1.JPA是什么?

实际上是规范

类似于JDBC规范

Java Persistence API：用于对象持久化的 API

2. JPA和Hibernate的关系

JPA 是 hibernate 的一个抽象（就像JDBC和JDBC 驱动的关系）：

– JPA 是规范：JPA 本质上就是一种 ORM 规范，不是 ORM 框架 —— 因为 JPA 并未提供 ORM 实现，它只是 制订了一些规范，提供了一些编程的 API 接口，但具体 实现则由 ORM 厂商提供实现

– Hibernate 是实现：Hibernate 除了作为 ORM 框架之外， 它也是一种 JPA 实现

3.为什么用？

1.少量修改，只要都是JPA框架就都可以通用

2. 在JPA 框架下创建实体和创建 Java 类一样简单，只需要使用 javax.persistence.Entity 进行注释；

3. JPA的查询语言是面向对象的，JPA定义了独特的JPQL

4.支持面向对象的高级特性

4.什么构成了JPA？

1.映射关系-ORM 映射元数据

什么意思？

就是数据库中的数据和实体类怎么一一对应？

表中的列名对应哪个属性？

2. JPA 的 API：用来操作实体对象，执行CRUD操作，框架 在后台完成所有的事情，开发者从繁琐的 JDBC和 SQL代 码中解脱出来。

JPA在java中提供的各种方法，只需调用，不需要写复杂的SQL语句，就可以实现CURD

3. 查询语言（JPQL）

JPA自身划分的查询语言

5.JPA的使用流程

1.JPA自身的配置文件+映射关系----xml文件

• 创建 persistence.xml, 在这个文件中配置持久化单元

– 需要指定跟哪个数据库进行交互;

– 需要指定 JPA 使用哪个持久化的框架以及配置该框架的基本属性

2.Java中的实体类，实体类就是数据

创建实体类, 使用 annotation 来描述实体类跟数据库表之 间的映射关系.

3. 使用 JPA API 完成数据增加、删除、修改和查询操作

调用Java中方法即可

**JPA独特的操纵类为：EntityManager**

– 创建EntityManagerFactory (对应 Hibernate 中的 SessionFactory);

– 创建EntityManager (对应 Hibernate 中的Session);

6.具体使用

1.导入依赖



2. persistence.xml-------JPA自身+映射关系配置类