Now, internal tables only ever exist when your program is running. They're internal to a running program only.

When you write your ABAP program, you've specified the structure for an internal table, and you write your program to make use of it. And once your program finishes using the internal table, it is then destroyed.

When you start to use internal tables in your programs you will see that they behave the same way as structures that you define in your program. And these structures are called work areas.

The only difference is that your structures have only one line whereas an internal table can have an unlimited number of lines just like a normal database table.

They allow us to hold results of calculations that we can use later in our program.

They allow us to hold records of data that we can access really, quickly instead of having to read data from database tables repeatedly and a whole host of other reasons. And internal tables are flexible because they can be defined using any number of other defined structures.

This allows you to have many normal table structures, grouped together and placed into one internal table.

You will see as we work with internal tables that the basic form consists of a table body which is all the records within the table. And then we have a header record.

Now remember, the header record means we're working with the older style internal table. But for the newer style, we would have the table body in a separate work area. Defined the same structure as the table but it is kept separate and works separately.

Now flipping back to the older method, the header line or work area is used when we read individual records from our internal tables. Well, when we read a record for the older style table that record gets placed into the header record which we can access directly.

For the newer style table, we read a record into a work area. Very similar to a header record but remember it's separate. And, when we want to add records to an internal table the older style, we placed the content into the header area.

For the newer style, we place content into work area and then we transfer the header record or work area into the body of the internal table which results in us creating a new record.

today it's best practice to use a separate work area. You see, using a header line has several restrictions and when SAP introduced the possibility to work with internal tables without header lines. By using our own work areas, we were given the ability to create multi-dimensional tables.

Which as I mentioned before with the multi-dimensional rays, it's just tables within tables.

There are some restrictions, the architecture of an SAP system limits the size of our internal tables to around two gigabytes in size.

You will also need to bear in mind, just how powerful your specific SAP system is, you know consider the hardware, the operating system, the amount of memory there is, and so on.

Now even though we can create internal tables up to two gigabytes in size, it's best practice to keep them as small as possible. Because if you have multiple programs taking up so much memory in your system, you'll run the risk of making your whole system run really, slow.