procedures. Now, for those new to ABAP, there are a couple of techniques for modularizing code that you should really concentrate on.

When we want to split our programs into separate modules, we can use what are called procedures. And procedures are processing blocks that we call from our main ABAP program. And procedures come in the form of subroutines or subprograms and function modules.

We use subroutines mainly for local marginalization of our code. And what I mean is, a small modular unit of code that is self-contained, that is called from the program in which they are defined.

Wrapping it up into a subprogram and just moving it to the end of your program. You know, right down at the bottom. Still in the same file, but just moved away in its own little container.

So, we can use these subroutines to write functions that we use many times within our programs. And at the same time, any ABAP program can include subroutines.

Now, function modules, they work a bit different. They allow us to create modular blocks of code that are held separate to our ABAP program, and they can be called from any other program. **So, subroutines are local to our own program. Function modules are global and can be called from any program.**

an SAP system is packed with function modules. They encapsulate all the processing logic used within the business system. And SAP have made sure that it can be reused both by their own development teams and by SAP customers.

Now, when we looked at include programs, I mentioned an include program cannot accept any parameters. Procedures are different, because they have an interface for transferring data from our calling program to the procedure itself. So, because we can pass data into a procedure, it also means we can define data definitions within the procedure itself that are only available to that procedure.