La idea de los INNER JOINs es poder acceder a campos de una tabla Z relacionada con una tabla X, posiblemente tambien teniendo que pasar por una tercera (o hasta 4ta o...n-esima) tabla relacionada Y.  
  
X---Y----Z  
  
Casi siempre se relacionan por IDs ... asi que digamos que X son compradores, Y son mercaderias y Z son proveedores, como se puede pensar... debo poder conocer el proveedor de un producto que compro un comprador ante una reclamacion (por ejemplo). Por supuesto cada comprador tiene su ID (podria ser su DNI), cada producto su ID\_producto y cada proveedor su ID\_proveedor.  
  
En general seria algo asi....

Código SQL:

[Ver original](http://www.forosdelweb.com/f86/inner-join-ejemplos-para-aprender-952003/)

1. SELECT \_\_(campos)\_\_\_ FROM \_\_\_(tabla1)\_\_\_
2. INNER JOIN \_(tabla2)\_\_\_\_
3. ON (id\_tabla1=id\_tabla2)
4. WHERE condiciones\_para\_cualquier\_campo\_de\_cualquier\_tabla

o para 3 tablas (como en el mini-ejemplo)

Código SQL:

[Ver original](http://www.forosdelweb.com/f86/inner-join-ejemplos-para-aprender-952003/)

1. SELECT \_\_(campos)\_\_\_ FROM \_\_\_(tabla1)\_\_\_
2. INNER JOIN \_(tabla2)\_\_\_\_
3. ON (id\_tabla1=id\_tabla2)
4. INNER JOIN \_(tabla3)\_  ON
5. (id\_tabla2=id\_tabla3)
6. WHERE condiciones\_para\_cualquier\_campo\_de\_cualquier\_tabla

donde las condiciones WHERE o cualquier LIMIT va al final y son opcionales.  
  
Asi se puede hacer PERFECTAMENTE pero es mucho mas facil de seguir y evitar confundirse si usamos "alias" para los nombres de las tablas:  
  
o... para 3 tablas

Código SQL:

[Ver original](http://www.forosdelweb.com/f86/inner-join-ejemplos-para-aprender-952003/)

1. SELECT t1.\* FROM TTTT AS t1
2. INNER JOIN  TTTT AS t2 ON
3. t1.id\_ = t2.id\_
4. INNER JOIN TTTT AS t3 ON
5. t2.id\_  = t3.id\_

y... para 4 ...

Código SQL:

[Ver original](http://www.forosdelweb.com/f86/inner-join-ejemplos-para-aprender-952003/)

1. SELECT t1.\* FROM TTTT AS t1
2. INNER JOIN  TTTT AS t2 ON
3. t1.id\_ = t2.id\_
4. INNER JOIN TTTT AS t3 ON
5. t2.id\_  = t3.id\_
6. INNER JOIN TTTT AS t4 ON
7. t3.id\_  = t4.id\_

Donde TTTT son los nombres de las distintas tablas. Puede verse como este tipo de estructura es totalmente predecible y se puede generar facilmente en cualquier ORM.