**Eager Loading**  
Eager Loading helps you to load all your needed entities at once; i.e., all your child entities will be loaded at single database call. This can be achieved, using the Include method, which returs the related entities as a part of the query and a large amount of data is loaded at once.

For example, you have a User table and a UserDetails table (related entity to User table), then you will write the code given below. Here, we are loading the user with the Id equal to userId along with the user details.

1. User usr = dbContext.Users.Include(a => a.UserDetails).FirstOrDefault(a => a.UserId == userId);

**Lazy Loading**  
It is the default behavior of an Entity Framework, where a child entity is loaded only when it is accessed for the first time. It simply delays the loading of the related data, until you ask for it.

For example, when we run the query given below, UserDetails table will not be loaded along with the User table.

1. User usr = dbContext.Users.FirstOrDefault(a => a.UserId == userId);

It will only be loaded when you explicitly call for it, as shown below.

1. UserDeatils ud = usr.UserDetails; // UserDetails are loaded here