

隔离级别



讲授内容

ANSI标准隔离级别

- READ UNCOMMITTED (读未提交)
- READ COMMITTED (读提交)
- REPEATABLE READ (可重复读)
- SERIALIZABLE (可串行化)





隔离级别中的锁协议

- 不同的隔离级别使用锁的方式不同，对应不同的封锁协议。
 - 锁被保持到事务提交的时候，称为“长期锁”
 - 锁只保持到语句执行完毕，称为“短期锁”
 - 所有的隔离级别上写锁都是长期锁
 - 读锁在每个级别上的处理方式不同



隔离级别中的锁协议

多粒度封锁模式实例

- NOLOCK: 读数据前不加锁（仅用于SELECT语句）。
- TABLOCK: 在表上加共享锁，在读完数据后**立即释放**锁。
- HOLDLOCK: 保持锁，用于TABLOCK后，将共享锁保留到**事务完成**。
- UPDLOCK: 在满足条件的元组上加更新锁。
- TABLOCKX: 在表上加排他锁。

短期锁

长期锁



隔离级别中的锁协议

- 不同的隔离级别使用锁的方式不同，对应不同的封锁协议。

隔离级别	脏读	非重复读
READ UNCOMMITTED (读未提交)	是	是
READ COMMITTED (读提交)	否	是
REPEATABLE READ (可重复读)	否	否(有幻影)
SERIALIZABLE (可串行化)	否	否(无幻影)



隔离级别类型

READ UNCOMMITTED (读未提交)

- 事务没有获得读锁也可以执行读操作。
- 事务可以读取被其他事务已经加上写锁的数据。
- 事务可能会读取没有提交事务所写的脏数据。



隔离级别类型

READ COMMITTED (读提交)

- 事务读数据前要获得数据对象上的**短期读锁**。
- 事务只会读取其他事务**提交后的数据**，不会读取脏数据。
- 事务对同一数据对象进行的再次读取操作，出现**不可重复读问题**。



隔离级别类型

REPEATABLE READ (可重复读)

- 事务要获取SELECT语句查询结果中每个元组上的**长期读锁**。
- 事务对查询结果中元组的再次查询**不存在不可重复读**问题。
- 事务中该SELECT 语句的再次执行有可能检索到前次查询没有检索到的新元组。

幻影



隔离级别类型

SERIALIZABLE (可串行化)

- 事务读操作前要获得数据对象上的**长期读锁**。
- 事务的查询操作**不会有幻影**现象。
- 事务的执行是**可串行化**的。



隔离级别的设置

● SET TRANSACTION ISOLATION LEVEL <隔离级别>

- READ UNCOMMITTED
- READ COMMITTED
- REPEATABLE READ
- SERIALIZABLE



隔离级别应用

- 事务的隔离级别是运行在该隔离级别下的事务遵守的封锁协议，与并发的其他事务的隔离级别无关，也不会影响到其他隔离级别下的事务对数据的读写方式。



隔离级别应用

- ▶ 基于创建的学生选课数据库，在DBMS上并发执行在不同隔离级别下的两个事务，分析封锁情况，并观察隔离级别对事务并发性和数据一致性的影响。

S (SNO, SN, SD, SB, SEX)

C (CNO, CN, PC)

SC (SNO, CNO, GRADE)

学生选课数据库



隔离级别应用

- ▶ 基于创建的学生选课数据库，在DBMS上并发执行在不同隔离级别下的两个事务，分析封锁情况，并观察隔离级别对事务并发性和数据一致性的影响。

【例1】两个事务在默认隔离级别（READ COMMITTED读提交）下并发执行。



隔离级别应用

The screenshot displays the Microsoft SQL Server Management Studio interface. The left pane shows the 'Object Explorer' with the 'TRANSACTIONS' database selected. The right pane shows the 'Query Editor' with the following SQL query:

```
BEGIN TRAN T1  
SELECT * FROM sc;
```

The 'Results' pane at the bottom shows the execution results of the query, displaying a table with 3 rows and 3 columns (SID, CID, GRADE):

	SID	CID	GRADE
1	301	002	94.0
2	302	001	79.0
3	302	002	89.0

The status bar at the bottom indicates the query was executed successfully on the 'TRANSACTIONS' database.



隔离级别应用

The screenshot displays the Microsoft SQL Server Enterprise Manager interface. The left pane shows the database structure for 'AA\SQLEXPRESS (SQL Server 12.0.2000 - SA)'. The right pane shows a transaction script with the following SQL code:

```
BEGIN TRAN T2
UPDATE sc SET grade=grade+2 WHERE sno='s01';
SELECT * FROM sc;
```

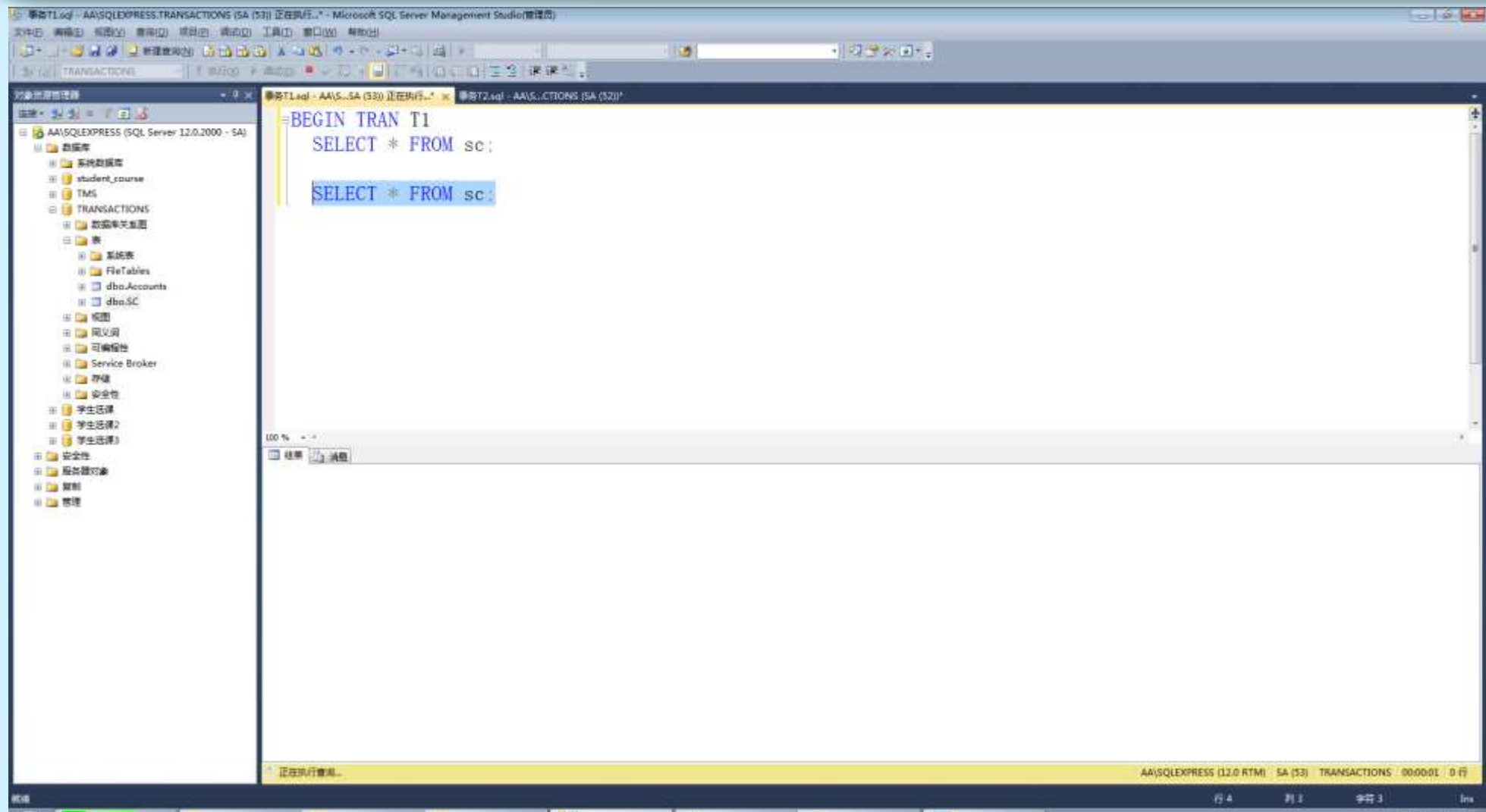
Below the script, the 'Results' tab shows the output of the SELECT statement:

	SNO	CNO	GRADE
1	S01	C01	96.0
2	S02	C01	79.0
3	S02	C02	89.0

The status bar at the bottom indicates: '查询已成功执行。' (Query executed successfully.) and 'AA\SQLEXPRESS (12.0 RTM) - SA (52) - TRANSACTIONS - 00:00:00 - 3 行' (3 rows).



隔离级别应用





隔离级别应用

Transaction T2 (Foreground):

```
BEGIN TRAN T2
UPDATE sc SET grade=grade-1 WHERE scid=101
SELECT * FROM sc;
COMMIT TRAN T2
```

Transaction T1 (Background):

```
BEGIN TRAN T1
SELECT * FROM sc;
SELECT * FROM sc;
```

Results of Transaction T1:

id	cid	grade
1	101	98.0
2	102	78.0
3	103	88.0



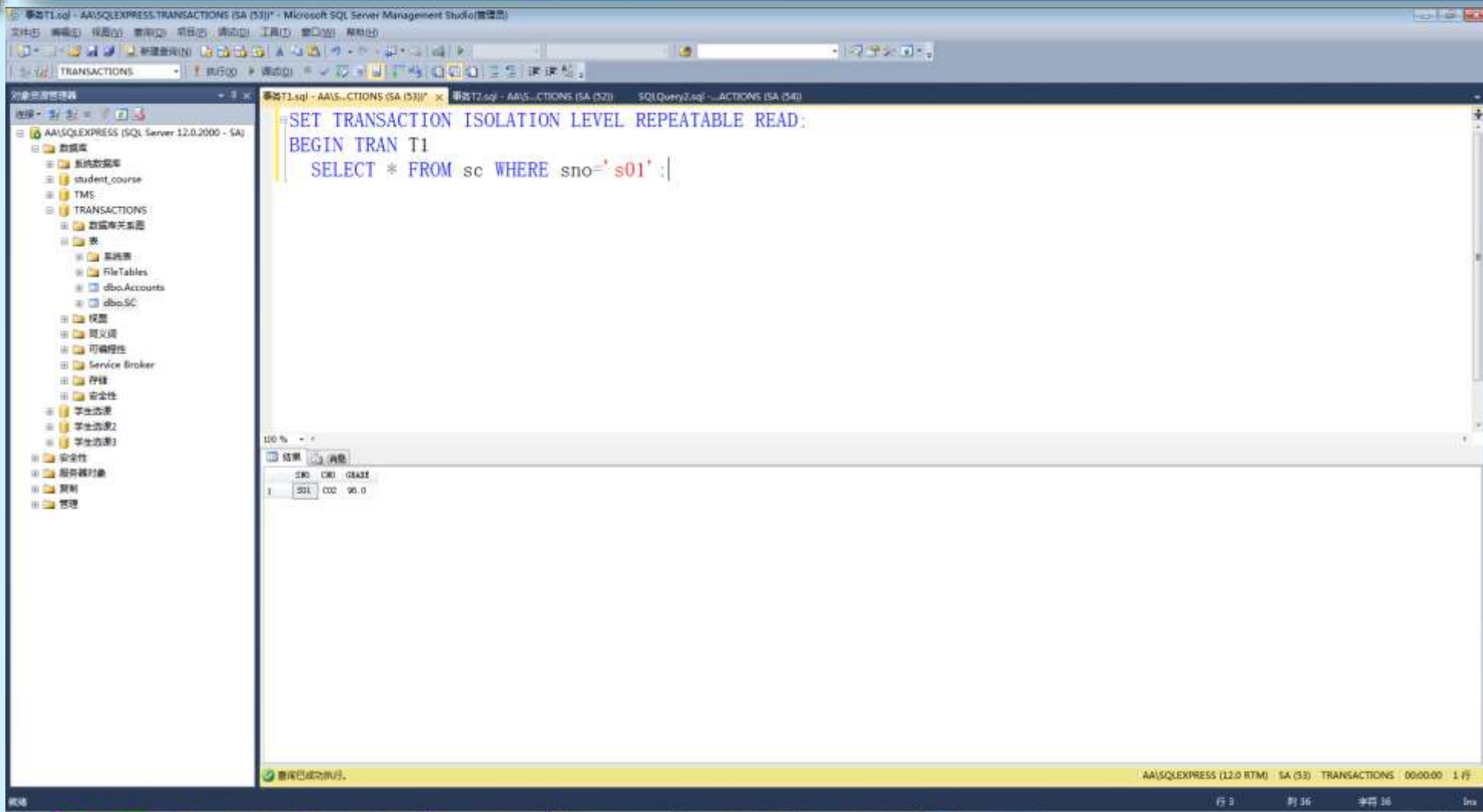
隔离级别应用

- ▶ 基于创建的学生选课数据库，在DBMS上并发执行在不同隔离级别下的两个事务，分析封锁情况，并观察隔离级别对事务并发性和数据一致性的影响。

【例2】一个在REPEATABLE READ可重复读隔离级别下的事务，与两个在默认隔离级别（READ COMMITTED读提交）下的事务并发执行。



隔离级别应用





隔离级别应用

Microsoft SQL Server Management Studio (管理员)

TRANSACTIONS

开始 T1.sql: AA\S...CTIONS (SA (52)) * 事务 T2.sql: AA\S...CTIONS (SA (52)) * SQLQuery2.sql: ...CTIONS (SA (54))

```
BEGIN TRAN T2
SELECT * FROM sc WHERE sno='s02';
UPDATE SC SET GRADE= GRADE+2 WHERE SNO ='s02';
SELECT * FROM SC WHERE SNO='s02';
```

100 %

	SNO	CNO	GRADE
1	s02	C01	79.0
2	s02	C02	89.0

	SNO	CNO	GRADE
1	s02	C01	81.0
2	s02	C02	91.0

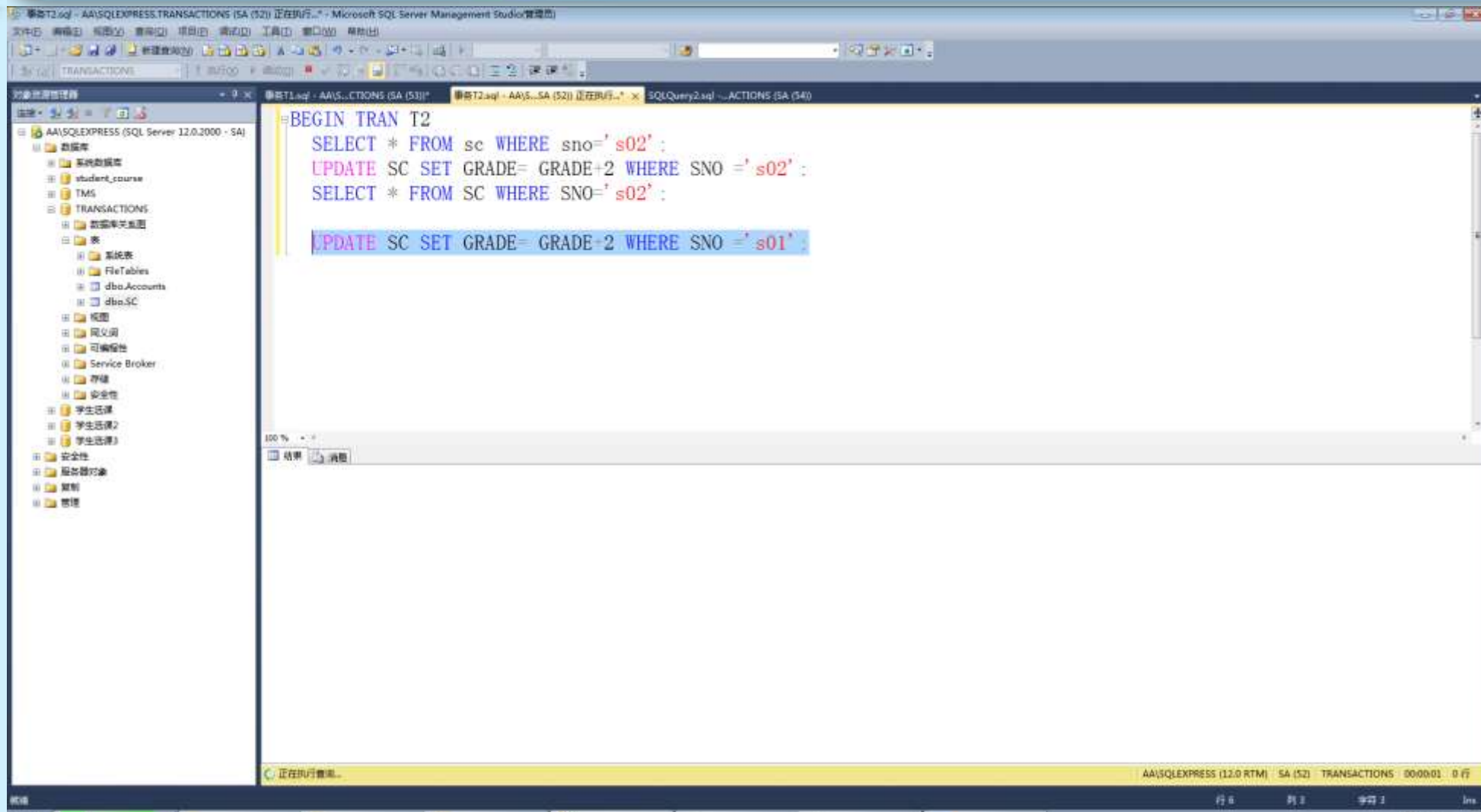
查询已成功执行。

AA\SQLEXPRESS (12.0 RTM) SA (52) TRANSACTIONS 00.0000 4 行

行 4 列 36 字节 36 3ms

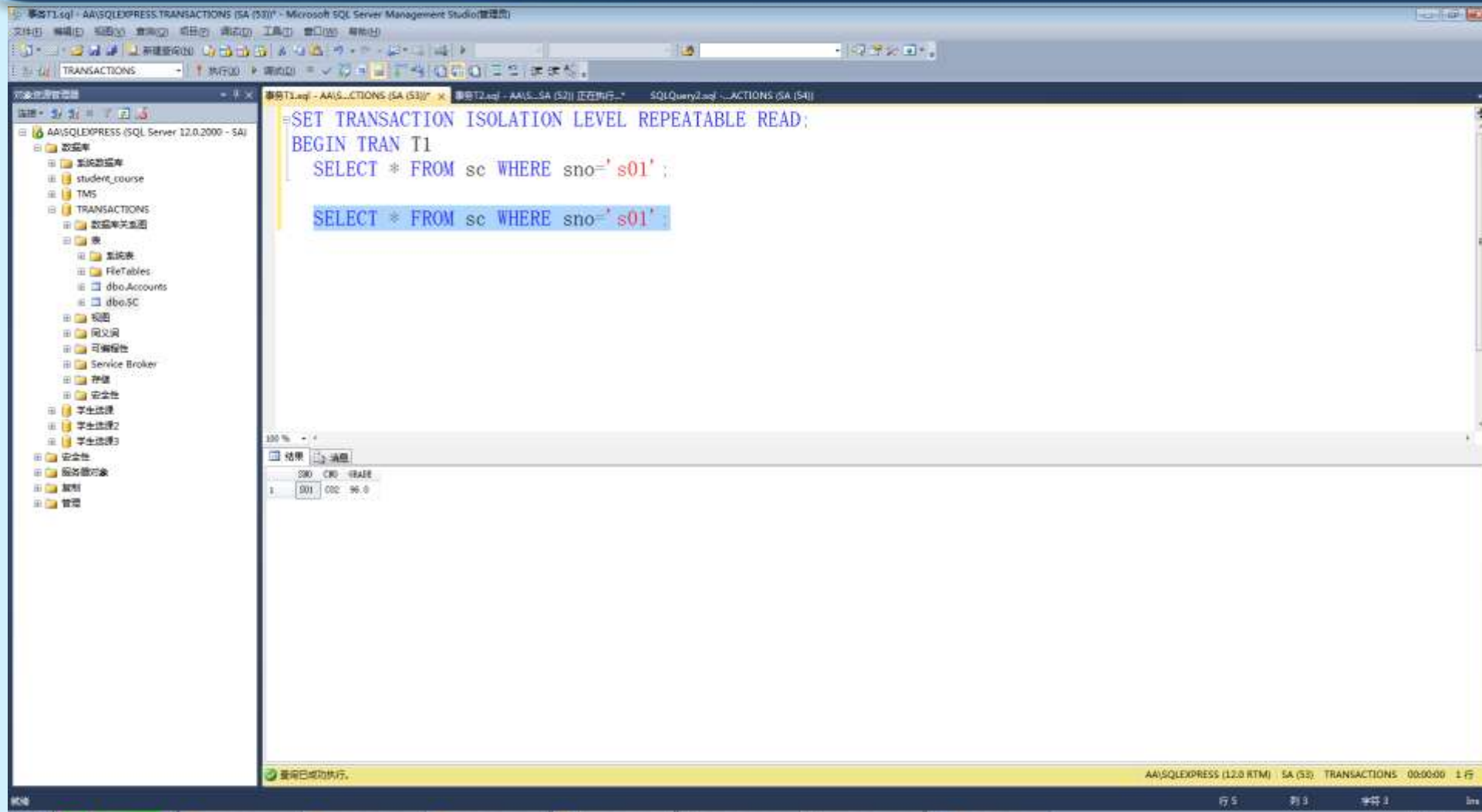


隔离级别应用



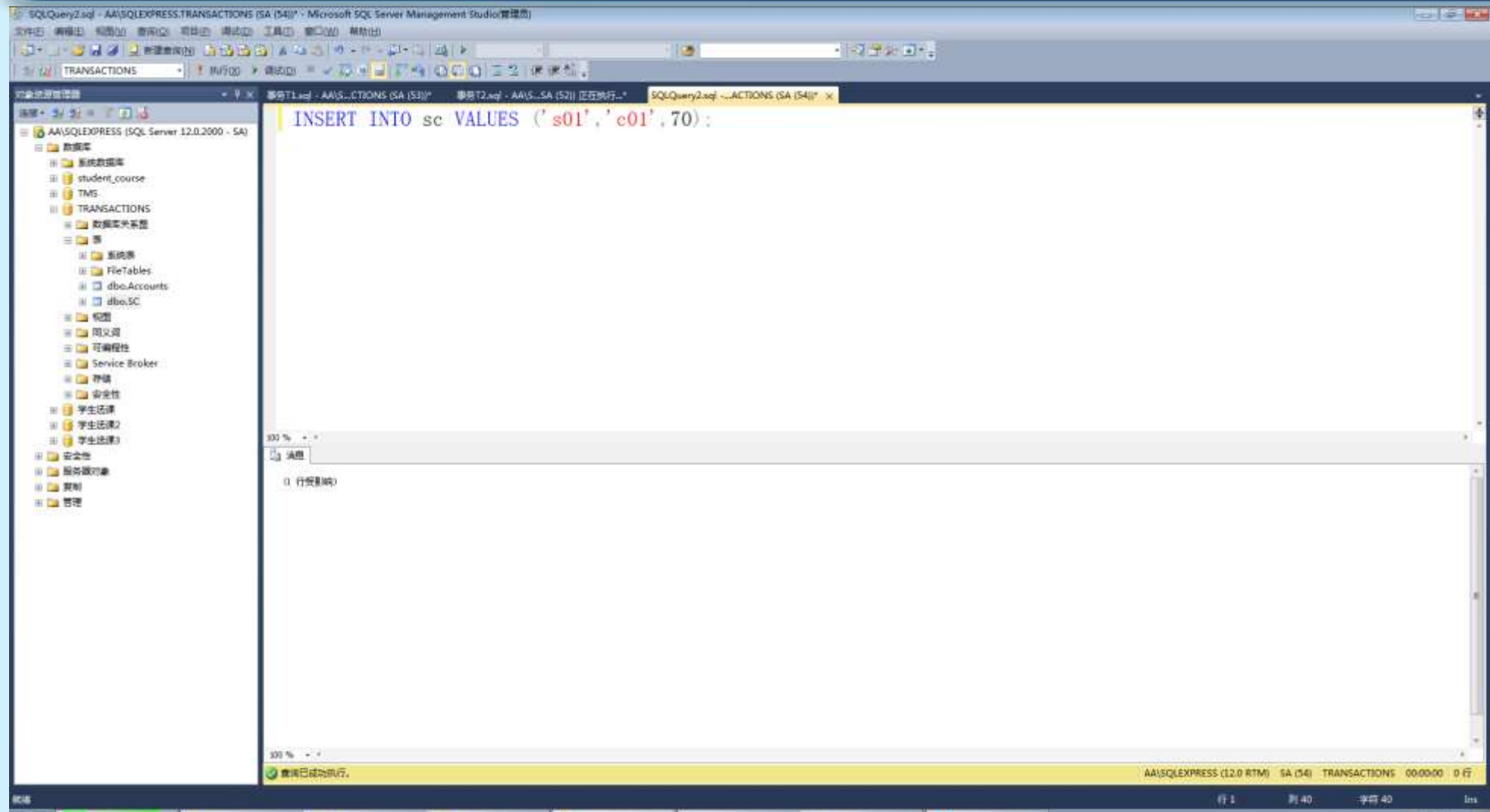


隔离级别应用





隔离级别应用





隔离级别应用

The screenshot displays the Microsoft SQL Server Enterprise Manager interface. The left pane shows the server structure for 'AA\SQLEXPRESS (SQL Server 12.0.2000 - SA)'. The right pane shows a SQL script in the 'TRANSACTIONS' database:

```
SET TRANSACTION ISOLATION LEVEL REPEATABLE READ;  
BEGIN TRAN T1  
  SELECT * FROM sc WHERE sno='s01';  
  
  SELECT * FROM sc WHERE sno='s01';  
  
  SELECT * FROM sc WHERE sno='s01';
```

Below the script, the 'Results' pane shows the output of the third query:

	SNO	CNO	GRADE
1	s01	001	70.0
2	s01	002	96.0

The status bar at the bottom indicates: 'AA\SQLEXPRESS (12.0 RTM) SA (53) TRANSACTIONS 00:00:00 2 行'.



隔离级别应用

Transaction T1: AA\SQLEXPRESS\TRANSACTIONS (SA (53)) - Microsoft SQL Server Management Studio (管理)

```
SET TRANSACTION ISOLATION LEVEL REPEATABLE READ;  
BEGIN TRAN T1  
SELECT * FROM sc WHERE sno='s01';  
SELECT * FROM sc WHERE sno='s01';  
SELECT * FROM sc WHERE sno='s01';  
COMMIT
```

Transaction T2: AA\SQLEXPRESS\TRANSACTIONS (SA (52)) - Microsoft SQL Server Management Studio (管理)

```
BEGIN TRAN T2  
SELECT * FROM sc WHERE sno='s02';  
UPDATE SC SET GRADE= GRADE+2 WHERE SNO ='s02';  
SELECT * FROM SC WHERE SNO='s02';  
UPDATE SC SET GRADE= GRADE+2 WHERE SNO ='s01';
```

100 % - 1
命令已成功执行。

100 % - 1
命令已成功执行。

AA\SQLEXPRESS (12.0 RTM) SA (52) TRANSACTIONS 2010/02/09



隔离级别应用

The screenshot displays the Microsoft SQL Server Management Studio interface. The left pane shows the database structure for 'AA\SQLEXPRESS (SQL Server 12.0.2000 - SA)'. The central query editor contains the following SQL script:

```
BEGIN TRAN T2
SELECT * FROM SC WHERE sno='s02';
UPDATE SC SET GRADE= GRADE+2 WHERE SNO ='s02';
SELECT * FROM SC WHERE SNO='s02';

UPDATE SC SET GRADE= GRADE+2 WHERE SNO ='s01';

SELECT * FROM SC WHERE sno='s01';
```

The bottom pane shows the results of the execution:

行	列1	列2	GRADE
1	s01	s01	12.0
2	s01	s02	60.0

The status bar at the bottom indicates: 'AA\SQLEXPRESS (12.0 RTM) SA (52) TRANSACTIONS 00:00:00 2 行'.




小结



隔离级别表示一个事务所能容忍的被其他事务干扰的程度。

隔离级别	脏读	非重复读
READ UNCOMMITTED (读未提交)	是	是
READ COMMITTED (读提交)	否	是
REPEATABLE READ (可重复读)	否	否(有幻影)
SERIALIZABLE (可串行化)	否	否(无幻影)

- 
- 隔离级别增高
 - 隔离性能增强
 - 干扰程度降低
 - 数据不一致问题减少
 - 事务并发程度降低