The Database Processing and Management Class was the second database class we took after the Database Management Systems. In this class we learnt about database models, how to control concurrency, transactions, security of the database system, recovery and so on. The structure of the relationships in most databases follow either the hierarchical, network or relational models. A hierarchical database model is one in which the data are organized in a top-down or inverted tree-like structure. This type is best suited for situations where the logical relationships between the data can be properly represented with the one-parent-many-children approach. A network model is an extension of the hierarchical database model where there is an owner-member relationship in which a member may have many owners, in contrast to a one-to-many-relationship. A relational model describes data using a standard tabular format where all data elements are placed in two-dimensional tables called relations, which are the equivalent of files.

While building and working with databases, we have to ensure that one user's activity does not impact the others. Since databases can be accessed by more than one user at a time, it is very important for us to use concurrency controls. To avoid concurrency issues, we can use resource locking that disallows transactions from reading, modifying and writing to a data which is locked. We can choose different locking mechanisms (implicit locking, explicit locking, shared locking, etc.) to lock our database. However, we are vulnerable to face a deadlock issue where two transactions or activities end up waiting on each other to release their resources. Transactions consists of series of stems and it is important for each step to be successful in order for that transaction to be saved. There are different types of transactions. An atomic transaction is where all of the database actions occur or none of them do. This helps to ensure that the transaction completes everything it needs to do before saving the changes. The next type is a consistent transaction where no other transaction is permitted until the current is finished. This is helpful as it ensures that the transaction integrity has statement level consistency among all the records. The next type is a durable transaction where all committed changes are permanent.

The other important thing we learned in this class is database security. We have to make sure that only authorized users can have access to a specified database. Authorization assigns the authenticated users specific right to do specific activities on the system and makes sure that the user has the basic right to use the system in the first place. Another important concept is data recovery. There are two ways to recover data, one via reprocessing where the system goes back to the known point and reprocesses the workload from there, and the other through the rollback/rollforward which is done by periodically making copy of the database and keeping the log of the changes made by transaction against the database since save. When failure occurs one of the two method can be used. Rollforward method restores the saved data and all valid transaction since the save are reapplied. Rollbackward method corrects mistakes by undoing the changes they made in the database, then the valid transactions that were in the process at the time of failure are restarted.

The above artifact is of an assignment where we used triggers. A trigger is a stored procedure that is automatically invoked by the DBMS when a specific activity takes place. Triggers can be defined over DML or DDL statement. DDL trigger can record every instance of a Create, Alter or Drop statement that is executed. DML trigger can be used to enforce referential integrity but it is not preferred. DML triggers are used to enforce business rules and to monitor modification to database table. In SQL, triggers can have definition to include multiple event types such as Before, Insert and Delete. I also chose this specific assignment because we use cursors which is a set of rows together with a pointer that identifies a current row.