启明星辰人力资源系统

数据库设计说明书

版本：X.X

|  |  |  |  |
| --- | --- | --- | --- |
| **作者：** |  | **日期：** |  |
| **审批：** |  | **日期：** | 2016-09-14 |

目录

[一、数据库设计概述 2](#_Toc461619810)

[二、数据库使用准备工作 2](#_Toc461619811)

[2.1、数据库设计（字段设计参考设计规定） 2](#_Toc461619812)

[2.2、数据库表数据的添加 2](#_Toc461619813)

[三、设计规定 2](#_Toc461619814)

[3.1、字段设计的规范 2](#_Toc461619815)

[3.2、数据库设计软件PowerDesigner的使用 4](#_Toc461619816)

[3.3、数据库客户端软件 5](#_Toc461619817)

[四、数据库开发使用规定 5](#_Toc461619818)

[4.1、数据库的数据使用 5](#_Toc461619819)

[4.2、数据的相关功能使用 6](#_Toc461619820)

# 一、数据库设计概述

在数据库的设计过程当中，为了编码的规范性，相应的应该约定一些规定，以此来防范出现数据库设计字段不统一，项目的开发人员不明白数据库字段所代表的意思，以及其他可能出现的问题。所以，制定了本约定来约束数据库设计规范。(本规范默认使用Mysql数据库，数据库设计软件采用Mysql-Font)

# 二、数据库使用准备工作

## 2.1、数据库设计（字段设计参考设计规定）

2.1.1、在拿到需求设计文档的时候，组长要对照着数据库设计文档进行完善数据库表的设计。在确保当前可以要使用的字段都有的情况下，再进行保留2、3个字段，以防后期需要修改数据库表。

2.1.2、在分配到每个组员的任务的情况下，组员应该将自己的所负责的那部分的数据库表进行检查，确保里面的字段以及外键关系等都是完善的，如果有缺陷，应该及时向项目经理或者组长反馈，及时改正。

2.1.3、由每组的组员将相关的数据文档整理发送给组长或者项目经理进行整合，并且由项目经理或组长整合为一个统一的数据库设计文档。

## 2.2、数据库表数据的添加

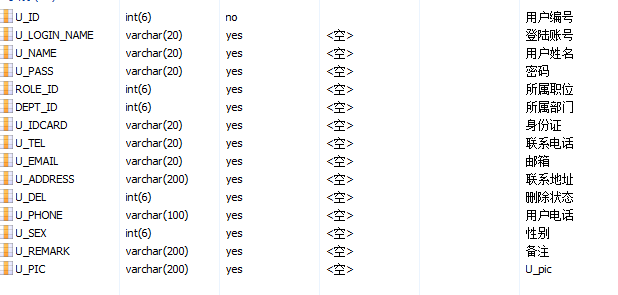
2.2.1、在完成上面的2.1的数据设计的工作的情况下，由每组的组员将各自的数据库表数据进行添加。数据量视情况而定。

2.2.2、在使用数据库之前，应该首先确认数据库的编码格式为“utf-8”。

# 三、设计规定

## 3.1、字段设计的规范

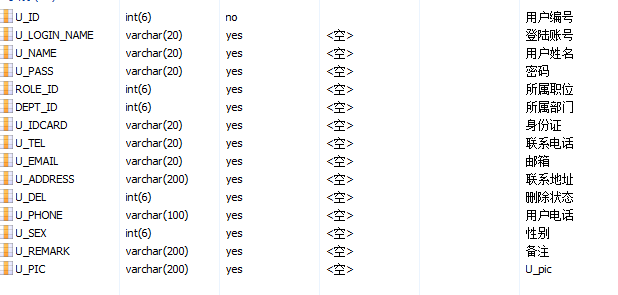
通过以下表为例（数据库的名称是不分大小写的，写小写也是大写）：



3.1.1、当进行数据库表字段命名的时候，应该达到见名知意的效果

例子：Users表中的邮箱字段，应该采用“email”或者“u\_email”的形式，并且应该给该字段加注释。而不是使用“abc”等不知道意思的命名方式。建议使用英文翻译软件。注释如下图可以查看：

在下面每个字段的后面就是当前字段的相应注释。



3.1.2、使用“\_”来区别两个单词，而不是使用驼峰式的命名规范。

例子：Users表中的登录帐号，应该使用login\_name,而不是使用loginName,或者也可以在前面加上表名的首字母：u\_login\_name。但每组最好使用统一的格式。

3.1.3、应该给字段注释，每个字段都应该给注释

例子：Users表中的密码，应该使用password，并且注释中应该写入“密码”。

3.1.4、用户的设计当中有性别字段时，规定如下表

|  |  |  |
| --- | --- | --- |
| 命名意义 | 女 | 男 |
| 字段设计（规定） | gender | gender |
| 字段值（number） | 0 | 1 |

3.1.5、当使用禁用启用状态的字段，规定如下表

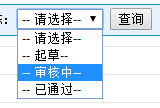
|  |  |  |
| --- | --- | --- |
| 命名意义 | 启用 | 禁用 |
| 字段设计（规定） | is\_forbidden | is\_forbidden |
| 字段值（number） | 0 | 1 |

2.1.6、当使用逻辑删除的字段时，规定如下表

|  |  |  |
| --- | --- | --- |
| 命名意义 | 未删除 | 已删除 |
| 字段设计（规定） | is\_delete | is\_delete |
| 字段值（number） | 0 | 1 |

3.1.7、当使用下拉列表框，有多个状态的时候，我们统一规定下拉列表框中的未选择时的字段（number）的值为“0”;

如下图所示：



使用“--请选择--”查询数据库的时候，应该传到java当中的数据应该是整数类型的“0”,其他的状态可以任意给定。

## 3.2、数据库设计软件PowerDesigner的使用

3.2.1、数据库设计时应该使用PowerDesinger软件进行数据库的设计，在使用PowerDesinger软件的时候，如果有外键约束的关系的时候，应该在.pdm文件当中体现出来。如下图所示：

3.2.2、数据库设计时如果对应的字段有相关的约束关系，应该将约束关系添加上，虽然在设计上没有相应的体现，但是应该有实际操作。

3.2.3、在数据库设计的时候，应该按规定尽可能的满足三大范式。

1、第一范式：   
    1）、每个表都要有主键   
    2）、列不可分，比如：   
    create table stu   
    ( id int(8),   
 name varchar2(20),  
      age int(3)   
       )   
    insert into stu values(1,'yu',21);   
    create table stu2   
    ( id varchar   
    )   
    insert into stu2 values('1\_yu\_21');   
    3）、虽然表stu2可以字符串解析后可以达到和表stu一样的效果，但是显然第一种方法更好，查询数据更加方便，而表stu2违反了第一范式的列不可分原则。   
   2、第二范式：   
    1）、当有两个以上字段作主键时，非主键字段不能部分依赖于主键字段，如有一个需求，一门老师教多名学生，一名学生可以选多个老师的课。然后设计了一张表，其中以老师编号和学生编号作联合主键，则些表就存在部分依赖，老师姓名部分依赖于老师编号，不满足第二范式，有数据冗余。要解决这个问题可以设计三张表。

|  |  |  |  |
| --- | --- | --- | --- |
| 老师编号 | 学生编号 | 老师姓名 | 学生姓名 |
| 1 | 1 | 老一 | 学一 |
| 1 | 2 | 老一 | 学二 |
| 2 | 1 | 老二 | 学一 |
| 2 | 2 | 老二 | 学二 |

3、 第三范式：   
    1）、不能存在传递依赖，如有一张表有以下字段：（学号、姓名、班级号、班级名称、班级位置），其中学号为主键，则班级号依赖于学号，每个学生都有相应的班级号，但是班级名称、班级位置是依赖于班级号，即它们通过班级号传递依赖于学号，不满足第三范式。

## 3.3、数据库客户端软件

3.3.1、在使用数据库的情况下，应该将有关系的表建立相应的外键关系，达到约束。

3.3.2、在使用的时候有要求使用存储过程、序列、索引等相关功能的时候，要建立这些关系或者功能，并且要将相关的sql语句进行保存。

3.3.3、在创建数据库和数据表的时候，要将相关的sql语句都保存起来。要存储相应的文档。

# 四、数据库开发使用规定

## 4.1、数据库的数据使用

4.1.1、开发过程中应该使用统一的数据库，或者在开发的时候，应该进行字段的统一，编码格式也应该要统一，建议使用编码格式为“utf-8”。注意：并且项目也设置为“utf-8”的格式。

4.1.2、在进行开发，发现需要修改字段名称、字段类型以及字段大小等操作的情况下，应该向项目经理或者组长进行报告，通过讨论、考核以后才可以由相关人员来修改数据库，并在需求的情况下进行数据的添加。

4.1.3、操作数据库的时候，确保只添加、不删除的原则、修改的也要尽可能的少修改。

## 4.2、数据的相关功能使用

4.2.1、开发过程中，需要进行序列、存储过程、索引等修改、创建的时候，应该要进行审核，通过审核后，可以由相关的开发人员或者项目经理（组长）进行修改和创建。