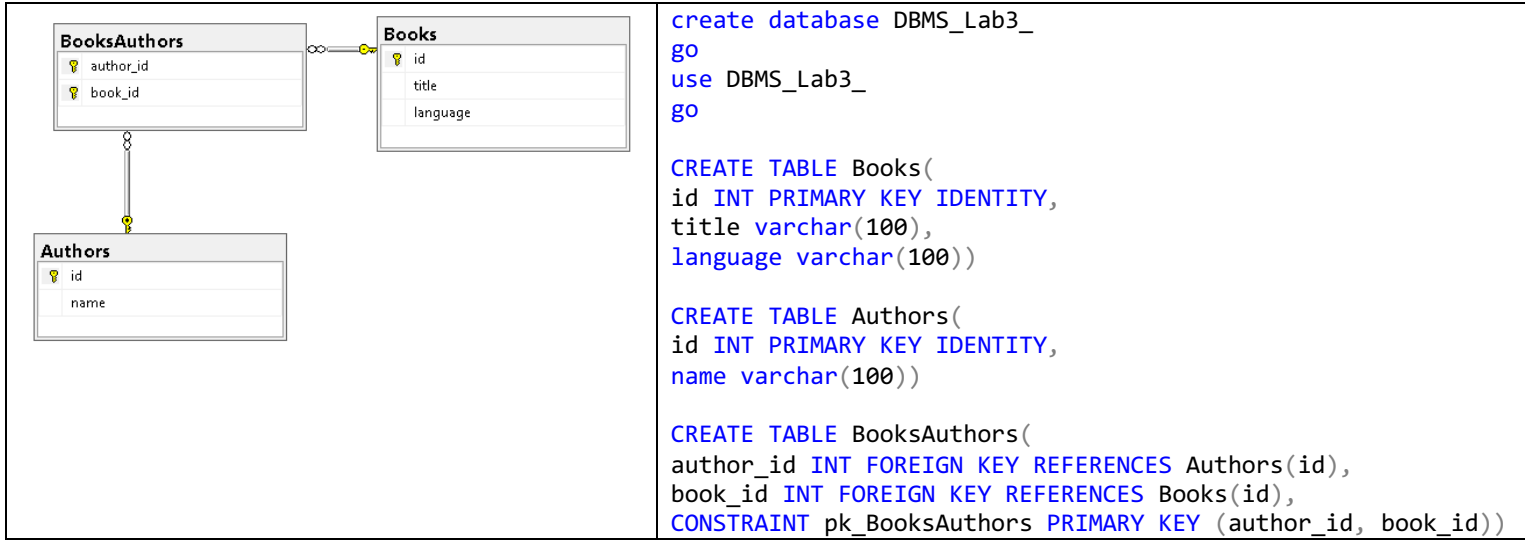


Concurrency problems – a simple example

We consider the database



In table Books we have

```
-- add records
insert into Books values
('Pride and Prejudice', 'english'),
('Harry Potter and the Chamber of Secrets', 'english'),
('Panda', 'english')

select * from Books
```

	id	title	language
1	1	Pride and Prejudice	english
2	2	Harry Potter and the Chamber of Secrets	english
3	3	Panda	english

Please, put in each file `use DBMS_Lab3_`

In what follows, we will work with the table **Books** and

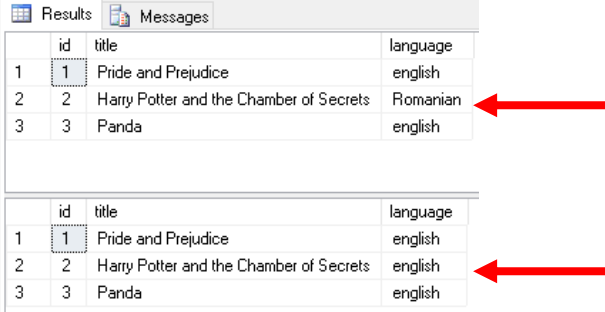
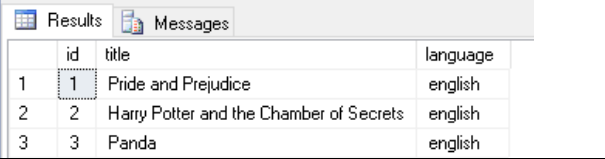
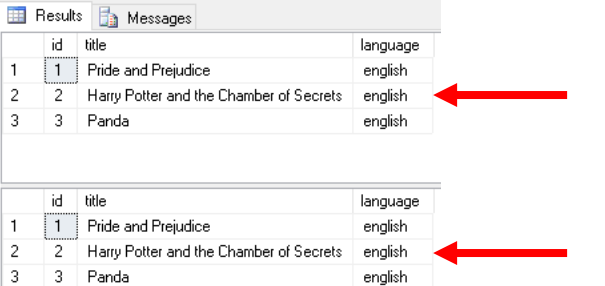
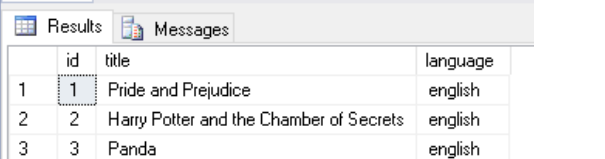
T1=Transaction 1 starts first and finish first (the first column form the table(s))

T2=Transaction start immediately after T1 and finish after T1 (the second column form the table(s))

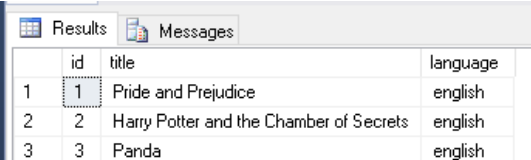
1. **DIRTY READS** – T1: update + delay + rollback, T2: select + delay + select -> see the update in the first select, even if it is rollback then (the order in the execution of the operations is: update – select – rollback – select)

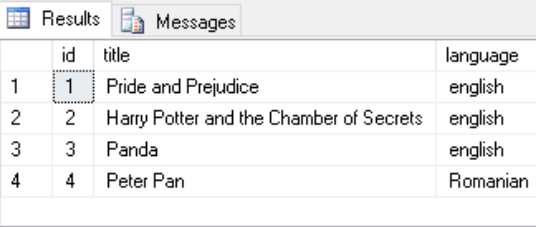
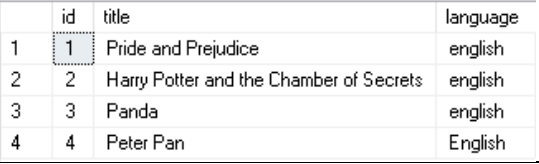
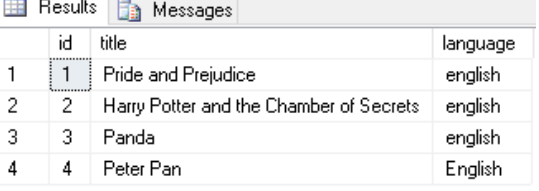
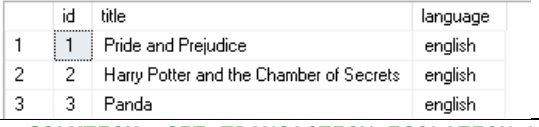
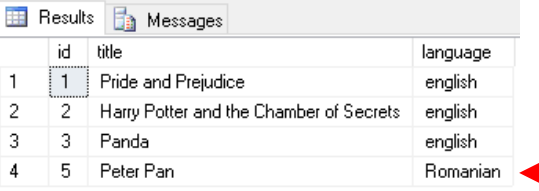
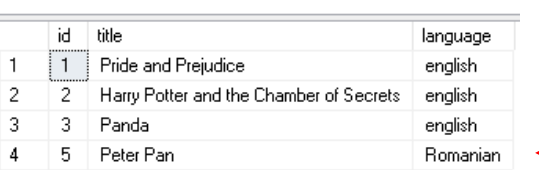
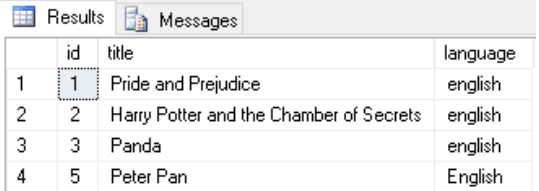
Isolation level: Read Uncommitted / Read Committed (solution)

<pre>-- dirty reads select * from Books</pre>	<table><tr><th colspan="3">Results</th></tr><tr><th>id</th><th>title</th><th>language</th></tr><tr><td>1</td><td>Pride and Prejudice</td><td>english</td></tr><tr><td>2</td><td>Harry Potter and the Chamber of Secrets</td><td>english</td></tr><tr><td>3</td><td>Panda</td><td>english</td></tr></table>	Results			id	title	language	1	Pride and Prejudice	english	2	Harry Potter and the Chamber of Secrets	english	3	Panda	english
Results																
id	title	language														
1	Pride and Prejudice	english														
2	Harry Potter and the Chamber of Secrets	english														
3	Panda	english														
<pre>BEGIN TRANSACTION UPDATE Books SET language='Romanian' WHERE id = 2 WAITFOR DELAY '00:00:10' ROLLBACK TRANSACTION</pre>	<pre>--PROBLEM: SET TRANSACTION ISOLATION LEVEL READ UNCOMMITTED SET TRANSACTION ISOLATION LEVEL READ UNCOMMITTED BEGIN TRAN SELECT * FROM Books WAITFOR DELAY '00:00:15' SELECT * FROM Books COMMIT TRAN</pre>															




(1 row(s) affected)	
-- dirty reads select * from Books	
BEGIN TRANSACTION UPDATE Books SET language='Romanian' WHERE id = 2 WAITFOR DELAY '00:00:10' ROLLBACK TRANSACTION	-- SOLUTION: SET TRANSACTION ISOLATION LEVEL TO READ COMMITTED SET TRANSACTION ISOLATION LEVEL READ COMMITTED BEGIN TRAN SELECT * FROM Books WAITFOR DELAY '00:00:15' SELECT * FROM Books COMMIT TRAN
(1 row(s) affected)	
-- dirty reads select * from Books	

2. **NON-REPEATABLE READS** – T1: delay + update + commit, T2: select + delay + select -> see the value inserted before the transaction from the first select of T2 + see the update of the value inserted before the transaction, from the second select of T2 (the order in the execution of the operations is: select – update – select)
Isolation level: Read Committed / Repeatable Read (solution)

-- nonrepeatable reads select * from Books	
INSERT INTO Books(title, language) VALUES ('Peter Pan', 'Romanian') BEGIN TRAN WAITFOR DELAY '00:00:10' UPDATE Books SET language='English' WHERE title = 'Peter Pan' COMMIT TRAN	-- PROBLEM: SET TRANSACTION ISOLATION LEVEL READ UNCOMMITTED SET TRANSACTION ISOLATION LEVEL READ COMMITTED BEGIN TRAN SELECT * FROM Books WAITFOR DELAY '00:00:15' SELECT * FROM Books COMMIT TRAN

<pre>(1 row(s) affected) -- from insert</pre> <pre>(1 row(s) affected) - from update</pre>	 
<pre>-- nonrepeatable reads</pre> <pre>select * from Books</pre> <pre>delete from Books where title = 'Peter Pan'</pre> <pre>select * from Books</pre>	 
<pre>INSERT INTO Books(title, language) VALUES ('Peter Pan', 'Romanian')</pre> <pre>BEGIN TRAN</pre> <pre>WAITFOR DELAY '00:00:10'</pre> <pre>UPDATE Books SET language='English' WHERE title = 'Peter Pan'</pre> <pre>COMMIT TRAN</pre>	<pre>-- SOLUTION: SET TRANSACTION ISOLATION LEVEL TO REPEATABLE READ</pre> <pre>SET TRANSACTION ISOLATION LEVEL REPEATABLE READ</pre> <pre>BEGIN TRAN</pre> <pre>SELECT * FROM Books</pre> <pre>WAITFOR DELAY '00:00:15'</pre> <pre>SELECT * FROM Books</pre> <pre>COMMIT TRAN</pre>
<pre>(1 row(s) affected)</pre> <pre>(1 row(s) affected)</pre>	 
<pre>-- nonrepeatable reads</pre> <pre>select * from Books</pre>	

3. **PHANTOM READS** – T1: delay + insert + commit, T2: select + delay + select -> see the inserted value only at the second select from T2 (the order in the execution of the operations is: select – insert – select)
Isolation level: Repeatable Read / Serializable (solution)

<pre>-- phantom reads select * from Books</pre>	<div>Results Messages</div> <table><tr><th></th><th>id</th><th>title</th><th>language</th></tr><tr><td>1</td><td>1</td><td>Pride and Prejudice</td><td>english</td></tr><tr><td>2</td><td>2</td><td>Harry Potter and the Chamber of Secrets</td><td>english</td></tr><tr><td>3</td><td>3</td><td>Panda</td><td>english</td></tr><tr><td>4</td><td>5</td><td>Peter Pan</td><td>English</td></tr></table>		id	title	language	1	1	Pride and Prejudice	english	2	2	Harry Potter and the Chamber of Secrets	english	3	3	Panda	english	4	5	Peter Pan	English																								
	id	title	language																																										
1	1	Pride and Prejudice	english																																										
2	2	Harry Potter and the Chamber of Secrets	english																																										
3	3	Panda	english																																										
4	5	Peter Pan	English																																										
<pre>BEGIN TRAN WAITFOR DELAY '00:00:10' INSERT INTO Books(title,language) VALUES ('At the end of the world','English') COMMIT TRAN</pre>	<pre>-- PROBLEM: SET TRANSACTION ISOLATION LEVEL TO REPEATABLE READ SET TRANSACTION ISOLATION LEVEL REPEATABLE READ BEGIN TRAN SELECT * FROM Books WAITFOR DELAY '00:00:15' SELECT * FROM Books COMMIT TRAN</pre>																																												
(1 row(s) affected)	<div>Results Messages</div> <table><tr><th></th><th>id</th><th>title</th><th>language</th></tr><tr><td>1</td><td>1</td><td>Pride and Prejudice</td><td>english</td></tr><tr><td>2</td><td>2</td><td>Harry Potter and the Chamber of Secrets</td><td>english</td></tr><tr><td>3</td><td>3</td><td>Panda</td><td>english</td></tr><tr><td>4</td><td>5</td><td>Peter Pan</td><td>English</td></tr></table> <div>Results Messages</div> <table><tr><th></th><th>id</th><th>title</th><th>language</th></tr><tr><td>1</td><td>1</td><td>Pride and Prejudice</td><td>english</td></tr><tr><td>2</td><td>2</td><td>Harry Potter and the Chamber of Secrets</td><td>english</td></tr><tr><td>3</td><td>3</td><td>Panda</td><td>english</td></tr><tr><td>4</td><td>5</td><td>Peter Pan</td><td>English</td></tr><tr><td>5</td><td>6</td><td>At the end of the world</td><td>English</td></tr></table> 		id	title	language	1	1	Pride and Prejudice	english	2	2	Harry Potter and the Chamber of Secrets	english	3	3	Panda	english	4	5	Peter Pan	English		id	title	language	1	1	Pride and Prejudice	english	2	2	Harry Potter and the Chamber of Secrets	english	3	3	Panda	english	4	5	Peter Pan	English	5	6	At the end of the world	English
	id	title	language																																										
1	1	Pride and Prejudice	english																																										
2	2	Harry Potter and the Chamber of Secrets	english																																										
3	3	Panda	english																																										
4	5	Peter Pan	English																																										
	id	title	language																																										
1	1	Pride and Prejudice	english																																										
2	2	Harry Potter and the Chamber of Secrets	english																																										
3	3	Panda	english																																										
4	5	Peter Pan	English																																										
5	6	At the end of the world	English																																										
<pre>-- phantom reads select * from Books delete from Books where title='At the end of the world' select * from Books</pre>	<div>Results Messages</div> <table><tr><th></th><th>id</th><th>title</th><th>language</th></tr><tr><td>1</td><td>1</td><td>Pride and Prejudice</td><td>english</td></tr><tr><td>2</td><td>2</td><td>Harry Potter and the Chamber of Secrets</td><td>english</td></tr><tr><td>3</td><td>3</td><td>Panda</td><td>english</td></tr><tr><td>4</td><td>5</td><td>Peter Pan</td><td>English</td></tr><tr><td>5</td><td>6</td><td>At the end of the world</td><td>English</td></tr></table> <div>Results Messages</div> <table><tr><th></th><th>id</th><th>title</th><th>language</th></tr><tr><td>1</td><td>1</td><td>Pride and Prejudice</td><td>english</td></tr><tr><td>2</td><td>2</td><td>Harry Potter and the Chamber of Secrets</td><td>english</td></tr><tr><td>3</td><td>3</td><td>Panda</td><td>english</td></tr><tr><td>4</td><td>5</td><td>Peter Pan</td><td>English</td></tr></table>		id	title	language	1	1	Pride and Prejudice	english	2	2	Harry Potter and the Chamber of Secrets	english	3	3	Panda	english	4	5	Peter Pan	English	5	6	At the end of the world	English		id	title	language	1	1	Pride and Prejudice	english	2	2	Harry Potter and the Chamber of Secrets	english	3	3	Panda	english	4	5	Peter Pan	English
	id	title	language																																										
1	1	Pride and Prejudice	english																																										
2	2	Harry Potter and the Chamber of Secrets	english																																										
3	3	Panda	english																																										
4	5	Peter Pan	English																																										
5	6	At the end of the world	English																																										
	id	title	language																																										
1	1	Pride and Prejudice	english																																										
2	2	Harry Potter and the Chamber of Secrets	english																																										
3	3	Panda	english																																										
4	5	Peter Pan	English																																										
<pre>BEGIN TRAN WAITFOR DELAY '00:00:10' INSERT INTO Books(title,language) VALUES ('At the end of the world','English') COMMIT TRAN</pre>	<pre>-- SOLUTION: SET TRANSACTION ISOLATION LEVEL TO SERIALIZABLE SET TRANSACTION ISOLATION LEVEL SERIALIZABLE BEGIN TRAN SELECT * FROM Books WAITFOR DELAY '00:00:15' SELECT * FROM Books COMMIT TRAN</pre>																																												
(1 row(s) affected)	<div>Results Messages</div> <table><tr><th></th><th>id</th><th>title</th><th>language</th></tr><tr><td>1</td><td>1</td><td>Pride and Prejudice</td><td>english</td></tr><tr><td>2</td><td>2</td><td>Harry Potter and the Chamber of Secrets</td><td>english</td></tr><tr><td>3</td><td>3</td><td>Panda</td><td>english</td></tr><tr><td>4</td><td>5</td><td>Peter Pan</td><td>English</td></tr></table>  <div>Results Messages</div> <table><tr><th></th><th>id</th><th>title</th><th>language</th></tr><tr><td>1</td><td>1</td><td>Pride and Prejudice</td><td>english</td></tr><tr><td>2</td><td>2</td><td>Harry Potter and the Chamber of Secrets</td><td>english</td></tr><tr><td>3</td><td>3</td><td>Panda</td><td>english</td></tr><tr><td>4</td><td>5</td><td>Peter Pan</td><td>English</td></tr></table> 		id	title	language	1	1	Pride and Prejudice	english	2	2	Harry Potter and the Chamber of Secrets	english	3	3	Panda	english	4	5	Peter Pan	English		id	title	language	1	1	Pride and Prejudice	english	2	2	Harry Potter and the Chamber of Secrets	english	3	3	Panda	english	4	5	Peter Pan	English				
	id	title	language																																										
1	1	Pride and Prejudice	english																																										
2	2	Harry Potter and the Chamber of Secrets	english																																										
3	3	Panda	english																																										
4	5	Peter Pan	English																																										
	id	title	language																																										
1	1	Pride and Prejudice	english																																										
2	2	Harry Potter and the Chamber of Secrets	english																																										
3	3	Panda	english																																										
4	5	Peter Pan	English																																										

```
-- phantom reads
select * from Books
```

Results		
id	title	language
1	Pride and Prejudice	english
2	Harry Potter and the Chamber of Secrets	english
3	Panda	english
4	Peter Pan	English
5	At the end of the world	English

DEADLOCK - T1: update on table A + delay + update on table B
 - T2: update on table B + delay + update on table A

T1- update on table A -> exclusive lock on table A	delay	Try to update table B	Table B is blocked in T2	One of the blocked transactions, T1 or T2, will be chosen as a deadlock victim and terminates with an error. The other transaction wins and update both table A and table B
T2 - update on table B -> exclusive lock on table B	delay	Try to update table A	Table A is blocked in T1	

The only solution is to decide which of the 2 transactions to win, by using the DEADLOCK_PRIORITY, that can be set (LOW, NORMAL, HIGH, or from -10 (-5) to 10 (5)). Implicit is NORMAL (0).

The victim transaction is chosen like this:

1. The transaction with the lowest DEADLOCK_PRIORITY
2. If both of the transactions have the same DEADLOCK_PRIORITY, the victim is the one, less expensive at ROLLBACK
3. If both of the transactions have the same DEADLOCK_PRIORITY and the same cost, the victim is chosen randomly

We consider tables: Books and Authors

```
-- deadlock
insert into Books values ('Book for Deadlock', 'english')
insert into Authors values ('Author for Deadlock')
select * from Books
select * from Authors
```

Results		
id	title	language
1	Pride and Prejudice	english
2	Harry Potter and the Chamber of Secrets	english
3	Panda	english
4	Peter Pan	English
5	At the end of the world	English
6	Book for Deadlock	english

id	name
1	Author for Deadlock

Deadlock example:

<pre>-- transaction 1 begin tran update Books set title='deadlock Books Transaction 1' where id=8 -- this transaction has exclusively lock on table Books waitfor delay '00:00:10' update Authors set name='deadlock Authors Transaction 1' where id=1 commit tran</pre>	(1 row(s) affected)
<pre>-- transaction 2 begin tran</pre>	(1 row(s) affected)

```

update Authors set name='deadlock Authors Transaction 2' where id=1
-- this transaction has exclusively lock on table Authors
waitfor delay '00:00:10'
update Books set title='deadlock Books Transaction 2' where id=8
commit tran

```

Messages

```

(1 row(s) affected)
Msg 1205, Level 13, State 51, Line 7
Transaction (Process ID 56) was deadlocked on lock resources with another process and has been chosen as the deadlock victim. Rerun the transaction.

```

```

select * from Books
select * from Authors

```

So, transaction 1 is the winner and update the tables Books and Authors, and transaction 2 is the victim

Results			
	id	title	language
1	1	Pride and Prejudice	english
2	2	Harry Potter and the Chamber of Secrets	english
3	3	Panda	english
4	5	Peter Pan	English
5	7	At the end of the world	English
6	8	deadlock Books Transaction 1	english

	id	name
1	1	deadlock Authors Transaction 1

If in transaction 2, we set DEADLOCK_PRIORITY to HIGH, or, if in transaction 1 we set DEADLOCK_PRIORITY to LOW, the winner will be transaction 2 and the victim transaction 1.

```

-- transaction 1
begin tran
update Books set title='deadlock Books Transaction 1' where id=8
-- this transaction has exclusively lock on table Books
waitfor delay '00:00:10'
update Authors set name='deadlock Authors Transaction 1' where id=1
commit tran

```

(1 row(s) affected)
 Msg 1205, Level 13, State 51, Line 7
 Transaction (Process ID 54) was deadlocked on lock resources with another process and has been chosen as the deadlock victim. Rerun the transaction.

```

-- transaction 2
SET DEADLOCK_PRIORITY HIGH
-- SET DEADLOCK_PRIORITY LOW
begin tran
update Authors set name='deadlock Authors Transaction 2' where id=1
-- this transaction has exclusively lock on table Authors
waitfor delay '00:00:10'
update Books set title='deadlock Books Transaction 2' where id=8
commit tran

```

(1 row(s) affected)
 (1 row(s) affected)

```

select * from Books
select * from Authors

```

100 %

Results			
	id	title	language
1	1	Pride and Prejudice	english
2	2	Harry Potter and the Chamber of Secrets	english
3	3	Panda	english
4	5	Peter Pan	English
5	7	At the end of the world	English
6	8	deadlock Books Transaction 2	english

	id	name
1	1	deadlock Authors Transaction 2

Another possible solution for Deadlock is to execute the statements in the same order in both of the transactions. As result, first are performed the UPDATE's from the first transaction executed and then the UPDATE's from the second transaction executed.

```
-- transaction 1
begin tran
update Books set title='deadlock Books Transaction 1' where id=1047
-- this transaction has exclusively lock on table Books
waitfor delay '00:00:10'

update Authors set name='deadlock Authors Transaction 1' where id=1006
commit tran

-- (1 row affected)
-- (1 row affected)
```

```
-- transaction 2
begin tran
update Books set title='deadlock Books Transaction 2' where id=1047
-- this transaction has exclusively lock on table Books
waitfor delay '00:00:10'

update Authors set name='deadlock Authors Transaction 2' where id=1006
commit tran

-- (1 row affected)
-- (1 row affected)
```

```
select * from Books
select * from Authors
```

```
-- deadlock
insert into Books values ('Book for Deadlock', 'english')
insert into Authors values ('Author for Deadlock')
select * from Books
select * from Authors

delete from Books where title like 'deadlock%'
delete from Authors where name like 'deadlock%'
```

150 %

Results		Messages	
id	title	language	
1	Pride and Prejudice	english	
2	Harry Potter and the Chamber of Secrets	english	
3	Panda	english	
4	Peter Pan	English	
5	At the end of the world	English	
6	deadlock Books Transaction 2	english	

id	name
1	1006 deadlock Authors Transaction 2