

Ict2 课后作业

题目一

1、什么是正向工程、逆向工程

正向工程：

就是将数据库设计软件（例如 ERWin）画好的数据表图通过设计软件在数据库中生成最终的数据表；

CDM—>PDM

PDM—> Database

PDM—>OOM

逆向工程：

把数据库中的数据表映射到设计软件中，以图表显示

(1) 从处理语言（Process Language）逆向生成业务处理模型（BPM）

(2) 从对象语言（Object Language）逆向生成面向对象模型（OOM）

(3) 从数据库（Database）或数据库脚本逆向生成物理数据模型（PDM）

(4) 从XML定义（XMLDefinition）逆向生成XML模型

[https://wenda.so.com/q/1384354492062008?](https://wenda.so.com/q/1384354492062008?src=140&q=%E6%95%B0%E6%8D%AE%E5%BA%93%E6%AD%A3%E5%90%91%E5%B7%A5%E7%A8%8B)

[src=140&q=%E6%95%B0%E6%8D%AE%E5%BA%93%E6%AD%A3%E5%90%91%E5%B7%A5%E7%A8%8B](https://blog.csdn.net/tianlesoftware/article/details/6884463)

<https://blog.csdn.net/tianlesoftware/article/details/6884463>

2、列举聚集、弱实体、细化/泛化的实用例子。

聚集：

在实际操作中，我们经常需要一些特点数据的，比如：1.我们常常要确定表中的行数；2.获得表中行组的和；3.找出表列（或所有行或某些特定的行）的最大值、最小值和平均值。

上面那些例子都是都是需要对表的数据，而不是实际表中的数据。因此根据表中的实际数据项目返回来统计处理，是对资源和时间的一种浪费。而我们想要的其实是关于表本身的信息的汇总。

下面给出SQL的聚集函数

函数说明

AVG（）返回某列的平均值

COUNT() 返回某列的行数

MAX（）返回某列的最大值

MIN() 返回某列的最小值

SUM（）返回某列值之和

part 1 AVG（）函数

AVG（）通过对表中行数计数并计算特定列值之和，求得该列的平均值。AVG()可用来返回所有列的平均值，也可以用来返回特定列或行的平均值。

```
SELECT AVG(users.user_age) AS avg_users FROM users
```

1

上面的语句将会返回users。user_age的平均值，为avg_users作为别名，来显示返回。

tip：为了获得多个列的平均值，必须使用多个AVG（）

part 2 COUNT 函数

COUNT函数进行计数。可利用COUNT()确定表中行的数目或符合特定条件的行的数目。他有两种使用方

式：1.使用COUNT(*)对表中 行的数目 进行计数，不骨干列表中包含的是空值还是非空值。2.使用COUNT (column) 对特定列中具有值的行进行计数u，忽略null值。

```
SELECT COUNT(*) AS NUM_USER FROM users
```

1

上面这种句子，就会返回users表中行数。

```
SELECT COUNT(user_password) AS NUM_USER FROM users
```

1

这条语句将会对具有user_password的行进行计数

part 3 MAX () 函数

MAX函数返回指定列中的最大值。MAX () 要求指定列名。

```
SELECT MAX(users.user_age) AS USER_BEST FROM users;
```

1

上面的语句将会返回users表中 年纪最大的值 。

part 4 MIN () 函数

和MAX () 的功能相反，返回指定列的最小值

```
SELECT MIN(users.user_age) AS USER_BEST FROM users;
```

1

part 5 SUM () 函数

SUM () 函数用来返回指定列值的和 (总计) 。

下面举一个例子，orderitems表包含订单中实际的物品，每个物品有相应的数量。可如下检索所订购物品的总数：

```
SELECT SUM(users.user_age) AS USER_BEST FROM users;
```

1

上面的例子将会计算出 user.user_age这一列的和 。

SUM () 也可以用来合计计算值。在下面的例子中，合计每项物品的item_price * quantity ，得出总的订单金额。

```
SELECT SUM(item_price * quantity) AS total_price FROM orderitems WHERE order_num = 2005;
```

1

part 6 聚集不同的值

这是MySQL5.0.3新增的功能：以上的五个聚集函数，就恶意如下使用：

1.对所有的行执行计算，指定ALL参数或不给参数（因为默认就是ALL）

2.如果指向包含不同的值，就指定DISTINST参数 来限定 。

下面的例子使AVG()函数返回特定供应商提供的产品的平均价格。它与上面的语句相似，但是使用了DISTINCT参数 来限定

```
SELECT AVT(DISTINCT prod_price)AS avg_price FROM products WHERE vend_id = 1003;
```

1

part 7 组合使用聚合函数

实际上 SELECT函数可以包含多个 聚集函数 如下：

```
SELECT COUNT(*) AS num_item,  
MIN(prod_price) AS price_min,  
MAX(prod)price) AS price_max,  
AVG(prod_price) AS price_avg
```

FROM prouducts

https://blog.csdn.net/qq_36120793/article/details/78823127

弱实体：

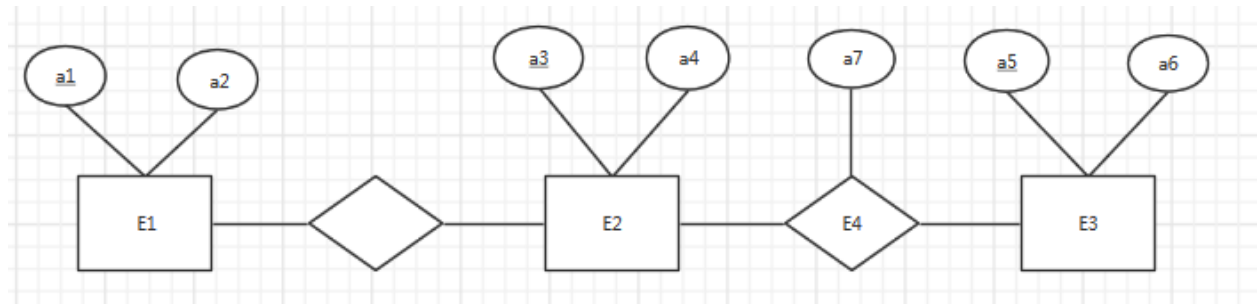
在户籍信息管理系统中，家庭成员的信息以户主的存在为前提，家庭成员实体是弱实体。

在出版书籍的过程中，书籍的存在以出版社的存在为前提，书籍是弱实体。

细化/泛化：

英语与商务英语、法律英语、医学英语

生物与动物、植物、微生物



题目二

