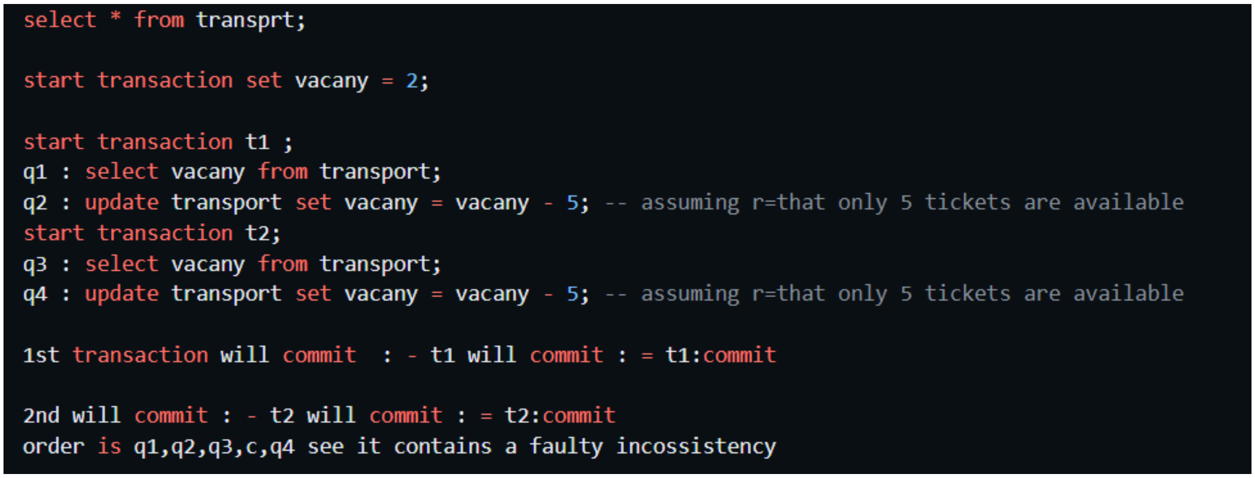
Project Deadline- 6 Transactions

Conflicting Transactions:

Transaction 1:

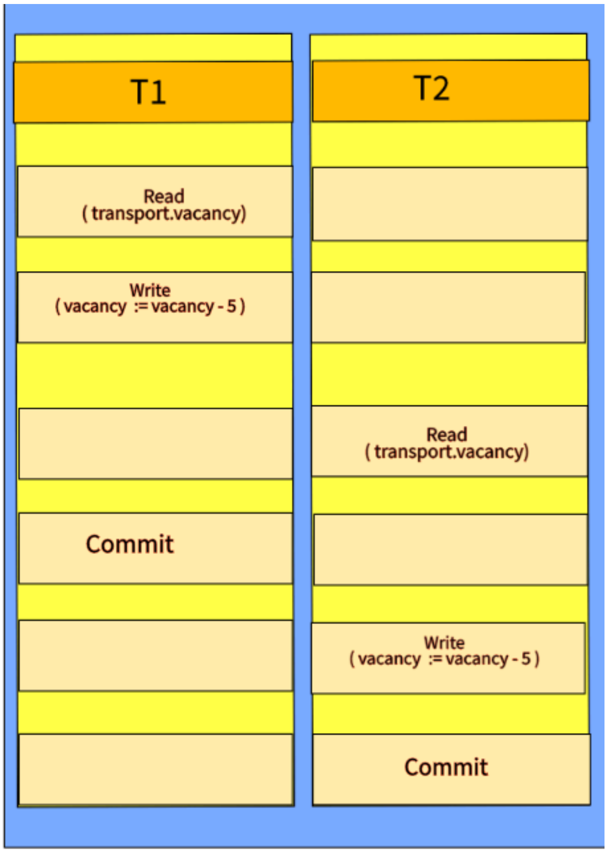


```
select * from transprt;

start transaction set vacany = 2;

start transaction t1 ;
q1 : select vacany from transport;
q2 : update transport set vacany = vacany - 5; -- assuming r=that only 5 tickets are available
start transaction t2;
q3 : select vacany from transport;
q4 : update transport set vacany = vacany - 5; -- assuming r=that only 5 tickets are available

1st transaction will commit  : - t1 will commit : = t1:commit

2nd will commit : - t2 will commit : = t2:commit
order is q1,q2,q3,c,q4 see it contains a faulty incossistency
```

Explanation:

Two transactions try to update the same vacancy count (limited tickets). Conflict arises because both read the initial vacancy before any update is confirmed.

For Example : T1 checks how many seats are left (let's say 5). Then T2 jumps in and also checks (still 5). T1 successfully buys a ticket (5 - 1 = 4). But T2 also tries to buy one (thinking there are still 5), leading to negative seats (-1).



T1	T2
Read (transport.vacancy)	
Write (vacancy := vacancy - 5)	
	Read (transport.vacancy)
Commit	
	Write (vacancy := vacancy - 5)
	Commit

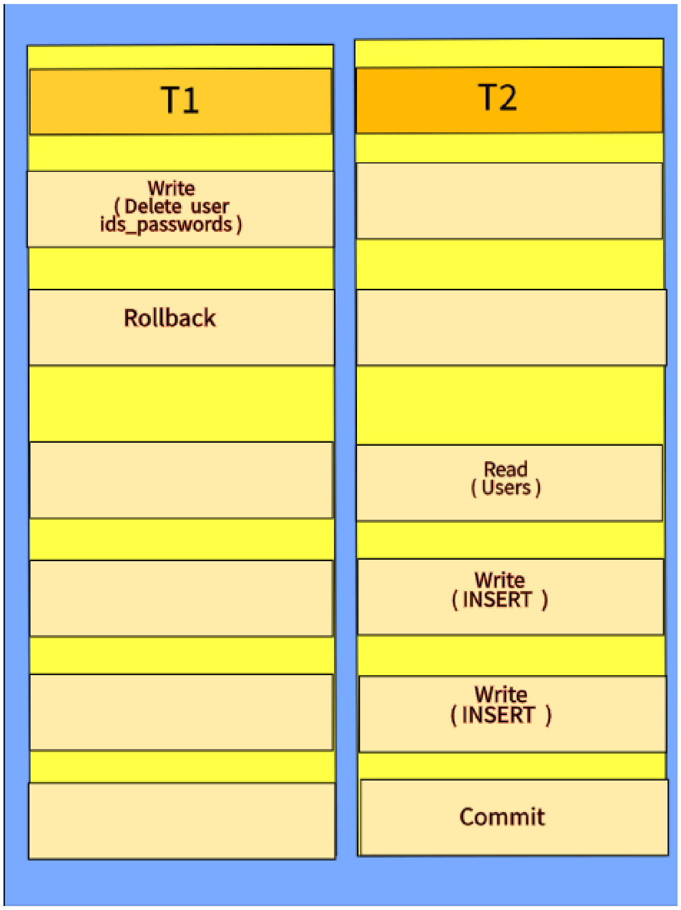
Transaction 2:

```
start transaction t3 ;
q1: delete from userids_passwords;
q2: Rollback

start transaction t4 ;
q3: select * from users;
q4: INSERT Into userids_passwords(userid,password,is_locked) VALUES
(8895,'Aditya@1998','F');
q5: INSERT INTO Users (userid, email, name, phnumber, gender, Address_hno, City, Pincode, dob) VALUES
(8895, 'zgonzalez@example.com', 'Connie Williams', '5152121722', 'F', '0569', 'New Karenview', '489077', '2005-11-26');
q6:commit

-- The 2nd transaction will be in inconsistent state as the userids_passwords table will be empty and the users table will
-- have a record with userid 8895 which is not present in the userids_passwords table. This violates the referential integrity
-- constraint between the two tables.

-- Also Lets say Vikraam (check spelling ) ji was there was wants to delee all transactions (by mistake you know he is a good
-- person then Mukesh ji came he found out userid 8895 is available as there will be no recrds so mukkesh ji tried to insert the
-- record but dbms has locked) so when vikraam ji did rollback mukesh ji got an error userid exist


order of execution q1,q3,q4,q5,q2,q6
Soon it will fail as it will give an error
```

Explanation:

The 2nd transaction will be in inconsistent state as the user ids_passwords table will be empty and the users table will have a record with user id 8895 which is not present in the user ids_passwords table. This violates the referential integrity constraint between the two tables.

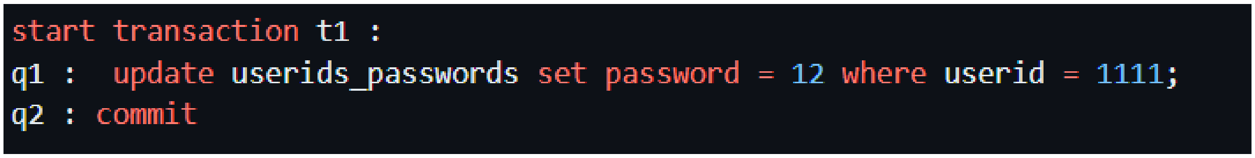
For example :
Lets say Vikram ji was there was wants to delete all transactions (by mistake you know he is a good person then Mukesh ji came he found out user id 8895 is available as there will be no records so mukesh ji tried to insert the record but DBMS has locked) so when vikram ji did rollback Mukesh ji got an error user id exist



T1	T2
Write (Delete user ids_passwords)	
Rollback	
	Read (Users)
	Write (INSERT)
	Write (INSERT)
	Commit

Non-Conflicting Transactions:

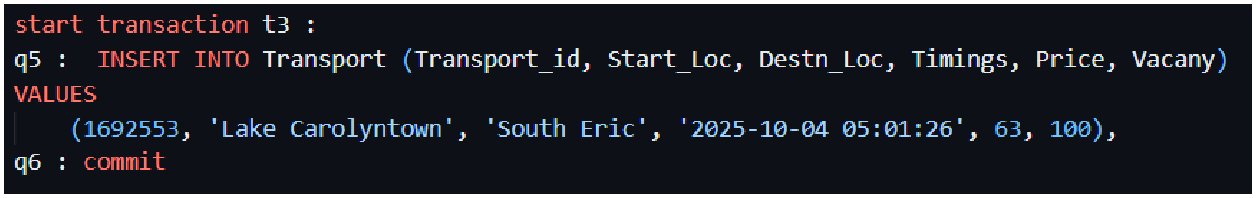
Transaction 1:



```
start transaction t1 :
q1 :  update userids_passwords set password = 12 where userid = 1111;
q2 : commit
```

Explanation: This transaction updates the login password for the person with userid = 1111 to a new value of 12.

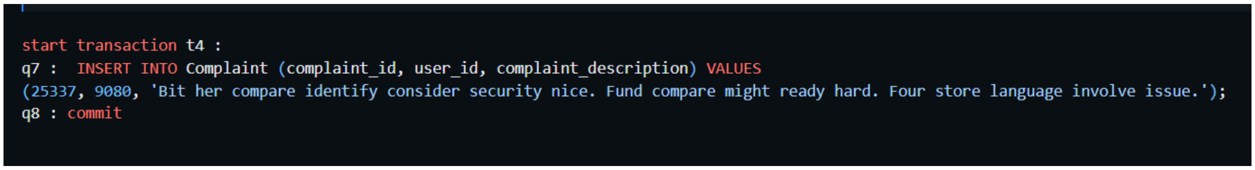
Transaction 2:



```
start transaction t3 :
q5 :  INSERT INTO Transport (Transport_id, Start_Loc, Destn_Loc, Timings, Price, Vacany)
VALUES
    (1692553, 'Lake Carolyntown', 'South Eric', '2025-10-04 05:01:26', 63, 100),
q6 : commit
```

Explanation: The above transaction inserts a new transport which is available on the website that adds the given transport_id, start and end location, the timings and the number of vacancies with price for the tickets into the transport table.

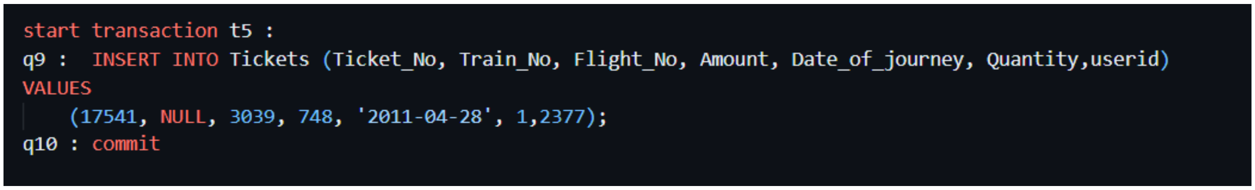
Transaction 3:



```
start transaction t4 :
q7 :  INSERT INTO Complaint (complaint_id, user_id, complaint_description) VALUES
(25337, 9080, 'Bit her compare identify consider security nice. Fund compare might ready hard. Four store language involve issue.');
q8 : commit
```

Explanation: The above transaction adds a complaint of a user with user_id 9080 to the compaint's table.

Transaction 4:



```
start transaction t5 :
q9 :  INSERT INTO Tickets (Ticket_No, Train_No, Flight_No, Amount, Date_of_journey, Quantity,userid)
VALUES
    (17541, NULL, 3039, 748, '2011-04-28', 1,2377);
q10 : commit
```

Explanation: This adds a entry into the tickets table for a ticket being booked for the user with user_id 2377.