

~~Spring~~

# JPA Annotations

~~to view 124~~

~~ManyToOne~~  
~~JPA~~

1

1 : 1

~~Person Employee~~

Account

1

1

Details

~~Department~~

@OneToOne\*  
Detail detail

...

puede ser  
en cualquier

1 : N

Employee

N

1

Department

...

@ManyToOne\*  
Department dep

@OneToMany(mappedBy="dep")  
List<Employee> employees;

N : N

Brand

N

N

BP

N

1

Provider

en ER

en BRD

...

Post

\*\* JPA will generate N:1 table but  
we want to customize it, so:

@JoinTable

(name="NNtable Name",  
joinColumns=@JoinColumn(name="a"),  
inverseJoinColumns=@JoinColumn(name="b"))

owner      decimos uno de los 2  
                como owner y  
                otro como target

@ManyToMany\*\*  
List<Provider> providers

target  
en Code  
First  
(no hay  
tabla  
N:1)

@ManyToMany  
(mappedBy="providers")  
List<Brand> brands

esto casará:

Brand	UNtable Name	Provider
id	b   a	id

\* Usamos

→ pq muestra Entity  
no tiene el campo FK  
sino el objeto de la  
otra tabla

@JoinColumn(name="columnaFK", referencedColumnName="idOtraTabla")

ya que así sabrá cómo enlazar las tablas y hacer la FK  
y positivamente evita una referencia recursiva,

donde <sup>depende</sup> tiene lista de empleados y ¿emp un dep. y vuelve a buscar? No

si son ~~FK~~ compuestas las FK, usamos

@JoinColumns({@JoinColumn(), @JoinColumn()})

Si no quiero traerme la otra entidad, al menos agrego las columnas de la FK