

Lecture 14 数据库系统实验

Yubao Liu (刘玉葆)

School of Data and Computer Science

Sun Yat-sen University

- 本节课提纲

- 实验目的
- 实验内容
- 实验示例
- 练习

- 实验目的

- 1、学会识别锁冲突，学会检查和处理死锁。
- 2、通过实验加深对数据安全性的理解，熟悉视图机制在自主存取控制上的应用。
- 3、利用DBCC命令读取日志。

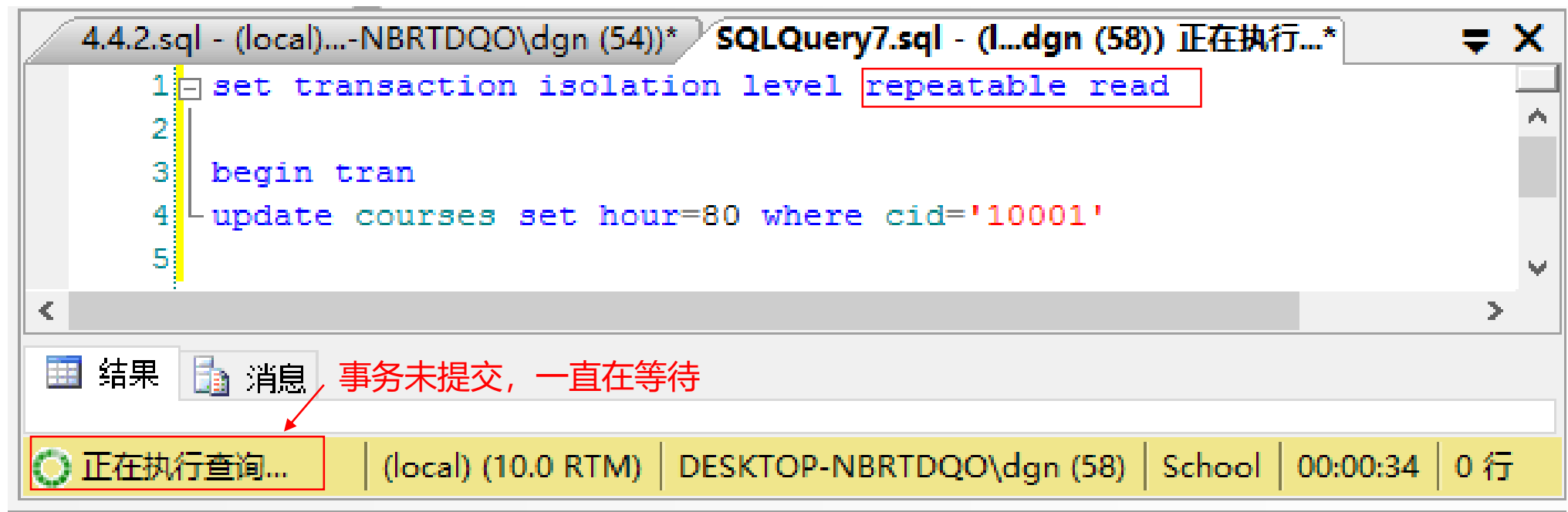
- 实验内容

1. 设计实验造成事务对资源的争夺，分析原因，讨论解决锁争夺的办法。
2. 设计实验制造事务之间的死锁，分析造成死锁的原因。
3. 关系数据库中授权的数据对象粒度从大到小为数据库、表、列、元组。直接使用授权机制所能达到的数据对象的最小粒度是列，为了使数据粒度达到元组这一级，必须利用视图机制和授权机制配合使用。
4. 利用DBCC命令读取日志。

实验示例

1.演示事务对资源的争夺。

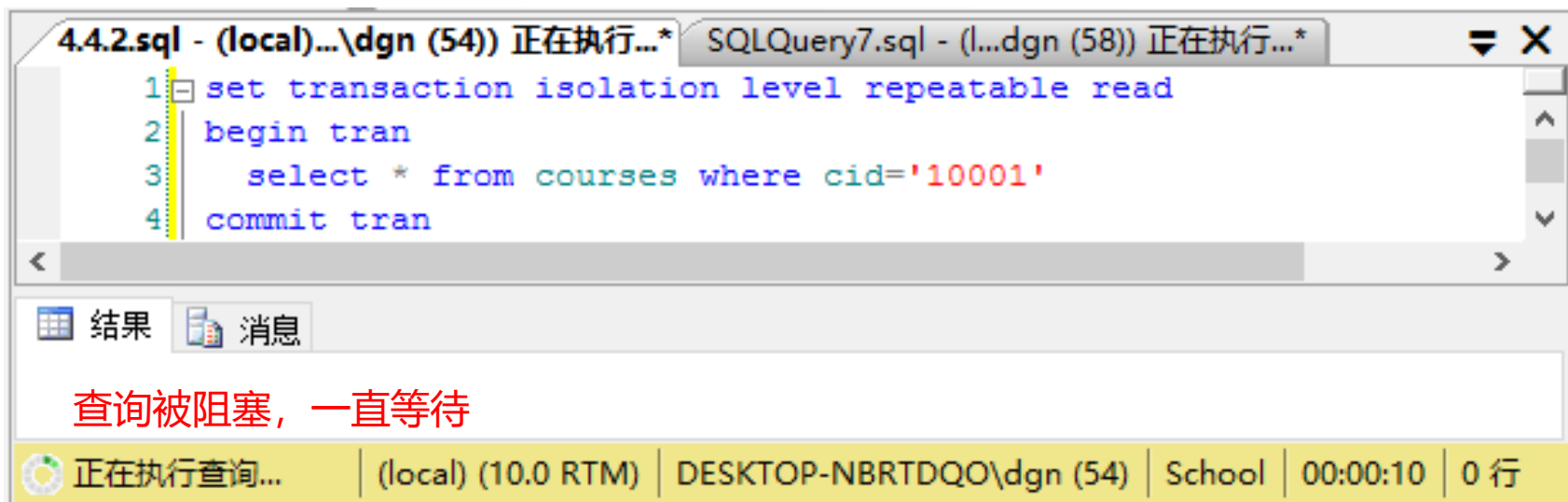
Step1: 建立一个连接, 更新courses表中的database的课时 (为了制造锁争夺, 更新事务没有提交) 。



注: 设置repeatable read隔离级别, 如果某行在被读取之后由事务进行了修改, 则该事务会获取一个用于保护该行的排他锁, 并且该排他锁在事务完成之前将一直保持 (即不允许其他事务对该行进行读取或者修改) 。

实验示例

Step2: 建立第二个连接, 执行查询事务, 发现被连接1阻塞。

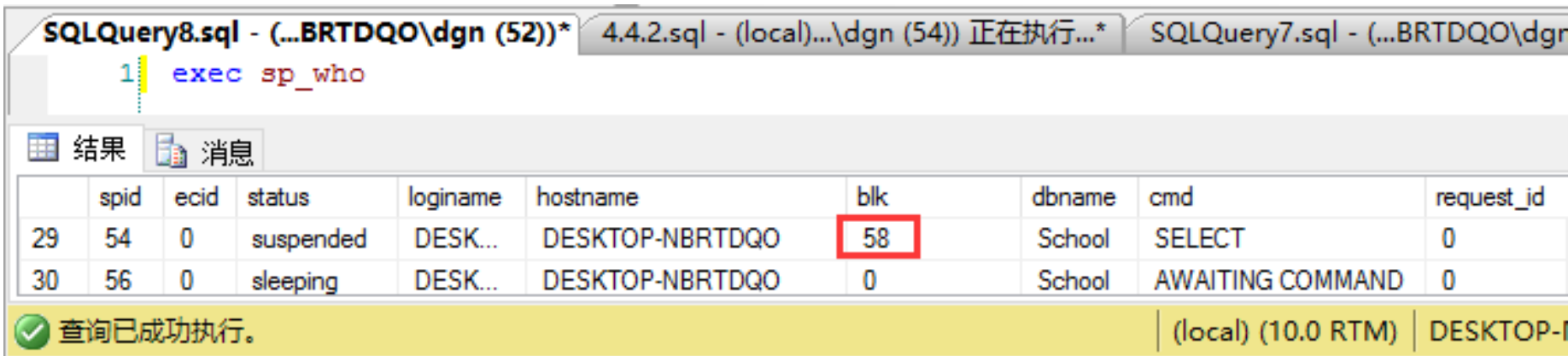


The screenshot shows a SQL Server Enterprise Manager interface. The top pane displays a query window titled "4.4.2.sql - (local)...\dgn (54)) 正在执行...". The query text is:

```
1 set transaction isolation level repeatable read
2 begin tran
3     select * from courses where cid='10001'
4 commit tran
```

The bottom pane shows a message icon and the text "查询被阻塞, 一直等待" (Query is blocked, waiting indefinitely). The status bar at the bottom indicates "正在执行查询..." (Executing query...), "(local) (10.0 RTM)", "DESKTOP-NBRTDQO\dgn (54)", "School", "00:00:10", and "0 行" (0 rows).

注: 验证进程是否受阻, 可通过" exec sp_who", 检查blk列非0值 (即阻塞), 如下图说明进程54被进程58阻塞。



The screenshot shows a SQL Server Enterprise Manager interface. The top pane displays a query window titled "SQLQuery8.sql - (...BRTDQO\dgn (52))*". The query text is:

```
1 exec sp_who
```

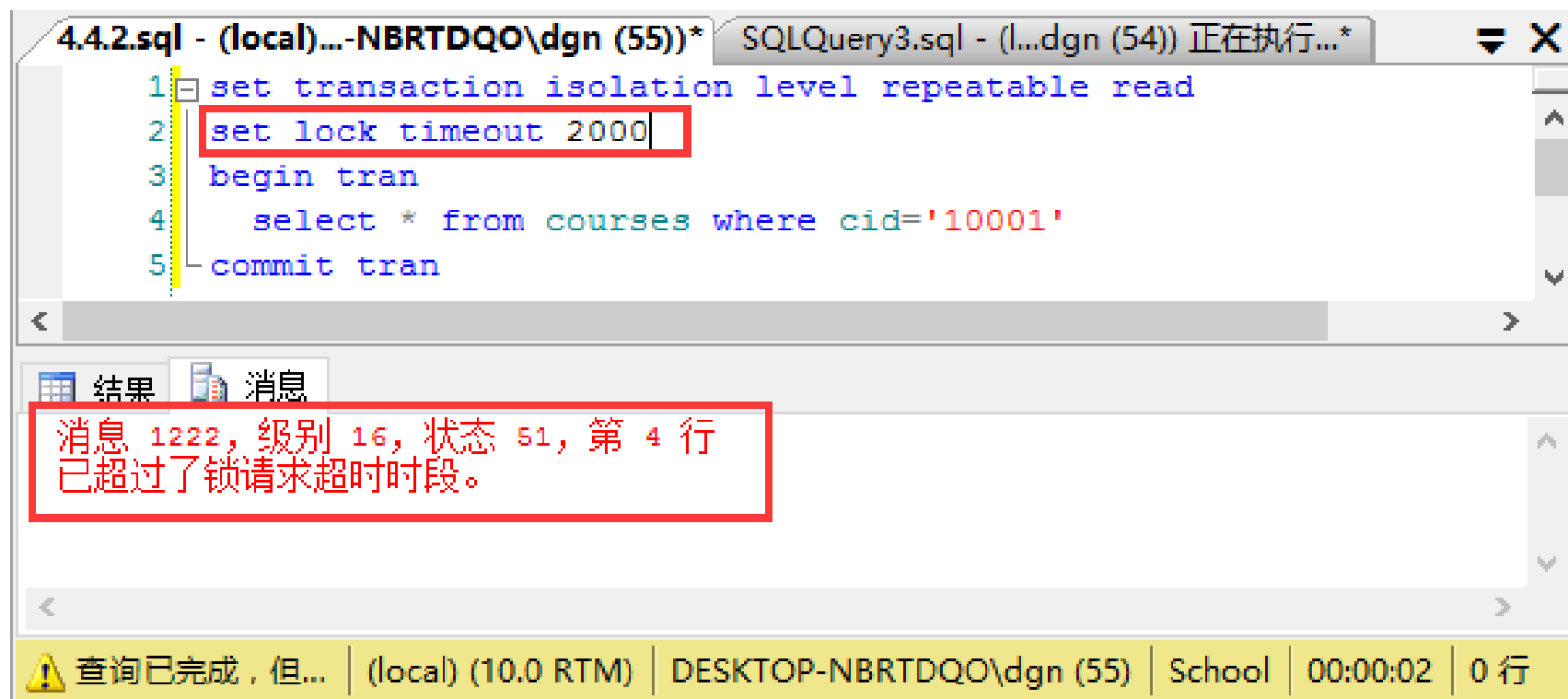
The bottom pane shows the results of the query in a table format. The table has the following columns: spid, ecid, status, loginame, hostname, blk, dbname, cmd, and request_id. The data is as follows:

spid	ecid	status	loginame	hostname	blk	dbname	cmd	request_id
29	54	suspended	DESK...	DESKTOP-NBRTDQO	58	School	SELECT	0
30	56	sleeping	DESK...	DESKTOP-NBRTDQO	0	School	AWAITING COMMAND	0

The status bar at the bottom indicates "查询已成功执行。" (Query executed successfully), "(local) (10.0 RTM)", and "DESKTOP-I".

• 实验示例

为了解决“永久等待”，可通过lock_timeout设置锁定超时时间间隔。超时后，锁定管理器将自动解除锁的争夺。



The screenshot displays the SQL Server Enterprise Manager interface. The top pane shows a query window titled "4.4.2.sql - (local)...\NBRTDQO\dgn (55))*" and "SQLQuery3.sql - (l...dgn (54)) 正在执行...". The query text is as follows:

```
1 set transaction isolation level repeatable read
2 set lock timeout 2000
3 begin tran
4     select * from courses where cid='10001'
5 commit tran
```

The second pane shows the "消息" (Messages) window, which contains the following error message:

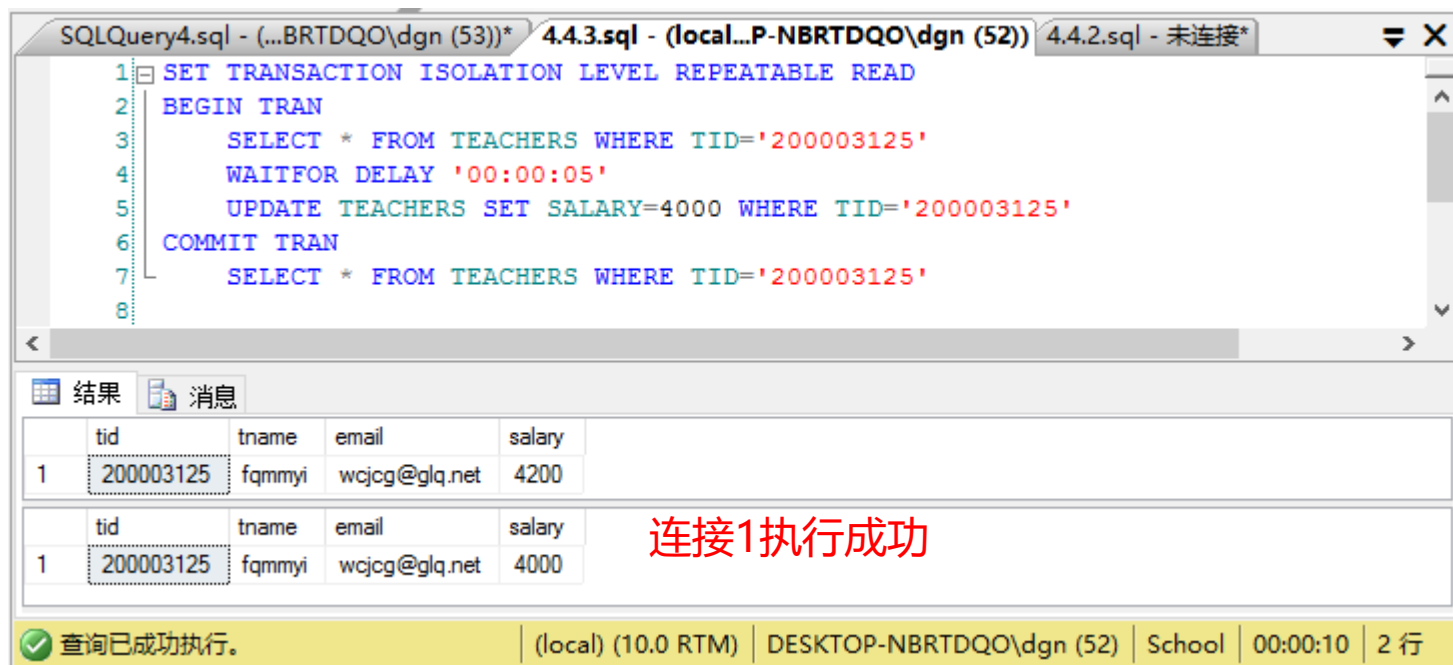
消息 1222, 级别 16, 状态 51, 第 4 行
已超过了锁请求超时时段。

The bottom status bar indicates: "查询已完成, 但..." | (local) (10.0 RTM) | DESKTOP-NBRTDQO\dgn (55) | School | 00:00:02 | 0 行

实验示例

2.演示事务间的死锁。

打开两个连接，同时执行下面的代码。可以发现有一个连接可以查询，另一个连接由于死锁，直接停掉了当前程序工作，并回滚之前的事务。



The screenshot shows a SQL query window with the following content:

```
1 SET TRANSACTION ISOLATION LEVEL REPEATABLE READ
2 BEGIN TRAN
3     SELECT * FROM TEACHERS WHERE TID='200003125'
4     WAITFOR DELAY '00:00:05'
5     UPDATE TEACHERS SET SALARY=4000 WHERE TID='200003125'
6 COMMIT TRAN
7     SELECT * FROM TEACHERS WHERE TID='200003125'
8
```

Below the query, the results are displayed in a table:

	tid	tname	email	salary
1	200003125	fqmyi	wcjcg@glq.net	4200

Below the first table, the results are displayed in a second table:

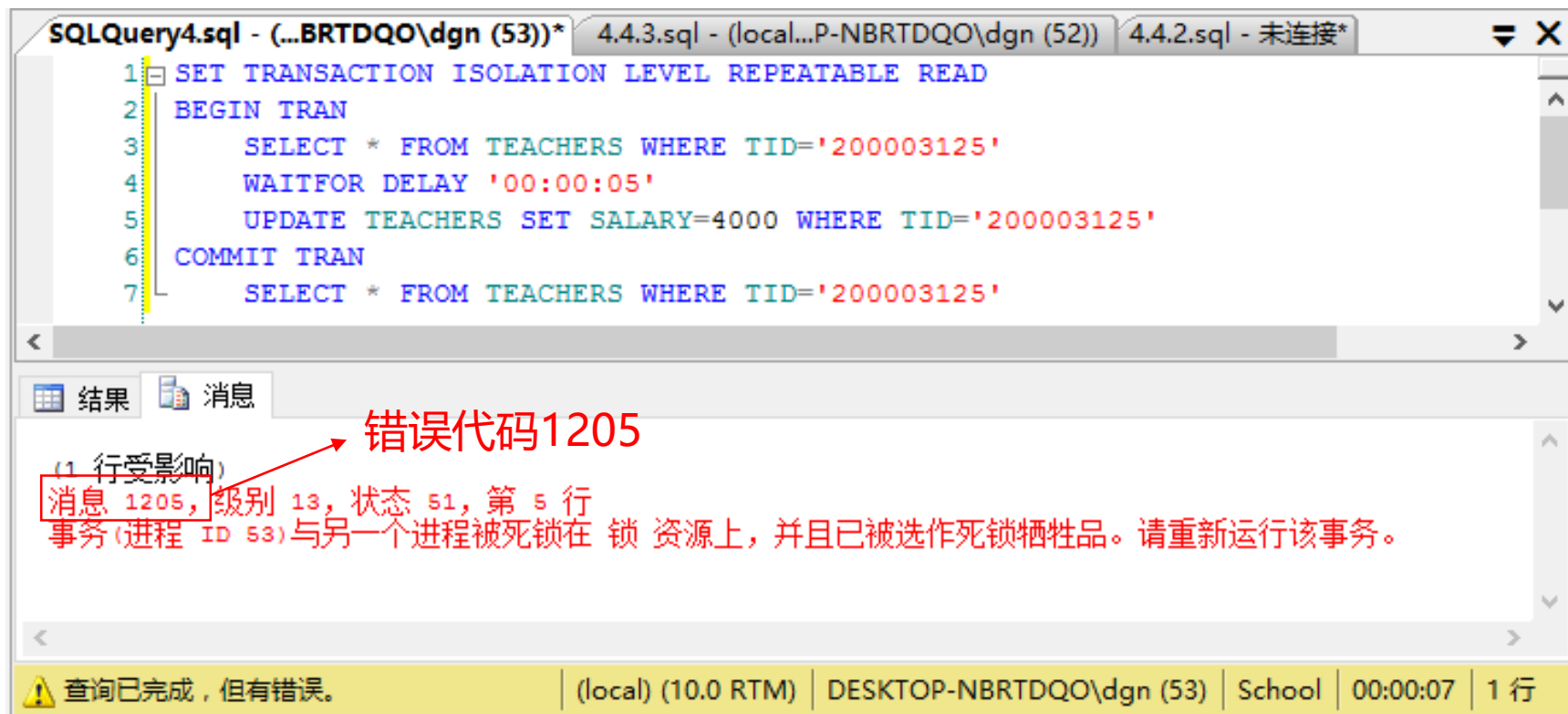
	tid	tname	email	salary
1	200003125	fqmyi	wcjcg@glq.net	4000

Red text "连接1执行成功" (Connection 1 execution successful) is overlaid on the second table.

At the bottom, a status bar shows: 查询已成功执行。 (local) (10.0 RTM) DESKTOP-NBRTDQO\dgn (52) School 00:00:10 2 行

实验示例

连接2由于死锁，报错：



出现死锁的原因：因为两个连接都通过设置共享锁（shared lock）对同一数据进行查询，并尝试转换为更新锁（update lock），进而到排它锁（exclusive）以完成更新操作。但隔离级别为“可重复读”，在事务完成之前，两个连接不可能释放共享锁而永远无法更新，因而导致死锁

- 实验示例

注：

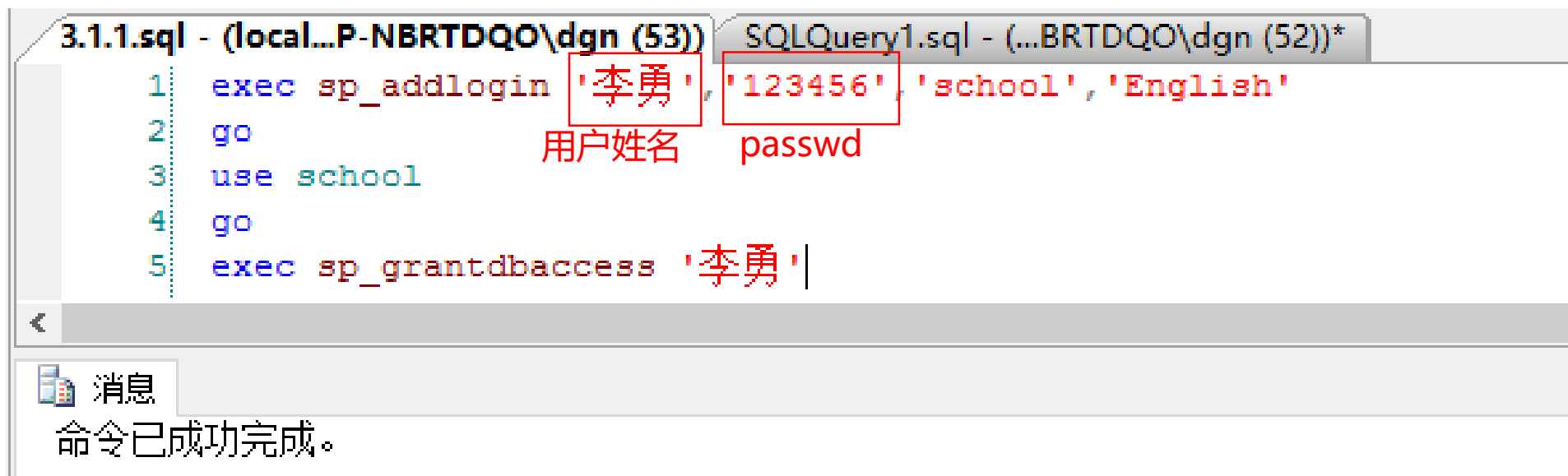
- 1.当系统发生锁争夺时，如果有事务超时，SQL Sever向用户返回错误号1222，当发生死锁时，如果有牺牲事务，SQL Sever向用户返回错误号1205，在应用时，需要在应用程序中处理锁争夺与死锁，通过在错误处理器中捕获消息1222或1205，然后让应用自动重新提交事务。
- 2.为了避免死锁，存取资源顺序最好相同。如连接A先存取甲数据库对象，再存取乙数据库对象，如果连接B的存取顺序刚好相反，则有可能发生死锁。

• 实验示例

3. 让用户李勇具有查询课程号为10010课程的权限。

创建在选课表choices上视图CS_VIEW（课程号为10010），授权给用户李勇，让李勇具有视图上select权限。

Step1：在数据库School上创建用户“李勇”。



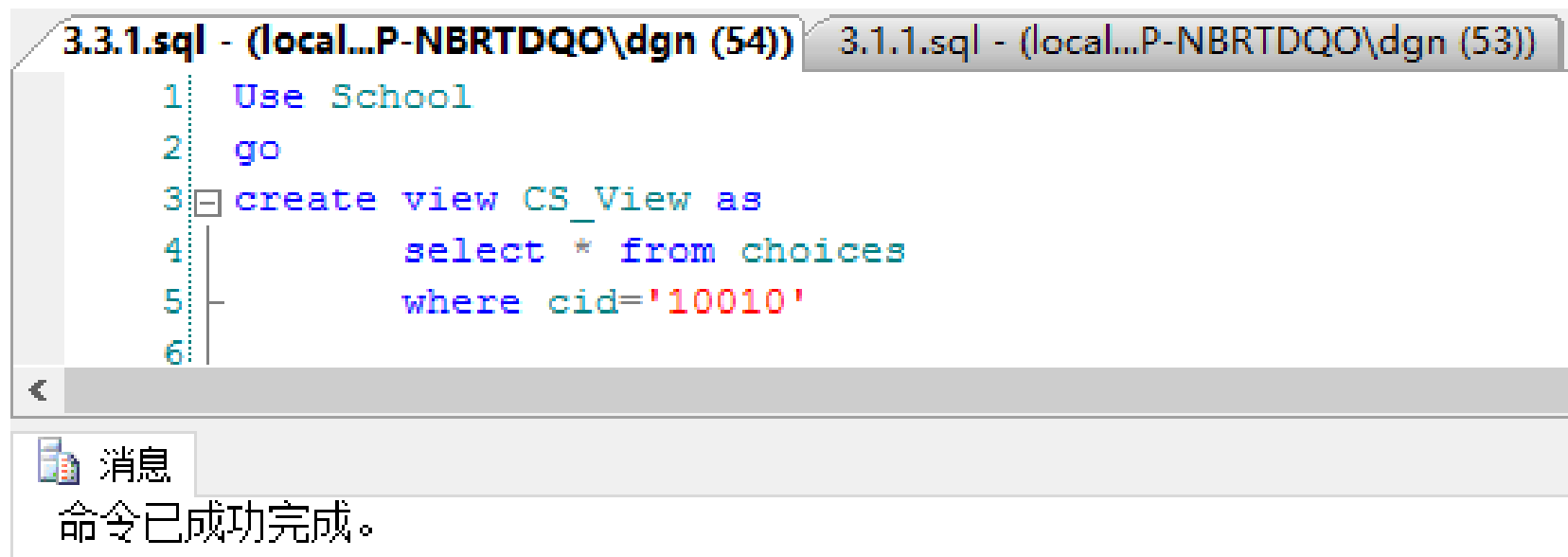
The screenshot shows a SQL Server Enterprise Manager interface. At the top, there are two tabs: '3.1.1.sql - (local...P-NBRTDQO\dgn (53))' and 'SQLQuery1.sql - (...BRTDQO\dgn (52))*'. The active window displays the following SQL commands:

```
1  exec sp_addlogin '李勇', '123456', 'school', 'English'
2  go
3  use school
4  go
5  exec sp_grantdbaccess '李勇'|
```

Red boxes highlight the values '李勇' and '123456' in the first command. Red text labels '用户姓名' (username) and 'passwd' (password) point to these values. Below the query window, a message box titled '消息' (Message) displays the text '命令已成功完成。' (Command completed successfully.).

• 实验示例

Step2: 用sa登录数据库，新建查询，在choices表上课程号为 '10010' 创建视图 CS_View。



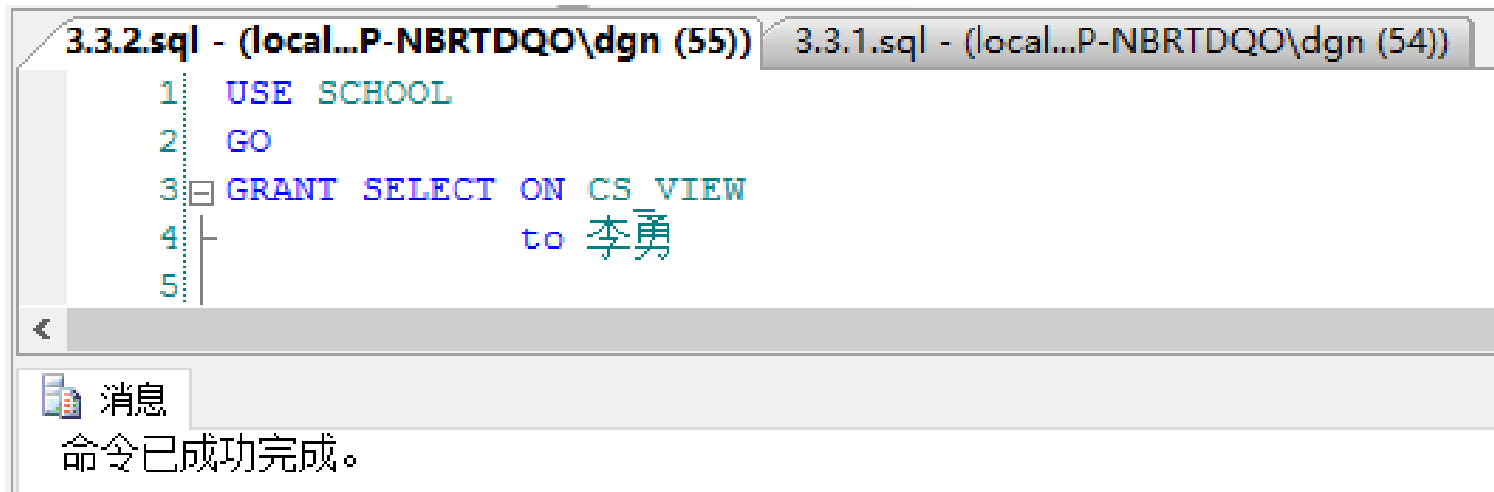
The screenshot displays the SQL Server Enterprise Manager interface. At the top, there are two tabs for query windows: '3.3.1.sql - (local...P-NBRTDQO\dgn (54))' and '3.1.1.sql - (local...P-NBRTDQO\dgn (53))'. The active window shows the following SQL code:

```
1 Use School
2 go
3 create view CS_View as
4     select * from choices
5     where cid='10010'
6
```

Below the query window, a message box is visible with the text '命令已成功完成。' (Command completed successfully.)

- 实验示例

Step3: 在视图CS_View上给用户李勇授予select的权限。



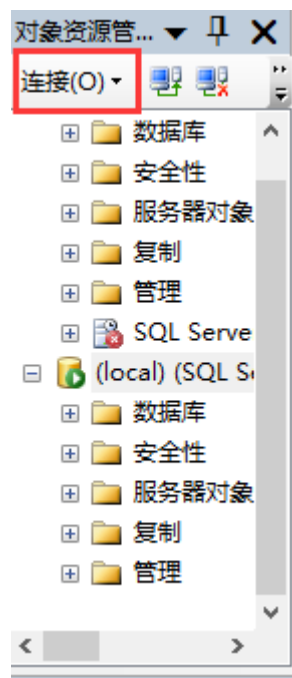
The screenshot shows a SQL query editor window with two tabs. The active tab is titled "3.3.2.sql - (local...P-NBRTDQO\dgn (55))". It contains a SQL script with five lines of code: "1 USE SCHOOL", "2 GO", "3 GRANT SELECT ON CS VIEW", "4 to 李勇", and "5". The text is color-coded: "USE", "GO", "GRANT", "ON", "VIEW", and "to" are in blue, while "SCHOOL", "CS VIEW", and "李勇" are in black. A vertical dotted line is positioned at the end of line 5. Below the code editor is a message bar with a document icon and the text "消息 命令已成功完成。". The background tab is titled "3.3.1.sql - (local...P-NBRTDQO\dgn (54))".

```
1  USE SCHOOL
2  GO
3  GRANT SELECT ON CS VIEW
4  to 李勇
5
```

消息
命令已成功完成。

• 实验示例

Step4: 以用户李勇身份登录，对CS_View视图进行查询，查询成功。



3.3.4.sql - (local).School (李勇 (58)) 3.3.3.sql - (local...P-NBRTDQO\dgn (56))

```
1 use school
2 go
3 select * from cs_view
```

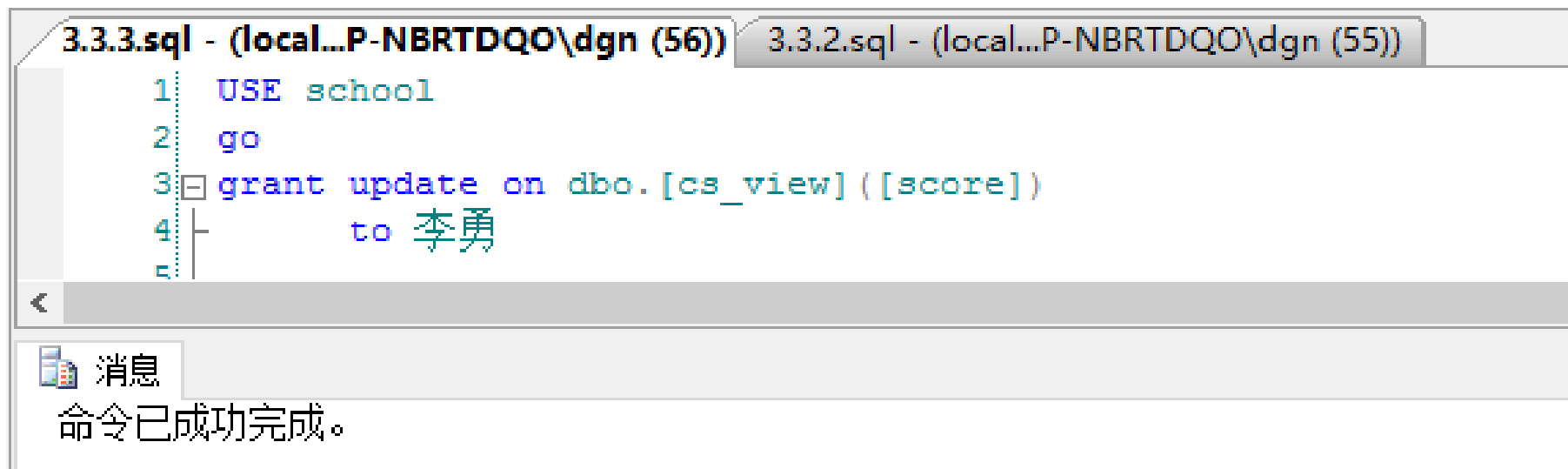
结果 消息

	no	sid	tid	cid	score
1	500000253	829348273	202560416	10010	87
2	500010915	894037661	200713929	10010	79
3	500023337	890644804	242635790	10010	88
4	500024940	829310417	221792985	10010	NULL
5	500048368	898738645	265304274	10010	77
6	500065154	821848893	254787674	10010	79
7	500077667	821494816	214751989	10010	98
8	500084641	881633930	250749054	10010	85

• 实验示例

4.对视图上score属性列的update权限授予给用户李勇，可以修改学生的成绩，但不能对学生的基本信息，例如学号，选课号进行修改。

Step1: 以sa身份登录，授予CS_View视图中score列update权限给李勇。



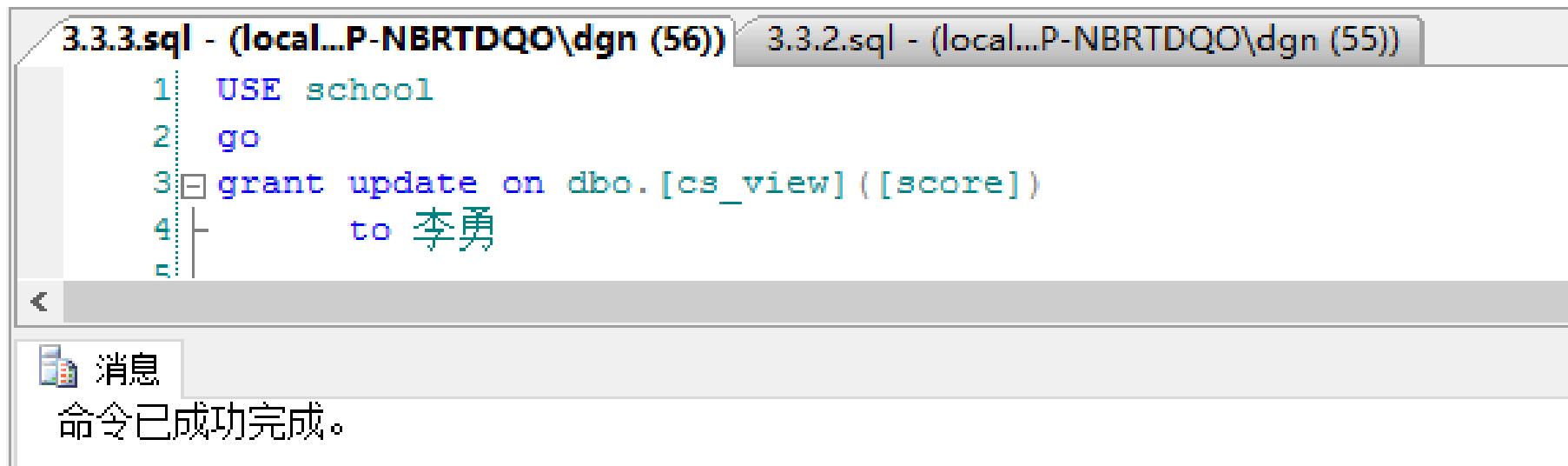
The screenshot displays the SQL Server Enterprise Manager interface. At the top, two tabs are visible: '3.3.3.sql - (local...P-NBRTDQO\dgn (56))' and '3.3.2.sql - (local...P-NBRTDQO\dgn (55))'. The active tab shows a SQL script with the following commands:

```
1 USE school
2 go
3 grant update on dbo.[cs_view] ([score])
4 to 李勇
5
```

Below the script editor, a message box is displayed with the text '命令已成功完成。' (Command completed successfully.)

• 实验示例

Step2: 再以李勇身份登录, 对学号为500024940的学生的成绩进行修改, 改为90分, 操作成功。



The screenshot displays the SQL Server Enterprise Manager interface. At the top, two tabs are visible: '3.3.3.sql - (local...P-NBRTDQO\dgn (56))' and '3.3.2.sql - (local...P-NBRTDQO\dgn (55))'. The active tab shows a SQL script with the following commands:

```
1  USE school
2  go
3  grant update on dbo.[cs_view] ([score])
4  to 李勇
5
```

Below the script editor, a message pane is visible with the text '命令已成功完成。' (Command successfully completed.)

• 实验示例

5.利用DBCC log命令从内存中直接读取当前活动的全部日志记录。

其语法为 **DBCC log ({dbid | dbname}, [, type=(-1 | 0 | 1 | 2 | 3 | 4)])**

注：

dbid dbname: 相应的数据库名或数据库的ID

type: 输出的类型,有如下6种:

0: 默认值,提供最少的信息(operation, context, transaction id)

1: 在0的基础上增加 flags,tags,row, description)。

2: 增加(object name, index name, page id, slot id)。

3: 有关操作的全部信息。

4: 有关操作的全部信息以及记录的原始数据。

-1:显示所有的信息,但返回不是记录集。

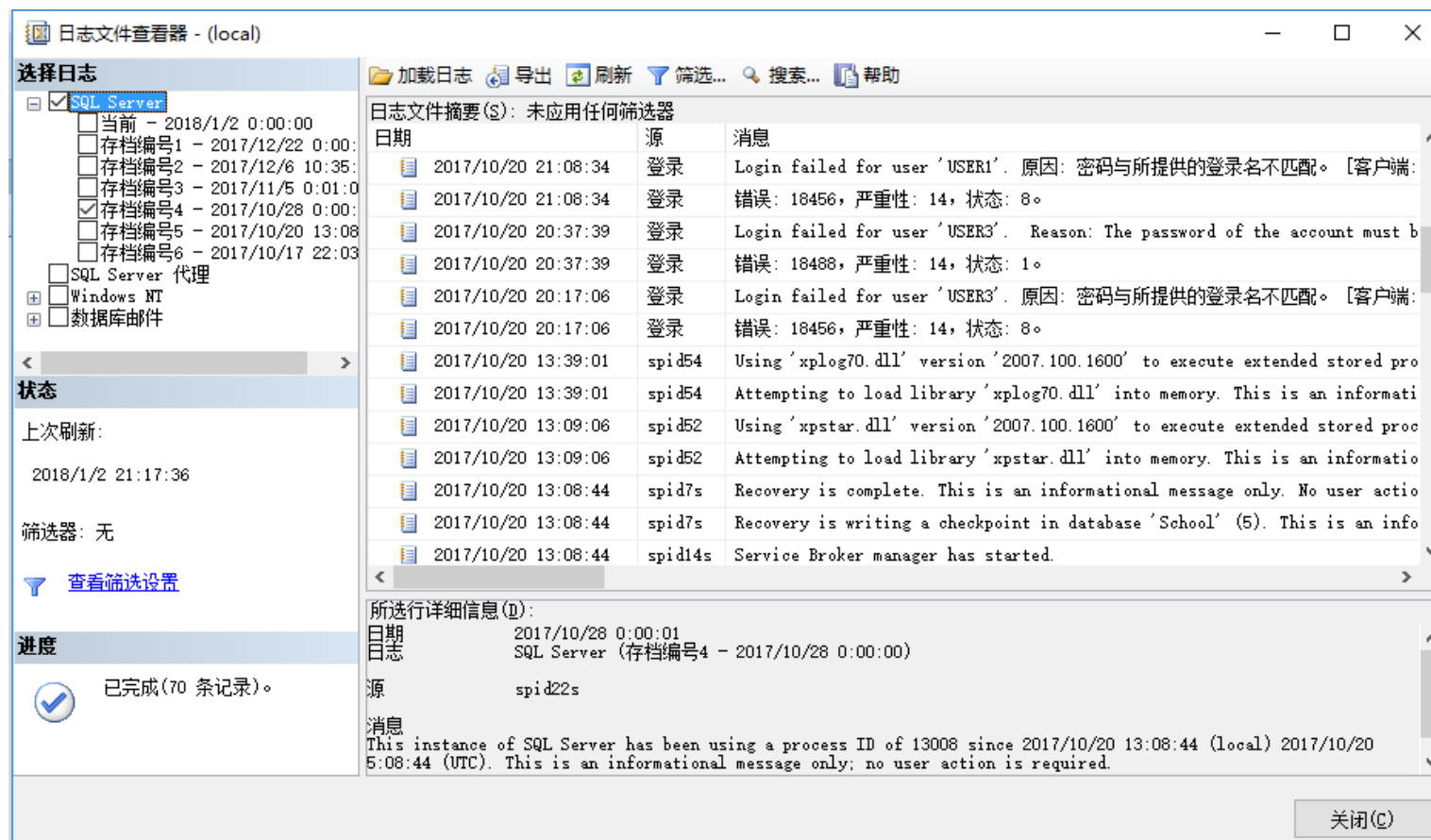
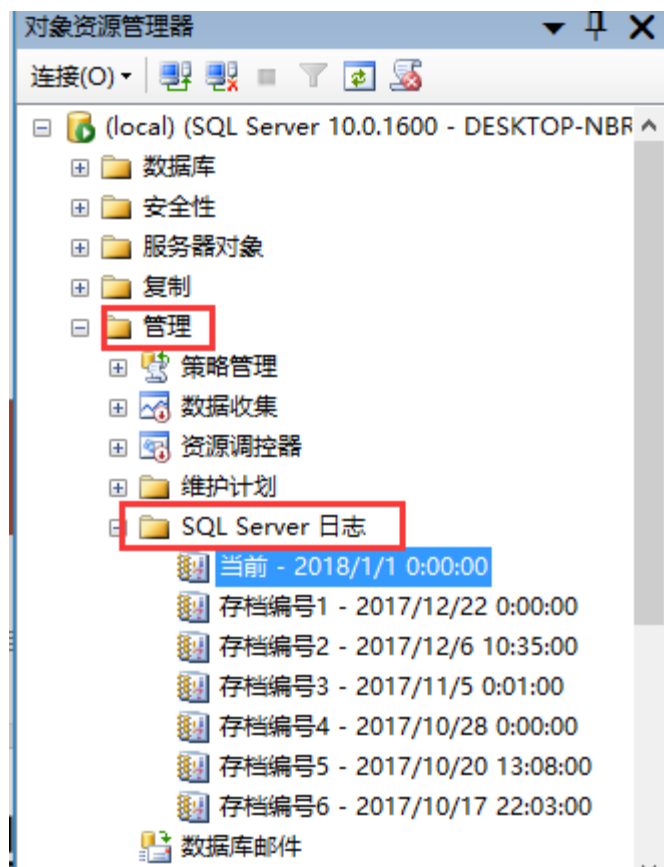
[illegible]

注：Log Record显示的十六进制，需要通过其他第三方日志分析软件才能得到直观结果，比如log explorer。¹⁶

实验示例

6.查看错误日志。

SQL Server中保留最近6个错误日志的档案，可以通过“管理”文件夹中“SQL Server日志”，选择相应存档查看日志详细信息（在“消息”列中有failed,problem等字样）。



- 练习

以下练习均在school数据库中students表上进行。

- 1.在students表上演示锁争夺，通过sp_who查看阻塞的进程。通过设置lock_timeout解除锁争夺。
- 2.在students表上演示死锁。
- 3.讨论如何避免死锁以及死锁的处理方法。
- 4.在school数据库上创建用户“王二”，在students表上创建视图grade2000，将年级为2000的学生元组放入视图。
- 5.授予用户王二在视图grade2000的select权限。
- 6.授予用户王二在视图grade2000的修改sname列的权限。
- 7.查看SQL Server错误日志。