Databases: Homework #4 - Reading portion

1 Consider and briefly describe the differences between foreign key constraints and joins, and whether these two things are conceptually related at all.

Foreign key constraints require that, for a column of relation B, that this column only contains values in the primary key of relation A. A natural join, on the other hand, picks out the rows from tables A and B where the shared attributes of A and B are equivalent, and returns a relation as a result. I definitely see a relationship between the two in terms of maintaining integrity of data between two tables - while foreign keys are explicitly stated, I do see a natural join as a sort of formed foreign key of A to B, where the attributes of A and B are identical. I think a difference, though, is the nature of the "constraint" - foreign keys say "you can only have values that exist in this other table", while natural joins say "you can do whatever you want, but we'll see where these two tables line up and return the sum of their parts."

2 Given an example of one of your queries from HW3 as well as a rewritten example that is an equivalent example but uses join syntax as well.

```
SELECT    a.area_of_business
FROM      area_of_business_info a
JOIN      employee_info e
USING     (company_name)
WHERE     e.employee_name = "dave l"
```

3 Briefly describe what the purpose of the "order by" clause is, and give examples of how you could meaningfully use it in queries against your music schema. When comparing two versions of the same query, where one uses order by and the other doesn't, which do you think is more expensive to execute and why?

The order clause orders the relation resulting from a query in a specified manner, by a specified attribute or calculation. This could mean alphabetizing, ordering numerically, or other means of ordering rows in the resulting relation. I can see this being useful when trying to parse through large chunks of data in nearly all tables in our Company schema. When it comes to individual names, company names, or monetary values, knowing how things are ordered is powerful. Knowing the format of your output cuts out a ton of time when parsing through output, especially when doing so by hand. I would imagine that the query containing order by takes a bit longer to run than a query not containing order by, as ordering requires the use of a sorting algorithm that otherwise isn't run. Algorithms, of course, take time, and as such an order by clause makes a query take longer to return.