Noah Dunn

14.1

**SELECT** TenantName **FROM**

(**SELECT** **DISTINCT** t.TenantName, t.TenantID **COUNT**(t.TenantID)

**FROM** Apartments **a** **JOIN** AptTenants **at**

**ON** **a**.AptID = **at**.AptID

**JOIN** Tenants t **ON** **at**.TenantID = t.TenantID

**WHERE** **COUNT**(t.TenantID) > 1

**GROUP** **BY** t.TenantName, t.TenantID

)

14.3

**UPDATE** **r**.Status **SET** **r**.Status = "Closed"

**FROM** Requests **r**

**JOIN** Apartments **a** **ON** **r**.AptID = **a**.AptID

**JOIN** Buildings b **ON** **a**.BuildingID = b.BuildingID

**WHERE** b.BuildingName = "Building #11"

14.4

There are even more than the following, but I will address the four most common:

**Inner Join OR Join**: The most common type of join. These are used when we need to link tables based upon column values that the two or more tables have in common.

**Left Join:** This join provides all elements from the table to the left of the join, combined with all matches available from the table to the right of the join. This type of join is particularly useful if we are searching for elements that do not match to the left side, as these columns are still presented in the resulting query, but they are filled with “NULL” values.

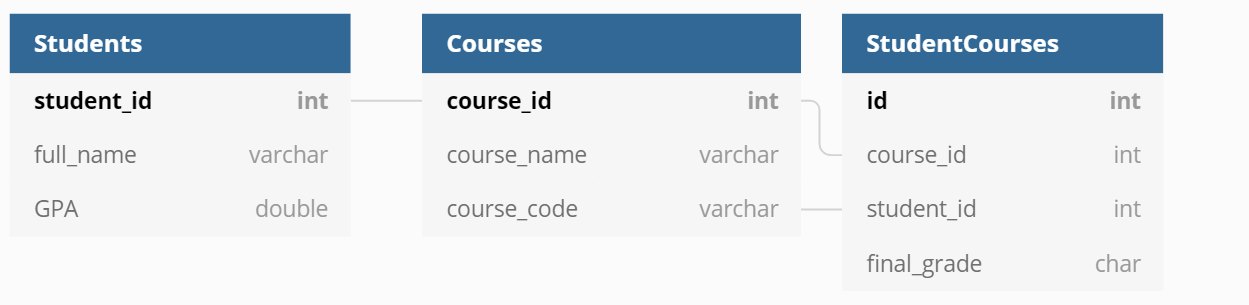
**Right Join:** This is identical to a left join, aside from the fact that we keep every row from the table to the right of the join. If we need all the information from the right table and want to check information available from the left table, this is the most useful join.

**Outer Join or Full Outer Join:** These are used sparingly and for very specific use cases due to their computational heavy nature. If we need **every possible combination** of columns rows from the table to the left of the join statement and those from the right, we can use this join.

14.5

**Denormalization -** The process of including redundant information in tables across a database to increase scalability in a system. Denormalized systems have simpler queries, faster updates, and faster deletions, but a well optimized, small normalized system gains ground on quicker read caching.

14.7



**SELECT** **TOP** 10 **PERCENT** **s**.full\_name

**FROM** STUDENTS **s**

**ORDER** **BY** **s**.GPA **DESC**