



Natural joins allow the selection from two tables without the need to write a complex where clause ...

```
select sd_section_name  
      , si_stock_number  
from section_details  
natural join stock_information;
```

Here, the join is based on columns with the same name in both tables, this can be quite a problem if audit columns are named the same as well as the key columns ! In fact this would not work on the above tables because the columns are not named the same.



Cross joins allow the selection from two tables, deliberately creating a cartesian product

```
select sd_section_name  
       , si_stock_number  
from section_details  
cross join stock_information;
```

No attempt is made to find columns which are identical or links with each other.



A join allows the User to name the columns that are to be joined, this is to be used when the tables share common column names but there are other common columns which should not be joined ...

```
select sd_section_name  
       , si_stock_number  
from section_details  
join stock_information using (section_no);
```



An On join allow the selection from two tables which have common Data, but not common column names

```
select sd_section_name  
      , si_stock_number  
from section_details  
join stock_information on  
      (sd_section_number = si_section_number);
```



To join more than two tables which have common Data, but not common column names

```
select sd_section_name  
    , si_stock_number  
from section_details  
join stock_information on  
    (sd_section_number = si_section_number)  
join stock_sales on  
    (si_stock_no = ss_stock_no);
```



This allows the User to specify whether the record should appear if a join fails on the right hand side of the join, in other words if the stock_information has no record matching with the section_details table ..

```
select sd_section_name  
       , si_stock_number  
from section_details  
left outer join stock_information  
on sd_section_number = si_section_number;
```



This allows the User to specify whether the record should appear if a join fails on the left hand side of the join, in other words if the `section_details` has no record matching with the `stock_information` table ..

```
select sd_section_name  
       , si_stock_number  
from section_details  
right outer join stock_information  
on sd_section_number = si_section_number;
```



This allows the User to specify whether the record should appear if either side of the join fails ..

```
select sd_section_name  
      , si_stock_number  
from section_details  
full outer join stock_information  
on sd_section_number = si_section_number;
```