When we are writing our programs using open SQL, we need to bear in mind the concepts of authorization in an SAP system.

Now an SAP system has its own security tools to ensure users can only access the data they are authorized to see. This includes individual fields as well as individual records. The way authorizations are set up can also limit how the data is used. Whether a user can only display information or whether they can modify the data such as adding, changing, and even deleting records. And all these rules that make up a user's authorizations as stored as authorization objects. Now we're not going to dig deep into authorizations here, but as a general rule a user is assigned a relevant authorization profile or composite profile against their user record.

Which for informational purposes is managed through transaction SU Z01. This authorization profile then gives the user the correct rites to carry out their job and SAP delivers lots of predefined authorization profiles with their bay system. The system of administrators can use and enhance to apply to their users for example, there are standard developer roles or HR administrator roles. The system administrators more than likely take these delivered authorization profiles files, tweak them a little bit to ensure they work just how their company wants them to work, and then apply them to individual users.

Once a user is assigned a role, and tries to execute a program, depending on the authorization role of the user, they will be allowed to execute the transaction or not. And a typical example is being able to run transaction SE38. The ABAP editor. A user can be given a role to allow them to access that transaction, but also it can be tweaked so that yeah, they can access it, but only in display mode. Or in a live system maybe, they can access the transaction in display mode. And be allowed to debug the program in the live environment, but not be able to make changes to the code itself. So, coming back to our ABAP programs, where specific authorizations have not been implemented, we can get our programmers to carry out an authority check using the ABAP statement authority check. This has to be used if the transaction or the ABAP program itself is not sufficiently protected by the standard authorization profiles already set up in the system. Now, we won't go any further into authorization checking in this course because it is a massive topic in itself. But you do need to be aware of authorizations. And make sure any programs that you create take into account authorizations already set up by the system or, if required, implement your own authorization checking, as just mentioned.