The conceptual data model! Interesting - this seems to be the culmination of what we have been trying to build throughout the first two units. The entity relationship diagram sounds like a good tool. Analogous to learning pseudocode diagrams in regular programming courses perhaps.

Interesting to see that \* means required - this is a fairly ubiquitous symbol for required. I laughed when I noticed a typo in the ER modeling pdf - the wrong ‘to’ was used instead of ‘two!

This topic seems very crucial, as before we have been looking and thinking about different entities in isolation (book, lender, etc), but now we are seeing how to effectively connect them all together.

The discussion question this week was a good example to learn about the infeasibility of many to many relations. Although the example wasn’t very realistic, as mentioned by the blog post, and also just knowing anecdotally, that I don’t typically think of one person owning more than one car, however I suppose it is possible. Can more than one person be registered to a car?

The solution in the discussion was to create another entity that essentially links the vehicle and the person. This entity would be unique and only exist in a one to one ratio with each of vehicle and person (one person cannot own the same car twice, one car paired with a single owner cannot belong to multiple licenses). To make this unique, we need the keys from each of the Driver and Vehicle entity. So the “Ownership” (KarenLopez 2009) entity would consist of attributes; Name, License Number, Vehicle ID Number, as an example.