

SQL Commands – Airport DB

Here's a basic query:

```
SELECT AirportName, CountryName,  
Terminals  
FROM Airport  
WHERE Terminals > 2  
ORDER BY AirportName ASC;
```

SQL Commands – Airport DB

What if we want to get data from both tables?

- I want a list of airports, the number of terminals and the currency used in the country

This needs to come from both tables...

SQL Commands – Airport DB

```
SELECT AirportName, Terminals,  
Currency  
FROM Airport, Country  
WHERE Terminals > 2  
ORDER BY AirportName ASC;
```

This doesn't work!

SQL Commands – Airport DB

To deal with fields across two tables, we need to use **dot notation** to specify where the data is going to come from

This means SQL can get the data from the right place...

SQL Commands – Airport DB

```
SELECT Airport.AirportName,  
Airport.Terminals, Country.Currency  
FROM Airport, Country  
WHERE Airport.Terminals > 2  
ORDER BY Airport.AirportName ASC;
```

Every time you specify a field, add the table name in front of it

SQL Commands – Airport DB

But this still doesn't quite work properly...

SQL seems to return multiple sets of data for each airport

It's getting confused...

SQL Commands - Airport DB

STOP

If you're following this in (or before) class stop here and let me explain the solution first

I can explain it better irl than I can on a slide

SQL Commands - Airport DB

No, really

STOP RIGHT THERE

Do not pass Go. Do not board the plane...

SQL Commands – Airport DB

We need to tell SQL how the tables are **joined** together

It needs to know about the **Foreign Keys**

We add this in the **WHERE** section

SQL Commands – Airport DB

```
SELECT Airport.AirportName,  
Airport.Terminals, Country.Currency  
FROM Airport, Country  
WHERE Airport.Terminals > 2 AND  
Airport.CountryName = Country.CountryName  
ORDER BY Airport.AirportName ASC;
```

The AND is really important

We've just added the FOREIGN KEY field

SQL Commands – Airport DB

```
SELECT Airport.AirportName,  
Airport.CountryName, Country.Currency  
FROM Airport, Country  
WHERE Airport.CountryName =  
Country.CountryName;
```

You can do the same sort of query without any search criteria or ORDER BY but you still have to add the Foreign Key if it's multi-table

SQL Commands-- Airport DB

Note that there is another way to link the two tables together.

This uses a command called **INNER JOIN**

I think this is trickier to use in an exam, so I don't teach it. But it might be worth knowing that it does the same thing.

You shouldn't be given code in an exam that asks you to write SQL using INNER JOIN, but it's worth knowing that it exists as a method. If you are given code that does then I will be complaining...