

Hi.  
  
In this video we're going to identify attributes.  
  
So so far we've seen how to identify nouns and the entity types, we've seen how to identify the relationship between entities.  
  
But now we're going to see how to identify the attributes, which is by now just easy.  
  
In fact.  
  
But just for completeness we have to do this because these are the things we actually want to store about the entity types.  
  
So let's go back to the description and here we have again the example and we have again highlighted all the nouns as when we discussed the entity types.  
  
But now we are just looking for the attributes, not for the entity types.  
  
We start by looking at the buses.  
  
Again, number of buses.  
  
This is not an attribute because it would be an attribute of the bus company, because it says a bus company owns a number of buses.  
  
But as already in the previous videos, I'm going to tell you bus company is not an entity type in our system.  
  
So this describes an attribute of something that is not part of our database.  
  
So we forget about it.  
  
Each bus, this was an entity type, root, root.  
  
These are all entity types.  
  
Also, again, several buses.  
  
Each bus has a unique bus number.  
  
Now we are hitting definitely an attribute.  
  
So a bus should have the attribute bus number.  
  
It's important to store information about the seating capacity and the make type of all buses.  
  
Seating capacity, make type.  
  
These are clearly attributes of buses.  
  
Each route is distinguished by a route number.  
  
So a route number should be again an attribute.  
  
Not only should it be an attribute, but it says here is distinguished by.  
  
Similarly, previously we had each bus has a unique bus number.  
  
So these things are not only indicating that you are having an attribute, these are attributes that allow you to identify entities within an entity type.  
  
If I know the unique bus number, I know which bus I'm talking about.  
  
If I know the root number, I know which root I'm talking about.  
  
So these are what we called previously the identifiers of those entity types.  
  
And when we write down the entities and list the attributes, we normally underline the identifiers of them to say this is how we can figure out what bus we are talking about.  
  
And so on.  
  
Information is available on the average number of passengers carried per day for each route.  
  
This is some information that is available for each route.  
  
So it is a piece of information which is an attribute and it is of the entity type route.  
  
So it's another attribute of route.  
  
We have route number and average passengers.  
  
Now we come to the stages.  
  
Do we have anything about stages?  
  
Well, unfortunately this description is quite brief.  
  
So normally I Say that if you end up with an entity type which doesn't have any attribute, that something must be wrong.  
  
But in this particular case it's just the description is too high level.  
  
So if you end up, for example, you get the description of the data from your customer and you end up with an entity type, but then there's no attribute, then you just have to go back to them and say, okay, fair enough, you want to store information in this case about stages, but what is the information that you want to store?  
  
For the sake of the argument here, we just say each stage will have a name.  
  
And so we use stage name here as an attribute of stage.  
  
Then we have still two more entity types which we didn't discuss, Drivers and towns.  
  
So for the drivers we just look at the last sentence because there we have an employee number, a name, address, and sometimes a telephone number.  
  
So we have indeed employee number, name, address and telephone number as the attributes.  
  
Finally, the towns have a similar problem than the stages.  
  
Nothing is really known here about the towns, but we all know that each town should have a name.  
  
So we just use here an attribute called name, also for town.  
  
So here we have listed all of the entities together with the attributes that we identified.  
  
But note, for buses and routes, in the text it was clearly marked what should be the identifiers.  
  
It said there's a unique bus number and there's a unique route number.  
  
So we got the identifiers from the text.  
  
For stages, drivers in towns, it was a bit less obvious.  
  
So what happens if we end up with an entity that doesn't have an obvious id?  
  
We just introduce it.  
  
So this is what's called a surrogate identifier, or later on for the tables, a surrogate key.  
  
So we just add an artificial attribute called stage id, employee id, town id, and say that these will be the identifiers.  
  
So later on, if we store information about employees, we will always have to assign a unique employee ID to the employee to make sure that we can identify each entity within this entity type.  
  
So whenever we don't get a clear identifier from the description, we just add what's called a surrogate identifier or surrogate key when it then comes later to the tables.  
  
So now we have here this list of entities and their attributes, and you see already where this is going.  
  
Eventually all of those entities will be turned into tables and the attributes will be attributes of those tables.  
  
But before we do this, we have to talk a bit in more detail about the relationships that we identified.  
  
Thanks for listening.  
  
See you in the next video.