# Que es un ORM

Mapeo objeto relacional. Es una capa que va a hacer mucho más sencillo la conexión entre la BB.DD y el código.

Cada fila de la BB.DD se convertirá en una instancia de objeto, y cada columna, en uno de sus atributos.

## Ventajas

* Independencia sobre la BB.DD
* Propio lenguaje. Por ejemplo, Hibernate tiene HQL. Lo que lo hace independiente de la BB.DD. Se puede portar y usar un ORM en distintas BB.DD.

## Desventajas

* Da menos rendimiento, sobr4e todo en relaciones 1 a n, y n a n. Pero usándolo con cuidado, no se nota tanto.
  + Solución parcial: usando lazy, solo se trae los datos que necesitas en cada momento.

JPA = Java Persistent API (es la base de todo)  
Hibernate = es un framework construido sobre JPA.  
Pasar de uno a otro es sencillo.

# Arquitectura.

* Objetos/archivos de configuración: solo se tocan una vez al principio, y, si acaso, luego cuando da error.
* Transacciones: funcionan igual que en BB.DD normal
  + Es otro objeto que tiene Hibernate
* Query: tiene su propio lenguaje (HQL). Es muy sencillo, parecido al SQL.

# Mapeo. Relaciones y cardinalidades.

Podemos hacer un mapeo de cualquiera de las cardinalidades.

# Herencia en BB.DD.

Podemos tener dos tablas que queramos después que sean un objeto abstracto

# Clases persistentes

Las clases que genera el ORM heredan de Serializable.

Además, si no ponemos los objetos en el persistence.xml, no nos lo detectará

# Parametrizando (para evitar inyecciones SQL)

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente