CTE's in Stored Procedures, Functions and Views

Transcription of Video:

Now on to CTE's in stored procedures, functions and views. When I first started doing this presentation several years ago, people would ask, "oh, can the CTE be used in a function or can be used in a view"? And the answer to that is yes. And as we saw above, we actually already put the CTE into a procedure. So we know that that works. So let's just create a procedure here, that's called Show departments. And inside of that, we're going to have the department CTE, that just runs a query and displays the results. And once that procedure is created, we're going to run it. And what we get back is just the list from that query. So no problem CTE's work great in stored procedures. How about an inline table valued function, that's where it's a SQL Server function that only has a single statement, and it returns a table. So what we've got is we're going to create the function. Okay, so we're gonna do is we're going to create the function in there to the same CTE we've been looking at before, and we're going to compile that. And then to use it, we just say select star from the function name, then with parentheses after it. And there we go, just a different way of getting access to that same CTE. And then we can also put them in multi statement functions, although you run into different issues, performance wise, around multi-segment. But we're going to have two different CTE's in there that are both inserting into the result set that is the return value from the from the multivalued. From the multivalued function. And then when we run it, there we go, we get the results from both of the queries that were shown up above where the parent equals one and the parent equals three, which is camping and snow sports. All right, how about in a scalar function, a function that does not return a result set? Well, the difference with this is, yes, we can use the CTE in there. But we have to assign the results into a return value rather than just displaying it as a result set. Let's take a look. When we compile that we select, and we're not selecting star from we're actually just selecting the function name. And when we run that we get back for as our result value that's coming from the count for the number of top level departments. Then on to creating a view with a CTE. A lot of people ask about performance issues around views and CTE's. Well, if you've written a query that's going to perform poorly, it's going to perform poorly when it's in a view, that doesn't really change anything. If you've got a query that works great. It should work great when you're working in a view. The difference is sometimes SQL Server will get a little bit confused with filtering and things on the performance when you're when you have nested views or multiple levels of nested views. So I kind of recommend staying away from that. But as far as putting a CTE into a view, you do it just like this. We select star from the view and we get the same results that we've been getting all along.