* In our examples, we are connecting a C# application to a back-end database. What kind of database are we using? Could we connect to other back-end databases using C#?  Why would you want to do this?
  + Microsoft SQL Server
  + Yes, there’s MySQL, MSSQL, Azure SQL Database and several others
  + You may be connecting to a database that’s provided by a cloud provider, webpage provider and/or connecting to different companies databases.
* We used the ADO.NET and the Entity Data Model. Are there other ways to connect to this and other back-end databases? Why might you want to use other connectivity methods?
  + When using the Entity Framework, you can use LINQ to entities “for querying entity types that are defined in a conceptual model”. Entity SQL is also a tool provided by the Entity framework that uses the “dialect of SQL that works directly with entities in the conceptual model and that supports Entity Data Model concepts”. An alternative to the Entity Client provider is the SqlClient for Entity framework. Typically, this would be used when using a backend database like SQL server 2000, 2005, 0r 2008.
* In the “real world”, a back-end database might contain numerous simultaneous connections. Some connections could come from web-based users, and some from desktop users, and some from mobile users. Can you think of situations in which this would occur? Try to back up your responses by citing situations where this has been the case with you.
  + There are several “real world” examples where this could happen but the one that will pop into most people’s minds is social media. Countless people sign in through a social media’s website, on their phone, and some even have a desktop application for their product. This is a perfect example of what a “real world” back-end database with multiple different connection types would be like.