**44-560 Advanced Topics in Database Systems**

**Assignment-04: Transaction Management**

Are you excited to know what happens when two concurrent transactions are being executed on the database?

Please follow the below steps to know by yourself.

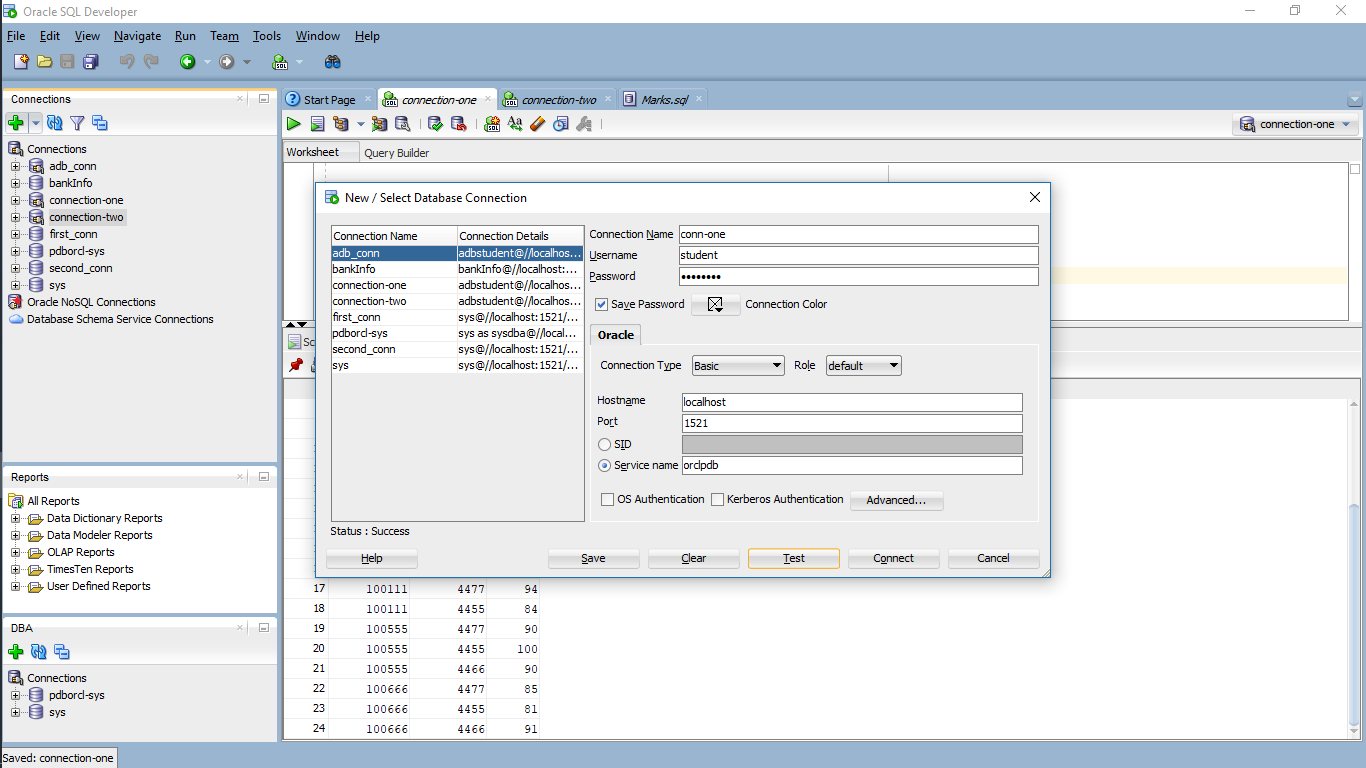
1. Create a user and grant Connect, resource and DBA permissions to the user in SQL plus. Refer to the SQL help worksheet provided earlier.

**For creating a user:** CREATE USER <username> IDENTIFIED BY <Password>;

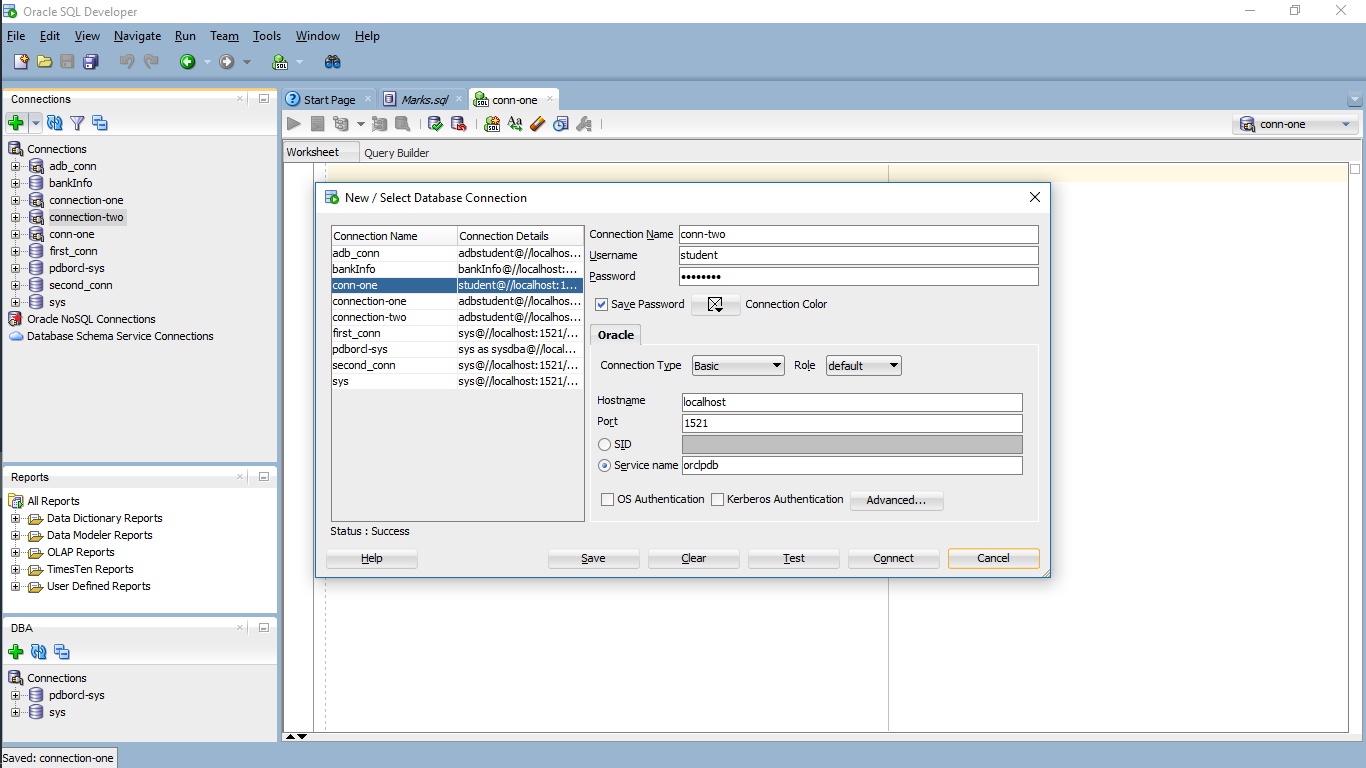
**For granting permissions:** GRANT CONNECT, RESOURCE, DBA TO <username>;

1. Create two different connections in SQL developer with the same user and same service name i.e., orclpdb or pdborcl (whatever you have given while installing the Oracle).

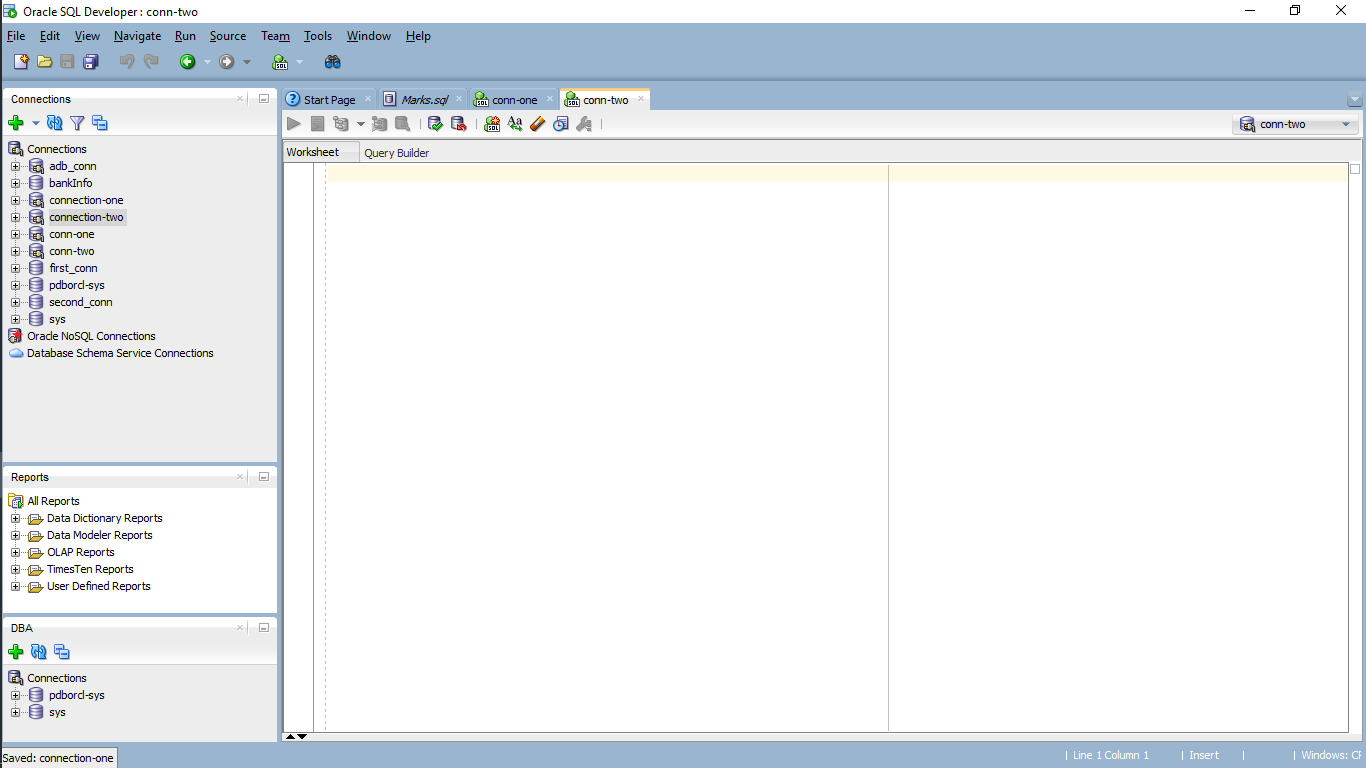
First Connection name: connection-one



Second Connection name: connection-two



1. Open the SQL worksheets of both the connections by right clicking the connections.

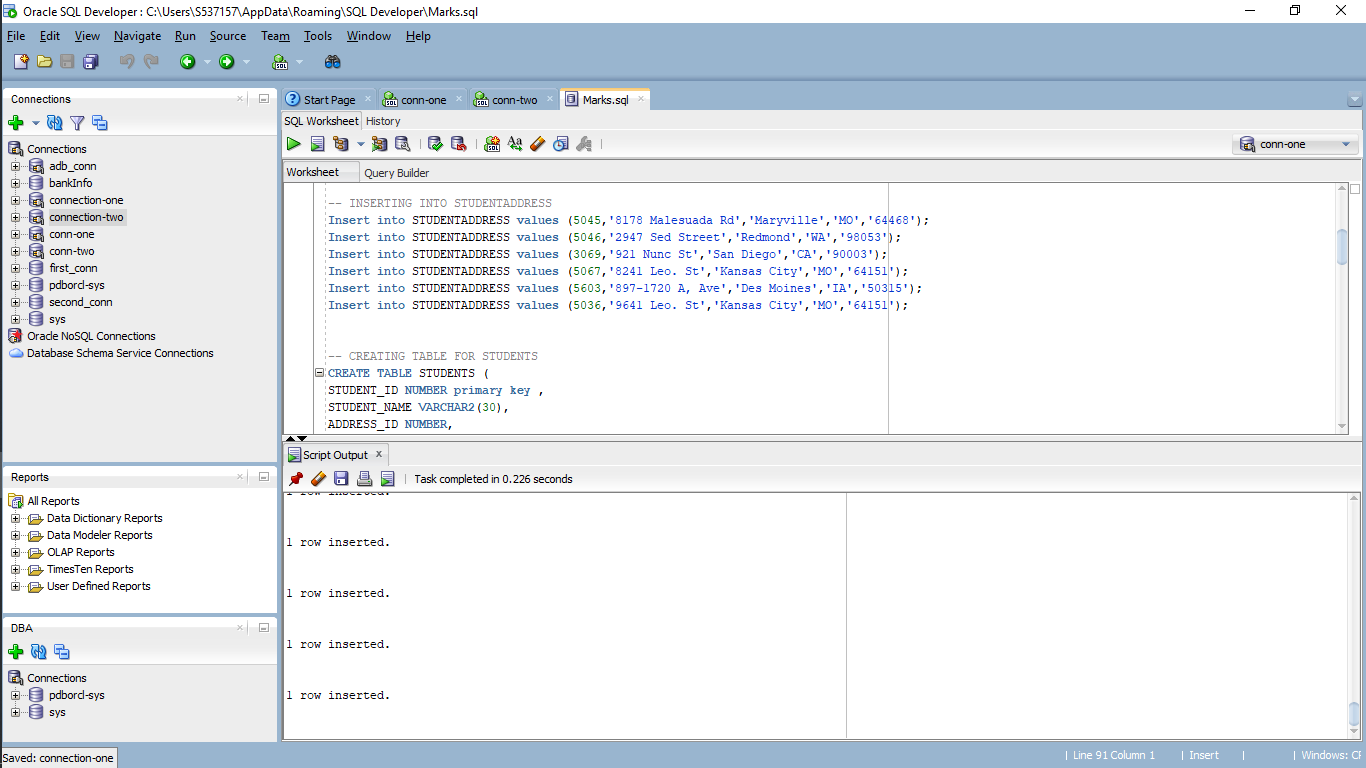


1. Open the Marks.sql file provided below and run the file line by line. Select any of the two connections that we created while running the script file.

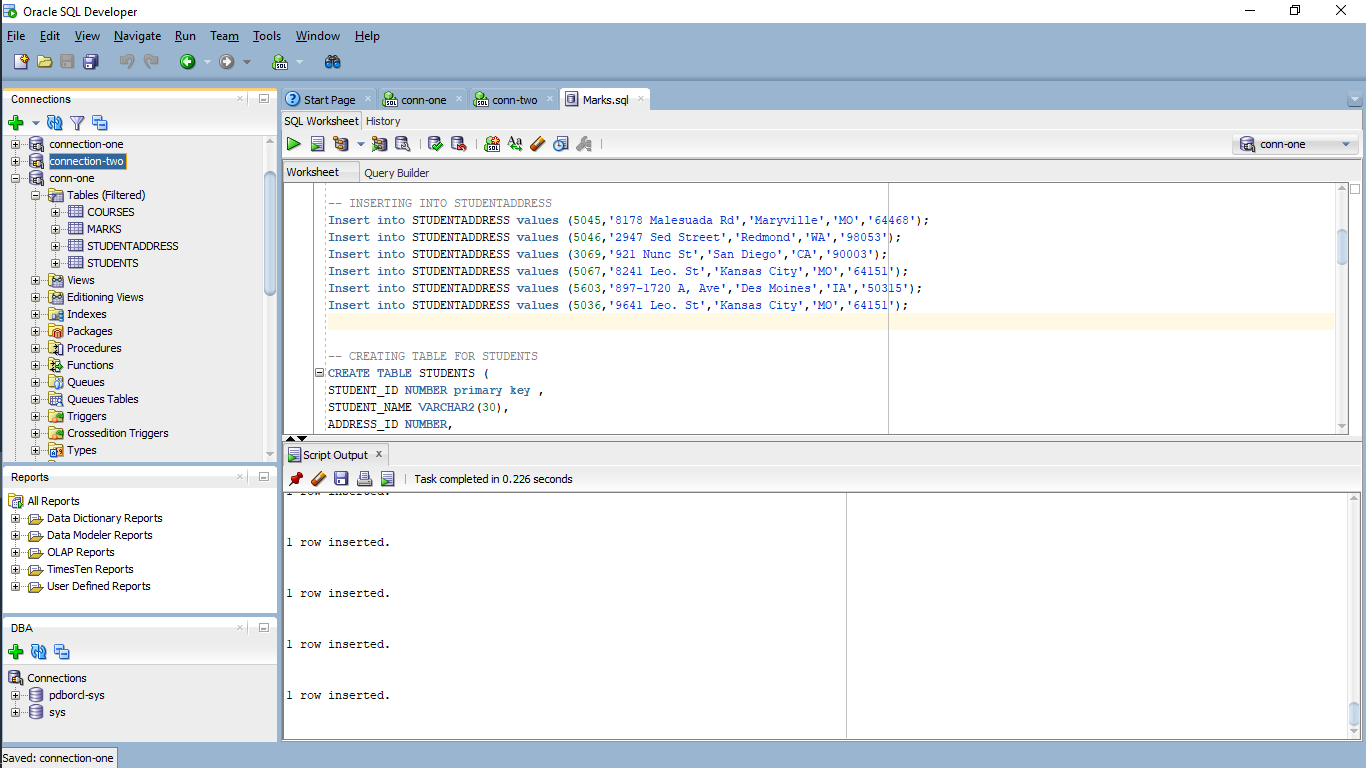
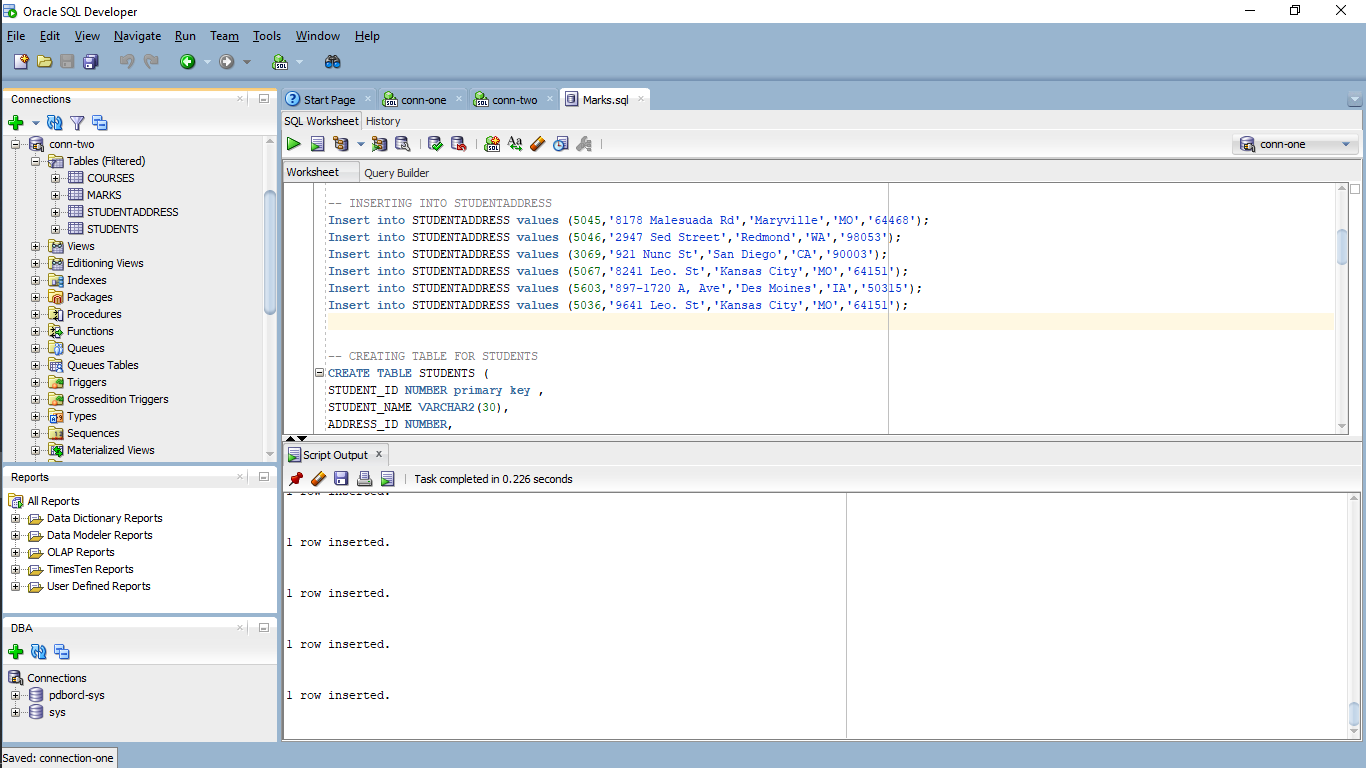
**SQL file:**



This process results as the following screen.



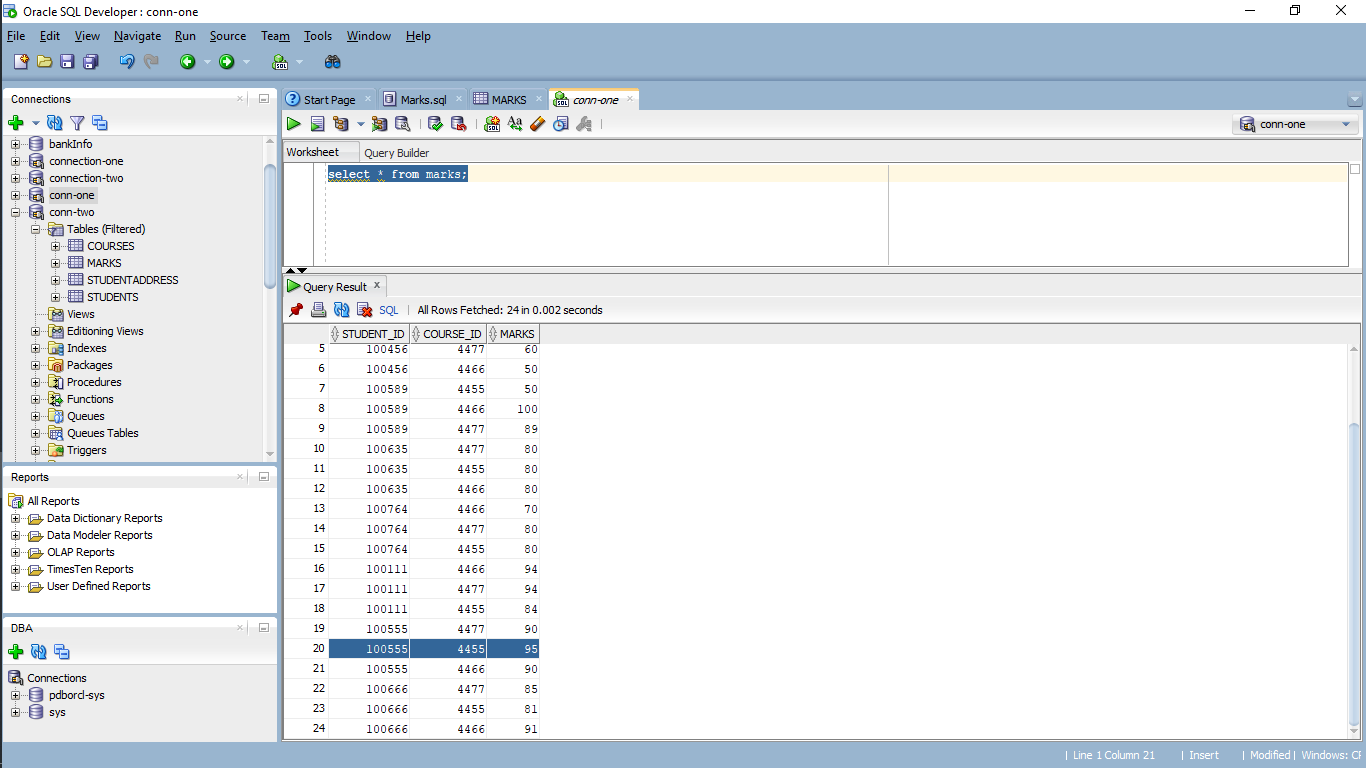
1. After running the script, you should observe same tables created in both the connections.

1. Now let us test by executing concurrent transactions on the same database from the two connections that we have created. We will update the marks of student with STUDENT\_ID **‘100555‘** in subject with SUBJECT\_ID **‘4455’** in both the connections.

In the first connection run the below queries in the order specified:

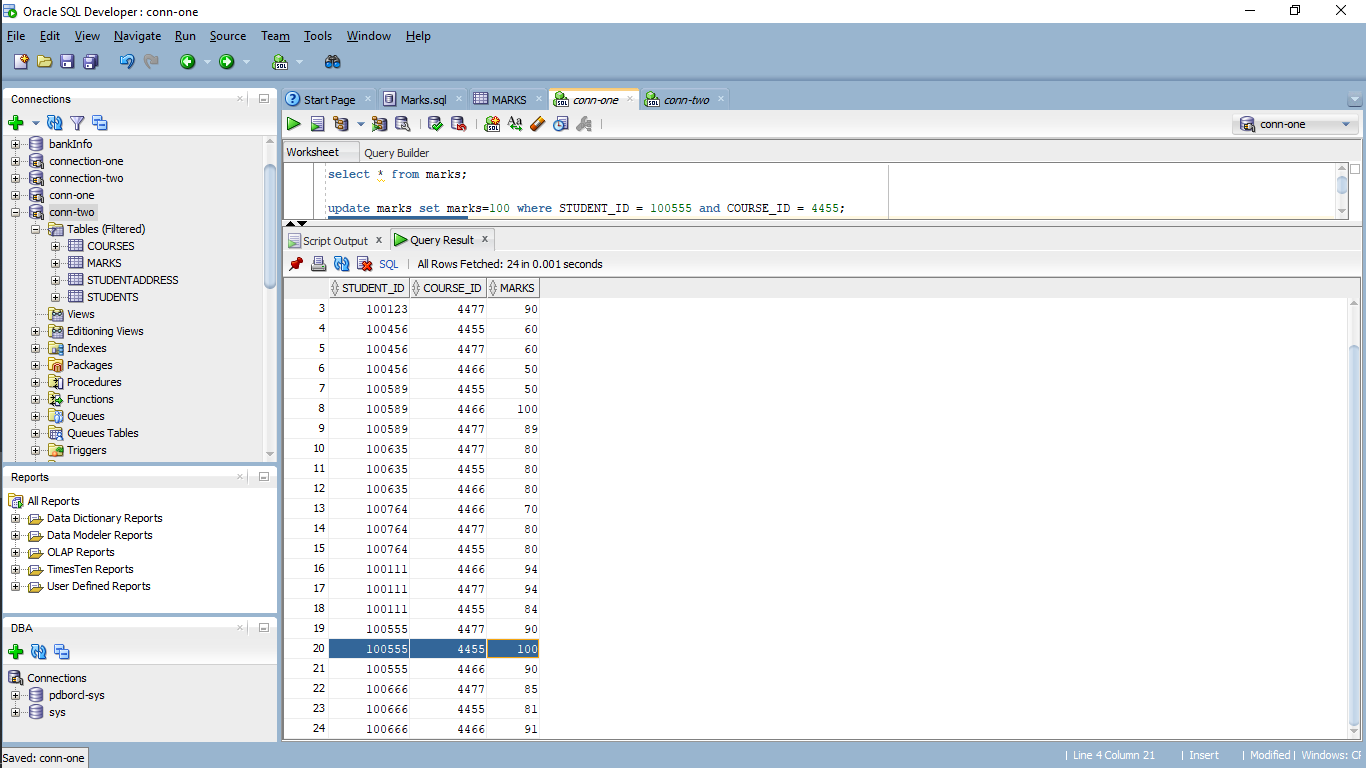
1. select \* from marks;



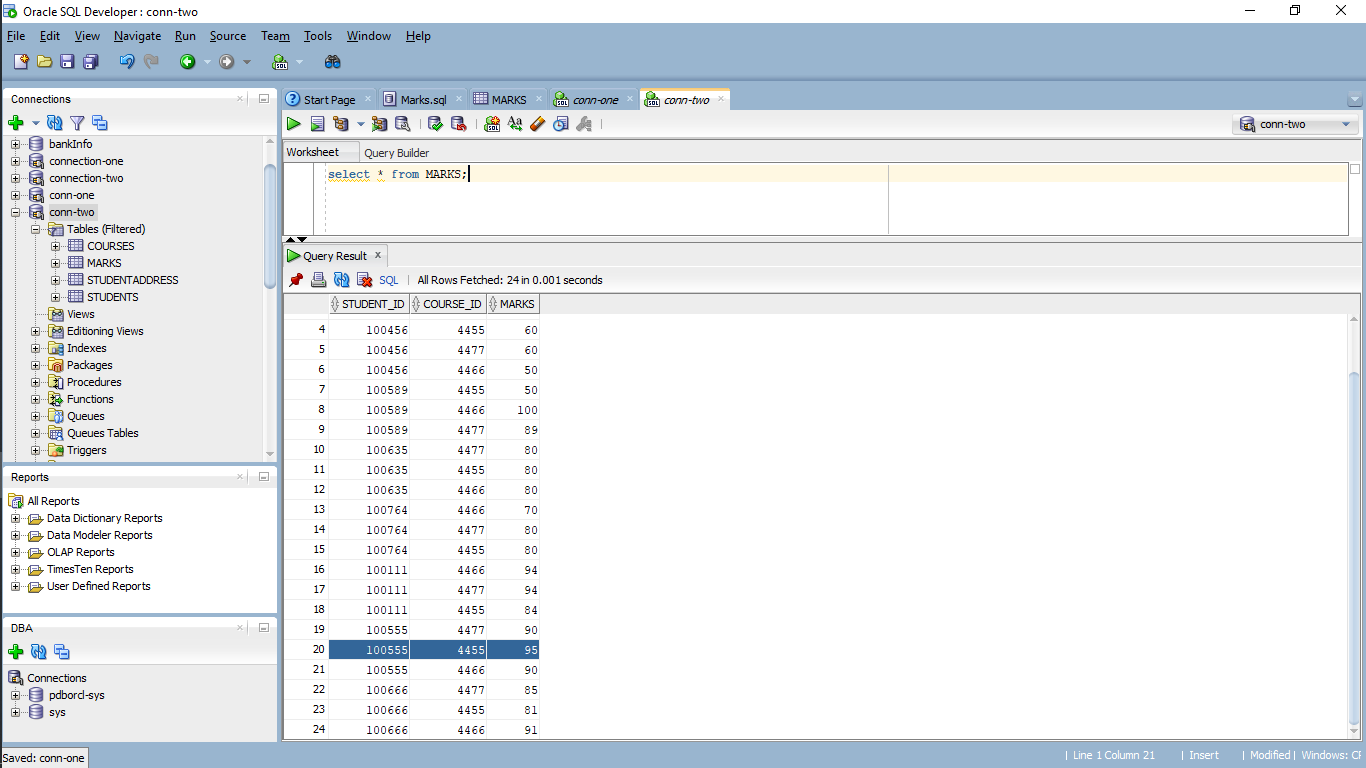
1. Now update the marks to 100 and check the description:

Use the below query to change it.

update marks set marks=100 where STUDENT\_ID = 100555 and COURSE\_ID = 4455; and check the data once again.



1. In the second connection run the below queries in the order specified:
2. Select \* from marks;



Here you can observe that the mark is still 95.

**Answer the below questions:**

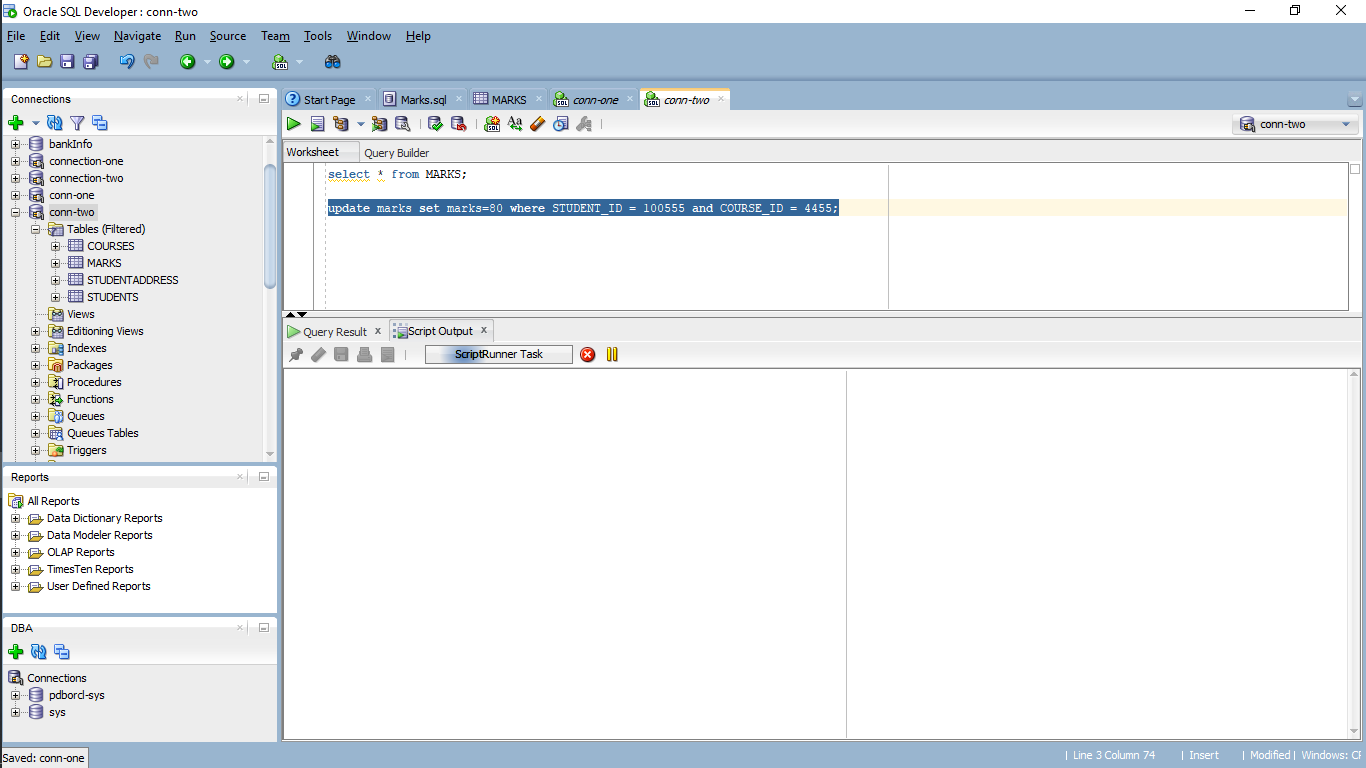
1. Why the marks is still 95 even though we have updated it in First connection?

A. We had updated data log of marks table for row 20 for connection 1 and in the mean while we did not fully complete the transaction for connection 1 and we are trying to access the updated value in connection 2 of same row where we are losing the updated value and we are seeing the data in that row is not updated value.

1. Update the marks to 80 in the second connect.

update marks set marks=80 where STUDENT\_ID = 100555 and COURSE\_ID = 4455;

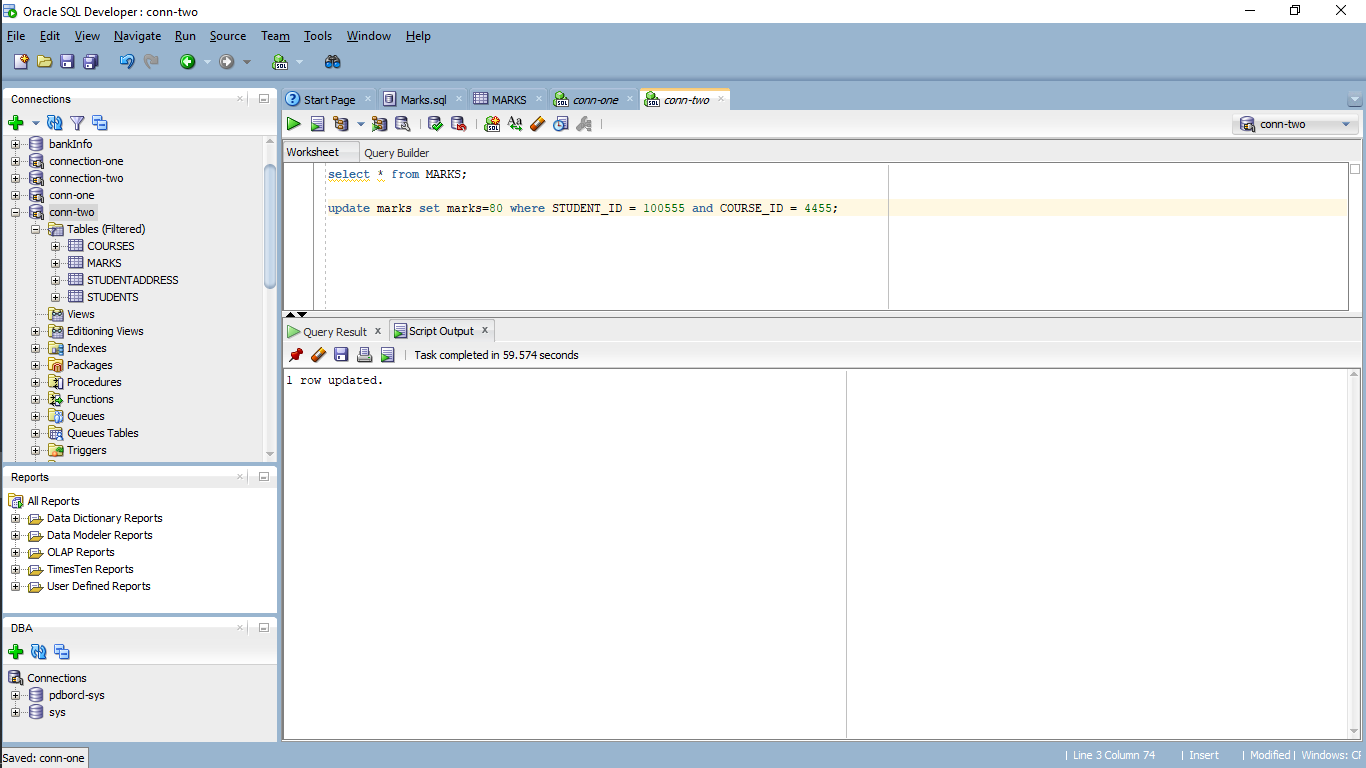
You can see that the task is still executing and spinning script output.



1. Commit the update we performed in first connection.

commit;

You can see that row is updated and the output in the second connection stopped spinning.



**Answer the below questions:**

1. Why was the output in second connection continuously spinning and not completing before?

A. The transaction in connection 1 is still not completed so the output is spinning and waiting for connection 1 to complete its transaction.

1. Why did the update complete in the second connection, once we committed in the first connection?

A. The transaction is completed in connection 1 as we committed the database to that point and shifted to connection 2 and performed update transaction which is carried in a serialized manner making the update complete.

What are the of marks for student with STUDENT\_ID = 100555 and COURSE\_ID = 4455 on each connection? Is the value the same or different in both connections? Why? (don’t commit in the second connection)

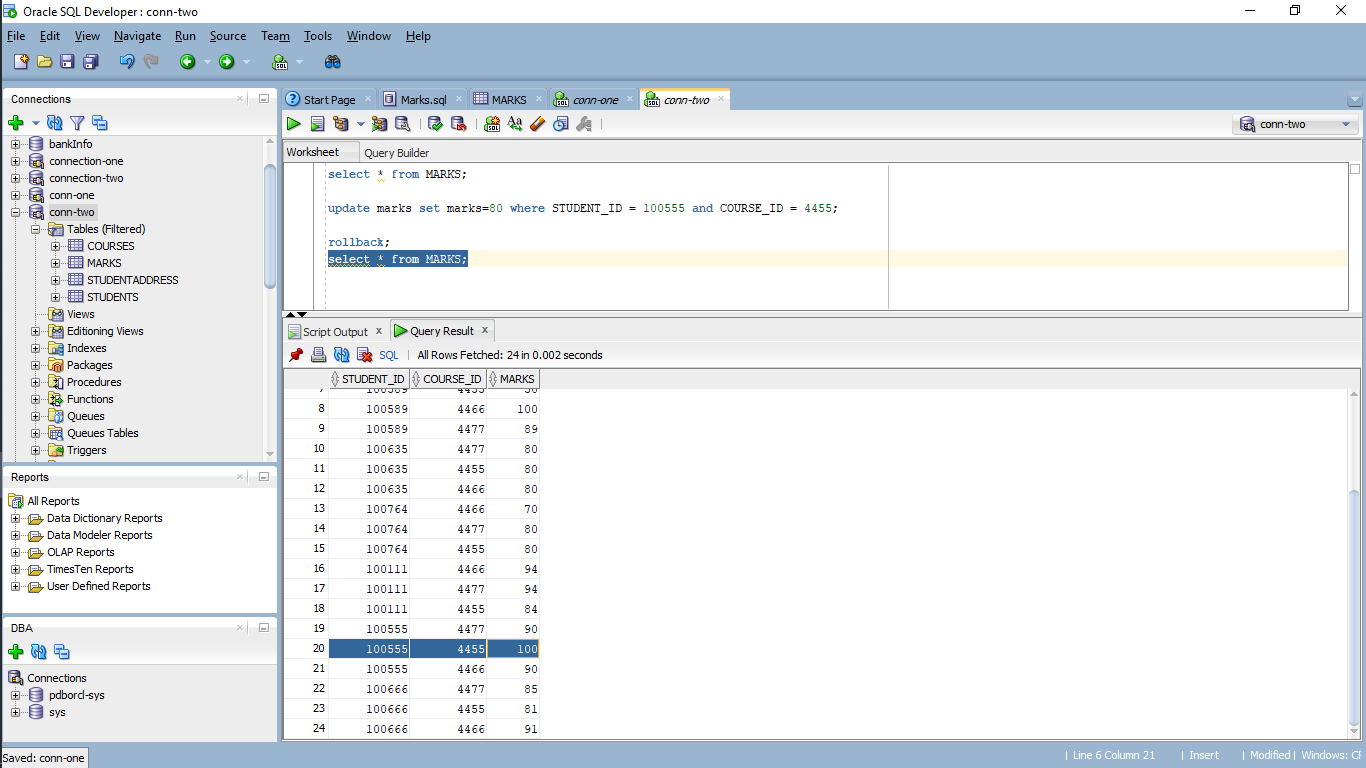
A. The marks are different for STUDENT\_ID = 100555 and COURSE\_ID = 4455 in both connections. In connection 1 the value is 100 and in connection 2 the value is 80 because we updated and committed the data in connection 1 so the value is 100 and in connection 2 we updated the value to 80 so the value is 80 and different.

1. Now, in the second connection, rollback the changes and check the marks value in that connection.

rollback;

select \* from marks;

You can see that the description value as “100”



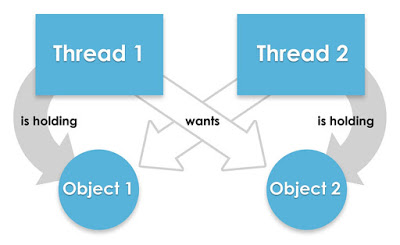
**Answer the below question:**

1. What happened immediately after we rollback in second connection?

A. We performed rollback command in connection 2 where we are getting the updated value of 100 which we updated in connection 1 because we committed the database to that point in connection 1 and when we performed rollback we are getting the committed data to that point (i.e value of 100).

**Questions about Transaction Management:**

1. In terms of a database, what is the picture below showing?



A. Deadlock because thread 1 want to access object 2 and thread 2 wants to access objects 1 as both are in waiting state to complete their respective transactions so deadlock occurs.

1. Consider the following two transactions made within a bank database:
   1. **T1:** The bank is preparing a report and runs a query that shows the sum of the balances of all its accounts.
   2. **T2:** A balance transfer of $100 is executed from account X to account Y in the following manner:

|  |  |  |
| --- | --- | --- |
| 1 | Read the balance of X | X: 400 |
| 2 | Compute X-100 |  |
| 3 | Write new balance of X | X: 300 |
| 4 | Read the balance of Y | Y: 300 |
| 5 | Compute Y+100 |  |
| 6 | Write new balance of Y | Y: 400 |

Suppose transaction T1 occurs after Line 3 of T2, but before Line 6 of T2. In that case, the sum computed in T1 will not have included the correct balance for account Y. What type of concurrency control problem is this?

A. Inconsistent Retrievals as T1 is not reading the updated value of T2 after the balance is transferred from X to Y.