

S2.7: TRANSACTIONS & CONCURRENCY

TASK 1: The First Hour

1. In your own words, what is meant by each of the following database terms:
 - a. Transactions
 - b. Concurrency Control
 - c. Locking

2. Now have a go at these exam style questions:
 - a. A database transaction should pass the ACID test. Discuss what is meant by the acronym ACID in relation to database transactions.
 - b. Discuss problems that can be caused if concurrency control is not implemented correctly to a database, and what mechanisms can be used to prevent those problems.
 - c. Brenda Biggs has a bank account with an initial balance of £2000. By coincidence, the account is being updated simultaneously by 2 database transactions; Transaction A is a monthly transfer of funds from Brenda's current account to her savings account and Transaction B is Brenda at the bank, depositing a £600 cheque into her current account.

Transaction A reads the initial balance of £2000 into the memory buffer, and deducts the £300 from the account, leaving a balance of £1700.

Transaction B reads the balance of £1700 and adds the deposit of £600, giving a new balance of £2300.

Transaction A fails before completion and executes a rollback, returning the balance to £2000.

Transaction B now writes its in-memory balance value of £2300 to the account balance within the database and commits the transaction.

Explain why the database has become corrupted and describe how both transactions can be carried out whilst maintaining the integrity of the database.

TASK 2: The Second Hour

This week you will start a 3-week mini project to bring together all the things you have learnt over this semester. You should treat this as an item of work that could be added to a personal portfolio to show to a prospective employer. Over the three weeks you will design and build a database of your own choice from scratch. You can choose any real-world scenario you like, but it should have a certain level of complexity (minimum of 4 tables). If you are struggling to come up with anything here are some ideas:

- A music streaming application
- A video sharing application
- An online marketplace application
- A railway reservation system
- A hospital management system
- An online job portal
- A hotel booking system
- A second-hand vinyl record store
- An online food delivery service
- A taxi service
- A dating application

1. Once you have chosen your idea you should create an initial un-normalised list of attributes. Remember to include everything you will need to store information about in your database.
2. The next step is to normalise your list of attributes following the stages from 1NF to 3NF, creating ERD's, as you have done in the labs for the past few weeks.
3. When you have a normalised design, create a data dictionary for each of the tables. Making sure to specify each attributes data type and size, and any sensible validation that could be used. You can use the table that was included in the Data Integrity lab.
4. Create the table scripts specifying some of your validation to each table. Specifying the primary and foreign keys is the minimum validation, in addition to this you should apply other types of validation for each of your tables.
5. Run each of your scripts in Oracle Apex to create the database. Remember that the order you create the tables is important. You can't create a table that references another table before the other table has been created.
6. Now add some sample data to each table. The more you add the better. You can write the SQL INSERT scripts to do this. However, the easiest way is via the form entry in Apex. Again, remember the order that data is entered is important.
7. Finally, write a few queries to demonstrate your database. The more complex these queries are, the better they will demonstrate your SQL expertise!