# Introduction to SQL

* Structured Query Language
* It is **NOT** a programming language. It is a **declarative query** language.
  + You tell the program what needs to be done instead of delivering the implementation details to do what needs to be done.
* It performs operations against a relational database.

## Database

* It is just an organize collection of data stored in some organized format.
* They allow us to input, manage, organize, and retrieve data quickly.
* Traditionally, it is organized into “tables” and each table will have a row and a column.
  + Rows will be the same as “records”
  + Columns will be the same as “fields”
* Data is the intersection between a row and a column

## RDBMS

* It stands for Relational Database Management System.
  + SQL is a specific style of RDBMS
* It upholds specified relationships between tables or our data.
* It includes functions that maintain the security, accuracy, integrity, and consistency of the data.

# SQL Sublanguages

## DDL

* Data Definition Language
* It is for the creation/alteration of table structures.
* Create – Most commonly used to create tables and their columns.
  + Can also be used to create other things.
* Alter – Will change the column of the table.
* Truncate – Removes all the data in a table.
  + You cannot roll back the changes.
* Drop – drops the table.

## DML

* Data Manipulation Language
* It is for changing/manipulating/modifying the data within a table.
* Insert – Add row(s) to your table
* Select – Retrieves the data from a table for us to read
* Update – Modify the rows in the table
* Delete – Delete the rows in the table

## DQL

* Data Query Language
* It is for reading the data from a table.
* It is bit controversial in that some people do not believe this is part of SQL sublanguage but for our case it will be.
* All it has is a bunch of select.

## TCL

* Transaction Control Language
* They are used to manage transactions.
  + Think of as methods in C#.
  + They are a logical work unit that will perform single or multiple statements in a database.
  + They help prevent data inconsistency because they will either execute all the statements inside the transaction or it won’t persists any of the statements at all.
  + Ex: You have a checking account, and you want to transfer some money to your savings account.
    - During the transaction of updating your checking account to have less money your database failed.
    - If you didn’t use a transaction, then it seems like the transfer took your money from your checking account and your savings account didn’t have that $100.
* Begin – It will start our transaction.
* Commit – To permanently save any transaction into the database.
* Savepoint – It is used to temporarily save a transaction.
* Rollback – Restores the database to the last committed state or save point.