

主表——> 外键 <——从表 关系：

外键：从表中一列的数据来源于主表的主键，我们把这一列称之为外键。外键的数据可以是null，当不为null时必须是主表主键中出现过的

主表

cust_id	cust_name	cust_address
1	传智播客	北七家
2	黑马程序员	软件园
3	中粮商务公园	留仙二路
4	华强北	华强北

从表

lkm_id	lkm_name	lkm_gender	customer_id
1	联系人 1	male	1
2	联系人 2	female	1
3	联系人 3	male	2
4	联系人 4	female	2
5	联系人 5	male	3
6	联系人 6	female	3
7	联系人 7	male	4
8	联系人 8	female	4

主表为客户表保存所有的客户信息一个客户可以有多个联系人

从表保存联系人多个联系人属于一个客户

多表关系类型：

- 1、一对多（实际使用最多）
- 2、多对一（实际使用最多）
- 3、多对多（实际使用较多）
- 4、一对一（实际基本不使用）

单向多对一实体类的关系映射

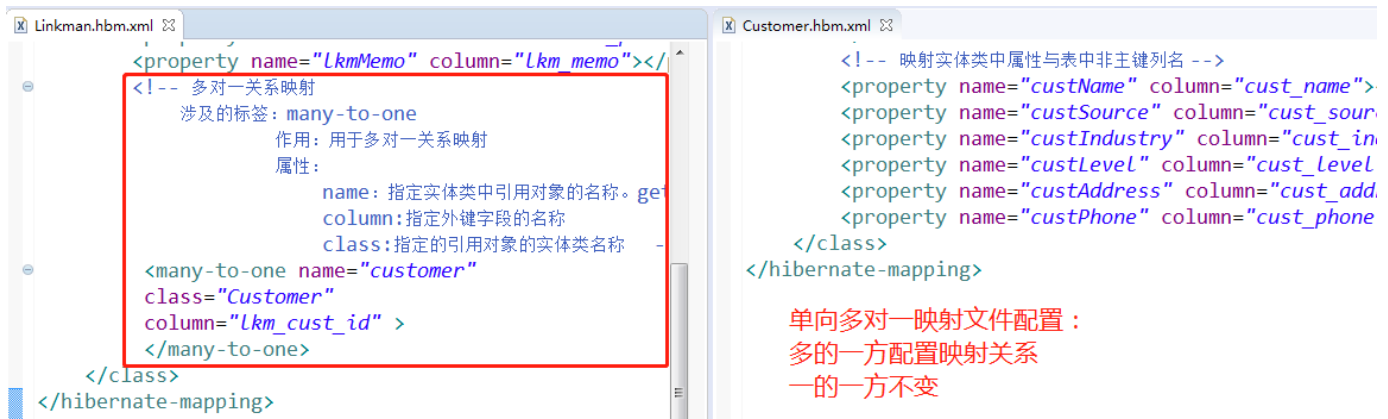
```
/**
 * 对应数据库表的实体JAVABean类 实现序列化 并生成get/set方法
 * 主表（客户）信息
 */
public class Customer implements Serializable {
    private Integer cusId;
    private String custName;
    private String custSource;
    private String custIndustry;
    private String custLevel;
    private String custAddress;
    private String custPhone;
```

单向多对一映射的实体类关系

```
/**
 * 联系人管理Bean信息
 * 从表 客户的联系人信息
 */
public class Linkman implements Serializable {

    private Long lkmId;
    private String lkmName;
    private String lkmGender;
    private String lkmPhone;
    private String lkmMobile;
    private String lkmEmail;
    private String lkmPosition;
    private String lkmMemo;
```

```
// 单向多对一关系映射：多个联系人对应一个客户
private Customer customer;
```



单向一对多实体类的关系映射

```
/**
 * 对应数据库表的实体JAVABean类 实现序列化 并生成get/set方法
 * 主表（客户）信息
 */
public class Customer implements Serializable {
    private Integer cusId;
    private String custName;
    private String custSource;
    private String custIndustry;
    private String custLevel;
    private String custAddress;
    private String custPhone;

    //单向一对多关系映射：一个客户可以包含多个联系人
    private Set<Linkman> linkmans =
        new HashSet<Linkman>(0);
```

```
/**
 * 联系人管理Bean信息
 * 从表 客户的联系人信息
 */
public class Linkman implements Serializable {

    private Long lkmId;
    private String lkmName;
    private String lkmGender;
    private String lkmPhone;
    private String lkmMobile;
    private String lkmEmail;
    private String lkmPosition;
    private String lkmMemo;
```

单向一对多实体类映射关系



双向一对多或多对一（主从都配置）

```

/**
 * 对应数据库表的实体JAVABean类 实现序列化 并生成get
 * 主表（客户）信息
 */
public class Customer implements Serializable {
    private Integer cusId;
    private String custName;
    private String custSource;
    private String custIndustry;
    private String custLevel;
    private String custAddress;
    private String custPhone;

```

```

//双向一对多关系映射：一个客户可以包含多个联系人
private Set<Linkman> linkmans =
    new HashSet<Linkman>(0);

```

```

/**
 * 联系人管理Bean信息
 * 从表 客户的联系人信息
 */
public class Linkman implements Serializable {

    private Long lkmId;
    private String lkmName;
    private String lkmGender;
    private String lkmPhone;
    private String lkmMobile;
    private String lkmEmail;
    private String lkmPosition;
    private String lkmMemo;

```

```

// 双向多对一关系映射：多个联系人对应一个客户
private Customer customer;

```

```

Customer.hbm.xml
<property name="custLevel" column="c
<property name="custAddress" column=
<property name="custPhone" column="c

<!-- 一对多关系映射 涉及的标签：set:作用：用于
table: 指定的是集合元素所对应的数据库表。
属性：class: 指定集合元素对应的实体类 -->
<set name="linkmans" table="cst_link
    <key column="lkm_cust_id"></key>
    <one-to-many class="Linkman" />
</set>
</class>
</hibernate-mapping>

```

```

Linkman.hbm.xml
涉及的标签：many-to-one
作用：用于多对一关系映射
属性：
    name: 指定实体类中引用对象
    column: 指定外键字段的名称
    class: 指定的引用对象的实

<many-to-one name="customer"
class="Customer"
column="lkm_cust_id" >
</many-to-one>
</class>
</hibernate-mapping>

```

多表映射具体的配置内容：重要

一对多：

<!-- 一对多关系映射 涉及的标签：

set:作用：用于映射set集合属性。

属性：

name: 指定集合属性的名称。找get/set方法后面的部分

table: 指定的是集合元素所对应的数据库表。在一对多映射时可以不写。

key:作用：用于映射外键。

属性：**column:** 指定外键的名称。

one-to-many:作用：指定当前实体和集合元素之间的关系是一对

多。

属性：class：指定集合元素对应的实体类 -->

```
<set name="linkmans"
    table="cst_Linkman" >
    <key column="lkm_cust_id"></key>
    <one-to-many class="Linkman" />
</set>
```

<!-- 一对多关系映射 涉及的标签：

set：作用：用于映射set集合属性。

属性：

name：指定集合属性的名称。找get/set方法后面的部分

table：指定的是集合元素所对应的数据库表。在一对多映射时可以不写。

key：作用：用于映射外键。

属性：column：指定外键的名称。

one-to-many：作用：指定当前实体和集合元素之间的关系是一对多。

属性：class：指定集合元素对应的实体类 -->

```
<set name="linkmans"
    table="cst_Linkman" >
    <key column="lkm_cust_id"></key>
    <one-to-many class="Linkman" />
</set>
```

多对一：

<!-- 多对一关系映射

涉及的标签：many-to-one

作用：用于多对一关系映射

属性：

name：指定实体类中引用对象的名称。get/set方法后面的部分。

column：指定外键字段的名称

class：指定的引用对象的实体类名称 --

>

```

<many-to-one
  name="customer"
  class="Customer"
  column="lkm_cust_id" >

</many-to-one>

<!-- 多对一关系映射
      涉及的标签: many-to-one
      作用: 用于多对一关系映射
      属性:
          name: 指定实体类中引用对象的名称。get/set方法后面的部分。
          column: 指定外键字段的名称
          class: 指定的引用对象的实体类名称 -->

<many-to-one
  name="customer"
  class="Customer"
  column="lkm_cust_id" >
</many-to-one>

```

多表映射的其它配置：作用于映射文件上的配置

- 1、更新时的联级保存更新配置 **cascade="save-update"**
- 2、双向保存时的主表放弃关联关系维护配置，只能在set标签上 **inverse="true"**

在Hibernate中实现

单向一对多，多的一方（从表映射不配置映射关系）

单向多对一，一的一方（主表映射不配置映射关系）

双向一对多或多对一，主表 从表映射都必须配置映射关系