

Hi.  
  
This video will be about database design, about entities and relationships.  
  
Why are we studying database design in detail?  
  
Well, we should note that designing a database is not an easy thing.  
  
So it's important to get to the right tables and it's really non trivial if you start from the raw data to see immediately what the right structure of the data is.  
  
The point here is that normally it's not you who gets to choose which data you want to store, it's somebody, the customer or the boss who wants to store some raw data.  
  
And you have to then figure out how to structure it.  
  
And how do we structure it?  
  
Well, we start by first of all identifying the critical components of the data.  
  
And then we turn those structures that we identified into a diagram.  
  
And this will be the content of the next few videos.  
  
And this particular video now is to identify the essential components.  
  
How does the story that I'm telling at the moment fit into the ground picture of database design?  
  
Well, database design can be roughly modelled into two streams.  
  
So the one starts with the data requirements and this is what we are going to do now.  
  
The other one starts with the processing requirements.  
  
So there are two aspects to modelling data.  
  
One is to look at what is the data we are going to store.  
  
This is the data requirements.  
  
And the second one is how are we actually then using the data, what are the kind of queries that we're going to ask and what are the transactions that take place using this data every day?  
  
And this is what we call processing requirements.  
  
So we will talk a bit about processing requirements towards the end of the whole class.  
  
But for now we really focus on the data requirements.  
  
And so what we do is we turn the data requirements into a conceptual schema that entity relationship diagrams.  
  
Then we turn this schema later on into what we call a logical design, where we just say these are the tables that we need based on this diagram, these are the attributes that they have.  
  
And then finally we move to the physical design.  
  
We actually implement the tables, we implement the integrity constraints and we might add some indices and so on.  
  
For now, we focus on going from the data requirement to the conceptual schema.  
  
So if we start from the data requirement, we just get a text maybe that describes what the data is that we're trying to store and what are the elements and so on.  
  
So basically it's like a description of a part of the world.  
  
And so how do we structure this?  
  
So we structure it in what is called entities and relationships.  
  
So what is an entity?  
  
So an entity is really a thing that we are Trying to describe an object, it could be a certain person.  
  
For example, I'm an entity, Anybody of you is an entity.  
  
As such, if you're trying to describe students, you might be an entity.  
  
An entity type, on the other hand, is a collection of entities that are all similar.  
  
So if we are talking about, for example, the participants in this class, you are all entities, and your entity type would be student in this case.  
  
Or to have a completely different example, we could have each bank account in a bank is an entity, and the entity type here is bank account.  
  
So the entity type is the abstract thing, and then the entity is the concrete instance of attributes are describing entities.  
  
So for example, in an employee database, where we have an employee entity type, everybody has an attribute, for example, which is like the employment date, since when they are employed, what's the salary, what is the specialisation, and so on.  
  
So the attributes are describing the entities, usually just with atomic values such as a number or a string, and so on.  
  
And then finally, an identifier is an attribute or a set of attributes that uniquely identify an entity.  
  
So the employee could have an employee id, and this ID uniquely identifies each instance of the entity type employee.  
  
It's all a bit of abstract, maybe, but the main message here is what we are trying to do is we are trying to identify the objects that the data is describing, the entities, and we group them into entity types, which later on will somehow correspond to the tables in which we store the information about those objects.  
  
And the information about those objects are composed by attributes.  
  
And then the attributes that allow me to identify a concrete entity is what's called the identifier, which will later on correspond to keys of a table.  
  
So this is how we structure the world into entities and entity types.  
  
I must warn you, in the rest of this class, I will often confuse the word entity or entity types.  
  
Even if I say entity, I usually mean entity type, but entity is just shorter and it normally doesn't lead to confusion.  
  
So again, to have some examples, entity types could be person, car, city, and then entities based on these types.  
  
So for the person, we could have John, famously John Smith.  
  
The car could be BMW, something, whatever, and the city could be Edinburgh.  
  
The attributes, of course, for John, Smith would be the first name for the car, it could be a car make, and for the city it could be population size.  
  
And then finally we have identifiers for the person.  
  
We have the national insurance number for the card, the registration number for the city, geographical coordinates, let's say.  
  
So these are the entities, and the entity types between those, we usually have relationships.  
  
And so a relationship is an association between different entities.  
  
For example, we could have an entity type product, an entity type warehouse.  
  
And then we have a relationship saying product is stored in a warehouse.  
  
So that stores relationship has a relationship type, which is just called stores.  
  
So a stores is again of the same level of entity types, is a relationship type.  
  
Then each relationship type has what's called occurrences, relationship occurrences, which are the concrete elements of the relationship.  
  
So in the case of stores, if product one is stored in warehouse three, then this information is what's called a relationship occurrence.  
  
And a relationship identifier is basically usually the composite of the identifiers of the things involved.  
  
So if I want to identify a relationship between a certain product and a certain warehouse, I take the identifier of the product together with the identifier of the warehouse, and this identifies a certain relationship occurrence.  
  
So we saw that we have entities and relationships.  
  
Now we will see how to identify them if we starting from the description of some data and from there on we then move on to the diagram later on.  
  
I hope you enjoyed this video.  
  
See you next time.