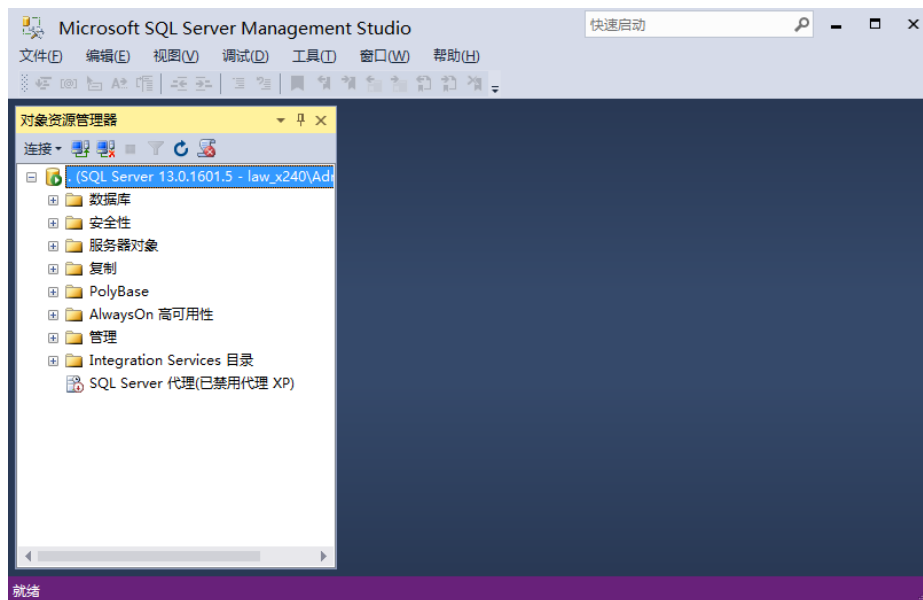


图2-21 Management Studio主界面

可以在左侧树形结构中浏览各种服务器和数据库对象信息，右键点击某个对象，弹出菜单中看到可以执行的更多操作。

点击“文件”菜单，选择“新建”按钮，再选择“数据库引擎查询”，可以在右侧打开 SQL 编辑窗口。输入 SQL 命令后，点击工具栏的“执行”按钮，可以执行编辑窗口中选中的 SQL 命令，在窗口下面除了显示执行结果外，还显示了命令执行的时间，在顶部的工具栏还可以选择“显示估计的执行计划”或“包括实际的执行计划”。可以把鼠标移动至相应按钮，通过提示查看其功能。

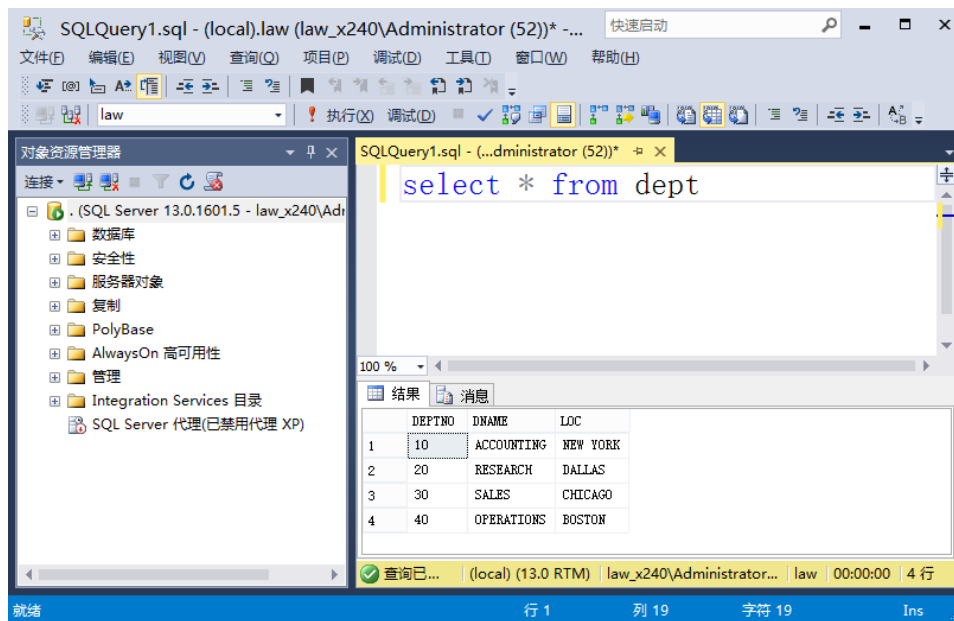


图2-22 Management Studio中执行SQL命令