

Hi.  
  
In this video we are going to talk about database management systems and database systems in general.  
  
So, first of all, what is the purpose of database systems?  
  
The aim of these systems is to bring the development of information systems closer to the way people think.  
  
The idea is that the information systems should follow our intuition rather than us understanding how the information system operates.  
  
That has some consequences.  
  
Namely, we want to insure three things based on this aim.  
  
Namely, we want to have physical data independence, we want to have access independence, and we want to have logical data independence.  
  
What does this mean?  
  
First of all, we want to be independent of the exact physical structure of the data.  
  
So the user should not need to know how data is stored in the information system.  
  
This is something that should not be of our concern.  
  
Access independence is similarly, if we want to access the data, we should only need to know what kind of data is stored there.  
  
We should not need to know exactly how, at which location it is stored.  
  
So the query should be just about employees, and it shouldn't be show me the employees that are stored on drive C in directory something, something.  
  
And closely related to this is the logical independence.  
  
So what matters to the user should only be the logical structure of the data and nothing else.  
  
So only the logical structure of the data changes.  
  
The user should notice this.  
  
So we have these three notions of independence and all three together should be somehow maintained.  
  
And the role of maintaining these principles is the database management system.  
  
So if you look at this diagram here, you could think of the database as some chunk of data lying somewhere in a corner, so to speak, and people are trying to access it.  
  
And people and users are of different types.  
  
It could be either some applications accessing the data or the user directly accessing the in the data.  
  
In order to make sure that these notions of independence are maintained, what we introduce is this layer in between the user and the database.  
  
We kind of encapsulate the data.  
  
We have this interface that allows the user to interact only in certain ways with the data.  
  
We cannot just directly interact with the raw data, but we must have some kind of layer in between.  
  
And this is the database management system.  
  
And so the database management system provides a layer with respect to one user.  
  
But you see already on the diagram that there might be several users.  
  
So there must be some kind of way of ensuring that several users are getting organised in such a way that they don't influence each other.  
  
This leads us to some tasks of a database management system.  
  
So first of all, the management system has to allow us to create the data in the first Instance, so we need some kind of data definition language.  
  
Secondly, we want to be able to query and modify the data.  
  
This is a data manipulation language.  
  
These are the obvious things and they are not yet related exactly to this notion of independence.  
  
But now we come to more advanced tasks.  
  
So first of all we need to be able to support to store large amounts of data.  
  
It's not enough to just have like three lines of data lying around and be able to query it quickly.  
  
It might be, I don't know, terabytes of data and we still have to be able to come by the answer fairly quickly.  
  
So we need to have efficient access.  
  
The next task would be to create something that is durable.  
  
So for example, databases are used in applications that can have massive consequences.  
  
Let's think about something not so massive, but still personally important.  
  
If you go for holidays and you book the holiday, you want to be able to know that even if the computer crashes of the company once they reboot, the data is still there.  
  
So there has to be some kind of guarantee that the data is stored in a durable way and nothing bad can happen.  
  
So we need durability and similarly we need consistency.  
  
It happens very rarely, but it does happen that two people are getting booked on the same seat in the flight, which is very bad.  
  
So I don't know how it's possible.  
  
But in general these are the things that you want to avoid.  
  
You want to keep your data consistent.  
  
And because we have this several users that access all the data in the same time, we want to have some kind of form of guaranteed isolation.  
  
So these were the five tasks of a database management system that we have listed here.  
  
Also on the slide, some of them can be also summarised with this term asset.  
  
So ACID stands for atomicity, Consistency, isolation and durability.  
  
Consistency, durability are pretty self explanatory.  
  
Let's go back to atomicity.  
  
So all parts of a transaction should succeed or none.  
  
So suppose you have a transaction in the bank, the customer gets a loan from the bank and his account should be then updated with a certain amount.  
  
Suppose only the loan gets basically assigned to the customer, but then the update of the account fails, or vice versa.  
  
You don't want this, so you want either both actions to succeed or none of them.  
  
So this is a very basic example of atomicity.  
  
And isolation is just that if several users access the data simultaneously, you want to make sure that they do not interact unintendedly.  
  
So you don't want that somebody enters a half baked, half finished record into the table and the other user gets this already into his query answer and you can There are several levels of isolation and modern database management systems allow you to exactly specify how users should be isolated from each other.  
  
So these were the tasks of the database management system and also this term acid, which is widely used in computer science.  
  
I hope you enjoyed this video.  
  
Thanks for listening.  
  
See you next time.