

INTERNET DATABASES QUESTION

Shihang 2016.1.7

1. Which problem will occur if only satisfy x,y,z -NF relationship model?
2. Advantage and Disadvantage of using SQL trigger or Stored Procedure?
3. What is the advantage of View?
4. DBMS, CRM, ERP (multiuser databases application)
5. Three layer architecture
6. Persistent object (durability)
7. Entity Relationship
8. Functional Dependency
9. ER diagram
10. SQL statement
11. Mysql_fetch_object / row / assoc
12. Table
13. CSS -3ways

（以上 13 个问题是最后一节课内老师叙述内容，都是对于初学者易懂的扫盲知识，可能作为考点，相比我们串讲所说的简单很多，且 php 部分他说不为考查重点，考试方式为六选四）
归纳和总结仅供参考，时间关系排版格式未做矫正，致以抱歉。

2016/1/7 史航

1. Which problem will occur if only satisfy x,y,z -NF relationship model?

NF: 范式是表示数据库内数据关系模式的集合，代表一个关系内各属性之间的联系的合理化程度。对于要求而言：1NF<2NF<3NF....<nNF 具体的问题分析按照以下要求进行合理分析即可。

串讲时提到过几个范式之间的一些特点如：

第一范式是作为一般数据库关系里面最基础的要求,每个属性不可划分，两个属性不能合为一个属性，eg: 属性 A:专业名称，B: 班级名称，此时就不可出现属性 C: 专业+班级名称。

第二范式在第一范式的基础上消除了非主属性对于码的部分函数依赖关系。（函数依赖：假如在建表的时候知道一个 pro_id 对应唯一的 pro_price 那么我们没必要在 table 里面写下 pro_price 这个属性）

第三范式在第二范式的基础上，消除了非主属性对于码的传递函数的依赖。（记住属性之间的传递函数关系即可，范式提升的目的是为了防止数据冗余过大，方便数据库维护的增，查，改，删。）具体案例可见之前转的推送。

2. Advantages and disadvantages of using SQL trigger and Stored Procedure?

Advantage and disadvantage of SQL trigger

触发器的优点：以事件方式来处理。当数据发生变化的时候，自动作处理。

缺点：当数据库之间导出导入的时候，可能会引起不必要的触发逻辑。（INSERT INTO没有触发器的话，马上就执行完毕了。有触发器的，就要等待触发器执行完毕后，才算是整体执行完毕。）

存储速度和执行速度：代码是存储在服务器上，执行速度主要取决于数据库服务器的性能与触发器代码的复杂程度。

Stored procedure advantage and disadvantage

存储过程（Stored Procedure）是在大型数据库系统中，一组为了完成特定功能的 SQL 语句集，存储在数据库中，经过第一次编译后再次调用不需要再次编译，用户通过指定存储过程的名字并给出参数（如果该存储过程带有参数）来执行它。存储过程是数据库中的一个重要对象，任何一个设计良好的数据库应用程序都应该用到存储过程。

```

1 CREATE PROCEDURE order_tot_amt
2 @o_id int,
3 @p_tot int output
4 AS
5 SELECT @p_tot = sum(Unitprice*Quantity)
6 FROM orderdetails
7 WHERE orderid=@o_id
8 GO

```

P2-1: Stored procedure code

存储过程是 SQL 语句和可选控制流语句的预编译集合，以一个名称存储并作为一个单元处理。存储过程存储在数据库内，可由应用程序通过一个调用执行，而且允许用户声明变量、有条件执行以及其它强大的编程功能。存储过程在创建时即在服务器上进行编译，所以执行起来比单个 SQL 语句快。

存储过程的优点：

- 1.存储过程只在创建时进行编译，以后每次执行存储过程都不需再重新编译，而一般 SQL 语句每执行一次就编译一次,所以使用存储过程可提高数据库执行速度。
- 2.当对数据库进行复杂操作时(如对多个表进行 Update,Insert,Query,Delete 时)，可将此复杂操作作用存储过程封装起来与数据库提供的事务处理结合一起使用。
- 3.存储过程可以重复使用,可减少数据库开发人员的工作量
- 4.安全性高,可设定只有某此用户才具有对指定存储过程的使用权
5. 减少客户端与服务器端的通讯。业务逻辑的封装，以后的修改比较方便

缺点：

- 1.如果更改范围大到需要对输入存储过程的参数进行更改，或者要更改由其返回的数据，则您仍需要更新程序集中的代码以添加参数、更新 GetValue() 调用，等等，这时候估计比较繁琐了。
- 2.可移植性差：由于存储过程将应用程序绑定到 SQL Server，因此使用存储过程封装业务逻辑将限制应用程序的可移植性。
3. 比单纯用 SQL 语句占用更多的系统资源。

3. What is advantage of View?

第一点：使用视图，可以定制用户数据，聚焦特定的数据。

在实际过程中，公司有不同角色的工作人员，我们以销售公司为例的话，采购人员，可以需要一些与其有关的数据，而与他无关的数据，对他没有任何意义，我们可以根据这一实际情况，专门为采购人员创建一个视图，以后他在查询数据时，只需 `select * from view_caigou` 就可以啦。

第二点：使用视图，可以简化数据操作。

我们在使用查询时，在很多时候我们要使用聚合函数，同时还要显示其它字段的信息，可能还会需要关联到其它表，这时写的语句可能会很长，如果这个动作频繁发生的话，我们可以创建视图，这以后，我们只需要 `select * from view1` 就可以啦～，是不是很方便呀～

第三点：使用视图，基表中的数据就有了一定的安全性

因为视图是虚拟的，物理上是不存在的，只是存储了数据的集合，我们可以将基表中重要的字段信息，可以不通过视图给用户，视图是动态的数据的集合，数据是随着基表的更新而更新。同时，用户对视图，不可以随意的更改和删除，可以保证数据的安全性。

第四点：可以合并分离的数据，创建分区视图

随着社会的发展，公司的业务量的不断的扩大，一个大公司，下属都设有很多的分公司，为了管理方便，我们需要统一表的结构，定期查看各公司业务情况，而分别看各个公司的数据很不方便，没有很好的可比性，如果将这些数据合并为一个表格里，就方便多啦，这时我们就可以使用 `union` 关键字，将各分公司的数据合并为一个视图。

4. DBMS, CRM, ERP (multiuser databases application)

DMBS:数据库管理系统(Database Management System)是一种操纵和管理数据库的大型软件，用于建立、使用和维护数据库，简称 DBMS。它对数据库进行统一的管理和控制，以保证数据库的安全性和完整性。

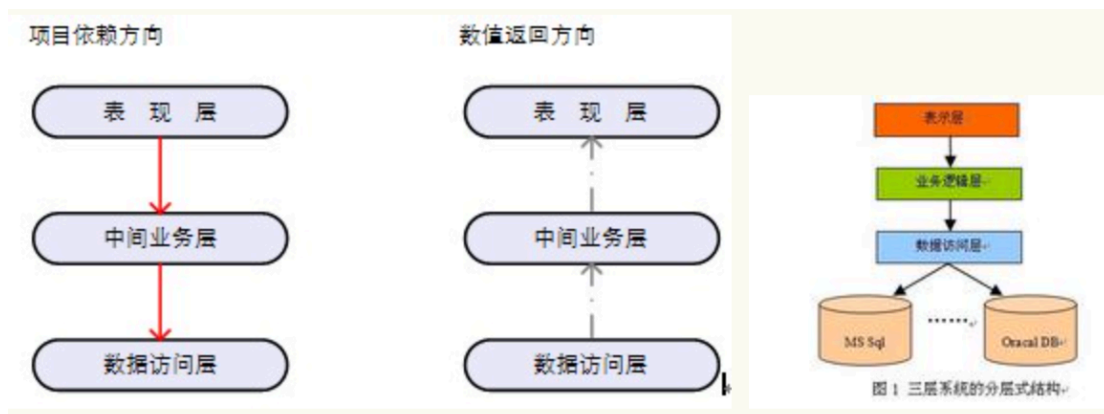
CRM:客户关系管理（Customer relationship management）是利用信息科学技术，实现市场营

销、销售、服务等活动自动化，使企业能更高效地为客户提供满意、周到的服务，以提高客户满意度、忠诚度为目的的一种管理经营方式。客户关系管理既是一种管理理念，又是一种软件技术。以客户为中心的管理理念是 CRM 实施的基础。

ERP: ERP 系统是企业资源计划(Enterprise Resource Planning)的简称，是指建立在信息技术基础上，以系统化的管理思想，为企业决策层及员工提供决策运行手段的管理平台。

5. Three layer architecture

“三层架构”中的三层是指：表现层（UI）、业务逻辑层(BLL)、数据访问层(DAL)



表现层: 通俗讲就是展现给用户的界面，即用户在使用一个系统时候他的所见所得。

位于最外层（最上层），离用户最近。用于显示数据和接收用户输入的数据，为用户提供一种交互式操作的界面。

业务逻辑层: 针对具体问题的操作，也可以说是对数据层的操作，对数据业务逻辑处理。

负责处理用户输入的信息，或者是将这些信息发送给数据访问层进行保存，或者是调用数据访问层中的函数再次读出这些数据。中间业务层也可以包括一些对“商业逻辑”描述代码在里面

数据访问层: 该层所做事务直接操作数据库，正对数据的增添、删除、修改、更新、查找。

仅实现对数据的保存和读取操作。数据访问，可以访问数据库系统、二进制文件、文本文档或是 XML 文档

6. PO persistent object (durability)

PO，持久对象(Persistent Object)，它们是由一组属性和属性的 get 和 set 方法组成。

在 o/r 映射的时候出现的概念，如果没有 o/r 映射，就没有这个概念存在了。通常对应数据模型(数据库),本身还有部分业务逻辑的处理。可以看成是与数据库中的表相映射的 java 对象。最简单的 PO 就是对应数据库中某个表中的一条记录，多个记录可以用 PO 的集合。PO 中应该不包含任何对数据库的操作。

PO 的属性是跟数据库表的字段一一对应的。PO 对象需要实现序列化接口。

7. Entity Relation ship

8. Functional Dependency

9. ER diagram

10. SQL statement

11. Mysql_fetch_object / row / assoc

（他明明说不重点考察 php 的.....好打脸）

php 从 mysql 中访问数据库并取得数据，取得结果的过程中用到好几个类似的方法。

1.mysql_fetch_row，这个函数是从结果集中取一行作为枚举数据，从和指定的结果标识关联的结果集中取得一行数据并作为数组返回。每个结果的列储存在一个数组的单元中，偏移量从 0 开始。注意，这里是从0开始偏移，也就是说不能用字段名字来取值，只能用索引来取值，所以如下代码是取不到值的：

2.mysql_fetch_array，从结果集中取得一行作为关联数组，或数字数组，或二者兼有，除了将数据以数字索引方式储存在数组中之外，还可以将数据作为关联索引储存，用字段名作为键名。也就是说他得到的结果像数组一样，可以用key或者索引来取值，所以

3.mysql_fetch_object,顾名思义，从结果集中取得一行作为对象，并将字段名字做为属性。所以只有这样才能取到值

4.mysql_fetch_assoc，从结果集中取得一行作为关联数组，也就是说这个函数不能像mysql_fetch_row那样用索引来取值，只能用字段名字来取，

比如：mysql_fetch_assoc(\$shihang_result)==mysql_fetch_array(\$result_shihang,ASSOC);

12. Table

13. CSS -3 ways（串讲提过 4 种，和 12 题一样给大家画过）

注：1,4,7,8,9,12,13 题目在手写笔记里面都有过书写和串讲重点解释，10 题请见附件（常用 sql 语句以及关键词 sh.doc）