**数据库系统原理作业 #1**

**1、（1）描述：**数据是数据库存储的基本对象。 模型是对现实世界中某个对象特征的模拟和抽象，数据模型是众多模型的一种，数据模型是数据库系统的核心和基础，数据模型是用来描述数据、组织数据和对数据进行操作的。数据库管理系统是对数据进行组织和存储、获取以及维护的一种软件，它基于操作系统实现对数据库的控制。而数据库、数据库管理系统都是数据库系统的一部分，因此数据库系统也是一种软件。

用VISIO作出语义网络如下所示：



**（2）描述：**实体是客观存在并可以相互区别的事物，属性是实体具有的某一特性，域是属性的特征，实体标识符是可以唯一标识实体的属性之一，与其他属性也有着内部联系。同一类型实体的集合称为实体集，用实体名及其属性名集合来抽象和刻画的同类实体叫作实体型。不同实体集之间有着（1:1,1：n或m：n）的联系。

因此，根据如上描述作出语义网络如下所示：



**2、**（1）12306网站余票查询业务模型

设计思路：用户登录12306账号，进入余票查询系统，用户通过输入出发地、目的地、出发日三个条件，查询可能存在的车次，用户可以看到每个车次经过的站点名称，以及每种座位的余票数量。

由此可得如下的实体型：

用户的12306账户（账号ID、登录密码）；

用户（用户账号ID、姓名、性别、身份证号、籍贯、联系电话、邮箱）；

订票（订单号、账号、密码、始发站、终点站、车次）；

车票（流水号、车次、票价、出发时间、到站时间、始发站、终点站、座位类型、余票数目）；

后台管理员（账号、密码、姓名、性别、所属部门）；

列车（列车编号、车次、始发站、终点站、出发时间、到站时间、使用时间、行车路线）

由此，用E-R图作出简易的概念模型如下：



**图1- 12306余票查询系统的E-R模型图**

用UML表示法作出简化的概念模型如下：



**图2- 12306余票查询系统的UML简易模型图**

**说明：**用户的12306账户是用户信息的一种，因此和用户存在继承关系，其为子类，用户为父类；用户和后台管理员、列车、车票之间均存在着关联关系，用户和管理员之间是管理和被管理的关系，用户和列车之间是乘坐与被乘坐的关系；用户与车票是订购和被订购的关系。于此同时，列车和车票之间也存在继承关系，列车的车次决定了车票的各种信息，列车是父类，车票是子类。

3、答：上述场景涉及非空约束（NK）、唯一约束（UK）、检查约束（CK）、主键约束（PK）和外键约束（FK）。

NK：用户账户ID、列车编号、流水号、身份证号、车次等都不能为空，满足非空约束；

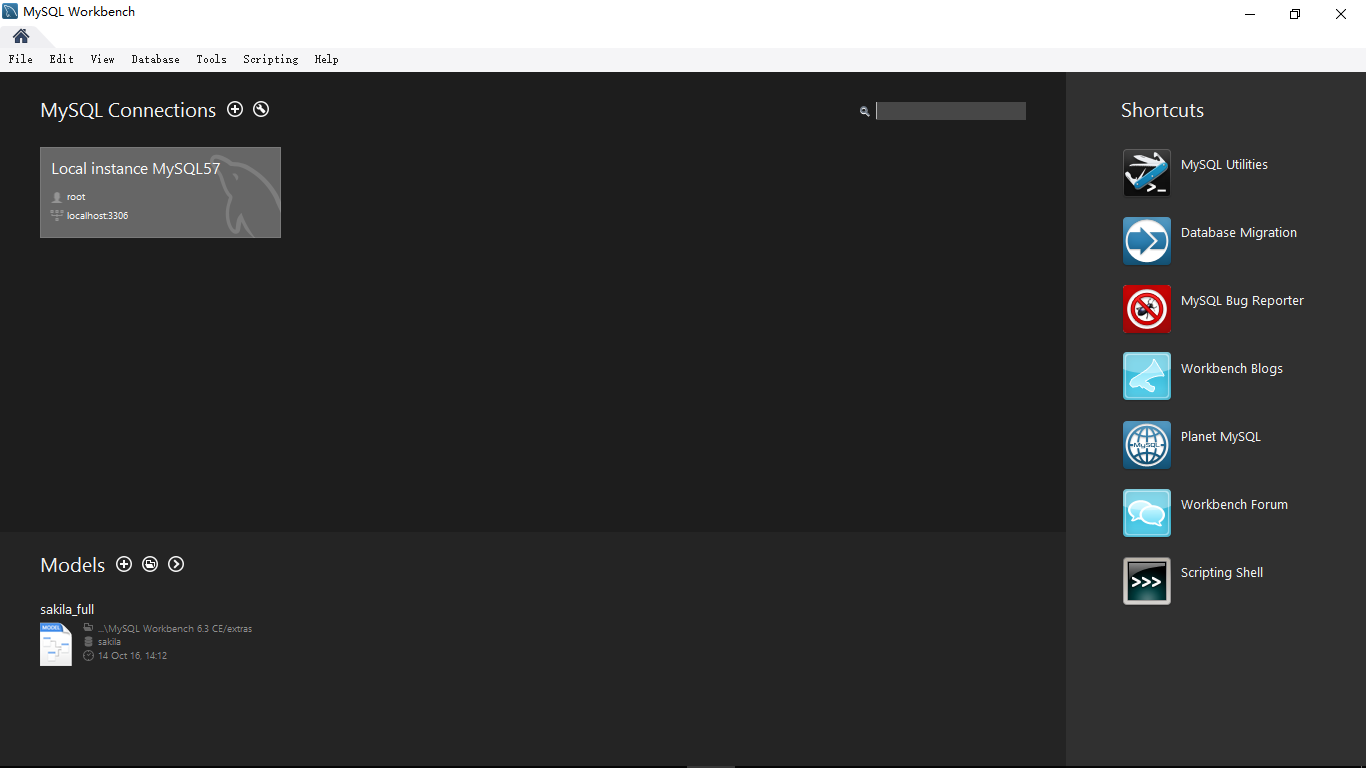
UK：姓名、密码、身份证号、籍贯等对于一个用户来说都是唯一的，需满足唯一约束；

CK：密码是否符合安全规范（例如是否包含数字、大小写字母等）、性别（男/女）、身份证号（是否为18位数字）、座位类型（是软卧/硬卧/软座/硬座/无座）等，都需满足检查约束；

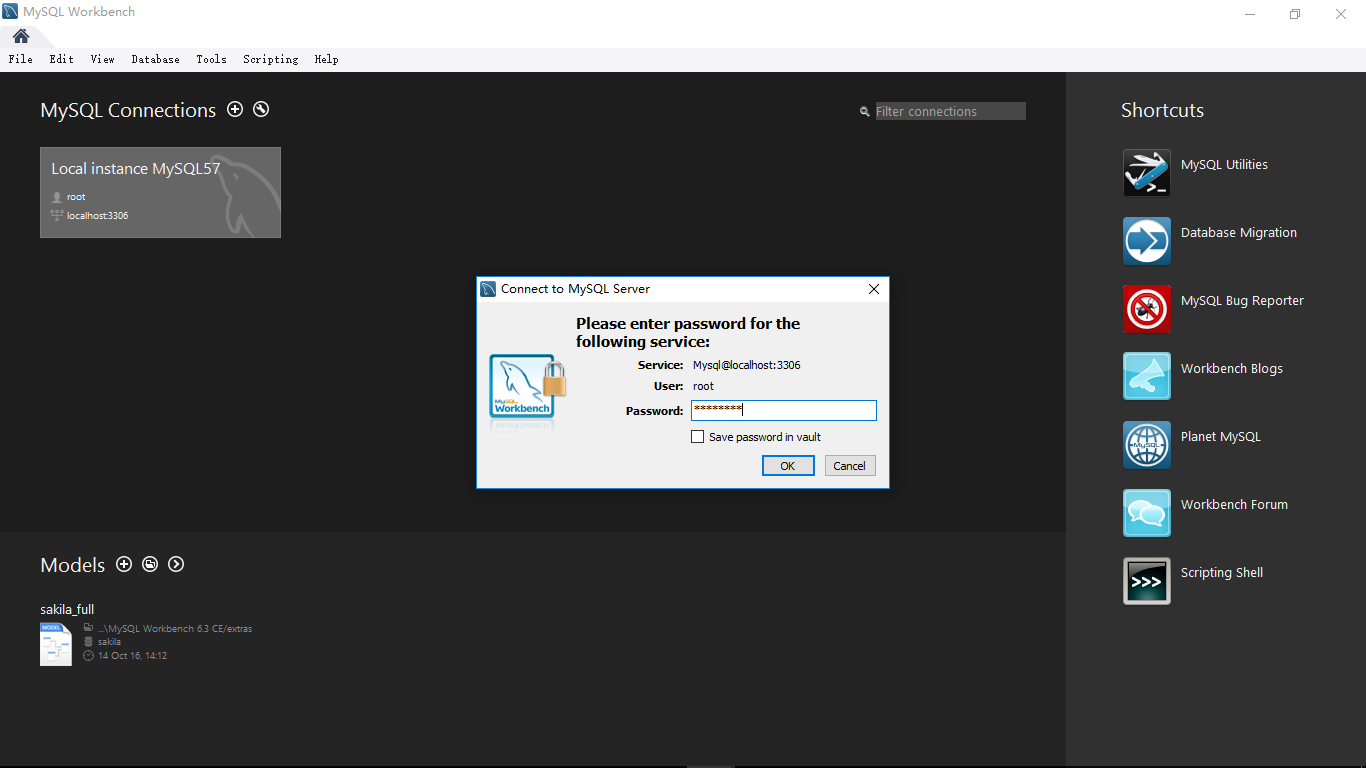
PK：同时满足NK和UK的需要符合主键约束，如：账户ID、列车编号、流水号等；

FK：用户的12306账户和用户之间存在关联关系，12306账户的账号需参考用户的账号；车票和列车之间也存在关联关系，车票的车次需参考列车的车次、车票的始发站需参考列车的始发站、车票的终点站需参考列车的终点站、车票的出发时间需参考列车的出发时间、车票的到达时间需参考列车的到达时间……这些均需满足外键约束。

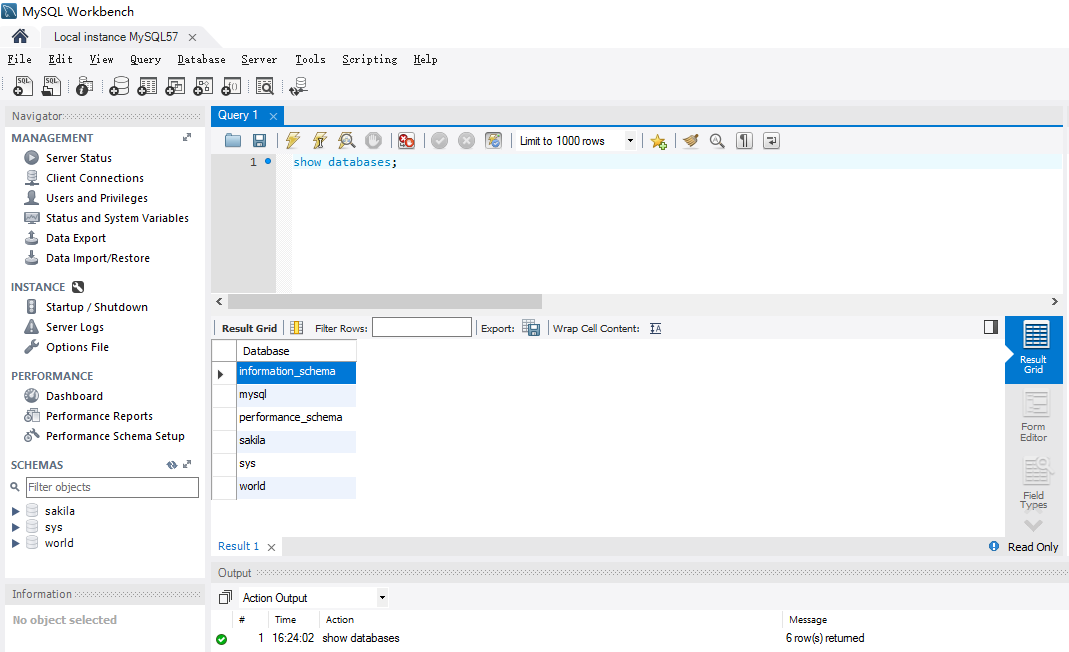
4、（1）mysql 5.7.17安装完之后的截图：



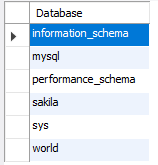
连接数据库：



查看存在的数据库：show databases;



查询结果如下：



上述过程证实数据库已经成功搭建，接下来进入建库和建表的操作。

（2）建立一个 职工-部门-奖金 数据库

建库脚本如下：

--创建部门表

CREATE TABLE dept (

deptno INT(50) PRIMARY KEY,

dname char varying(14) ,

loc char varying(13) ) ;

--创建职工表

CREATE TABLE emp (

empno INT(4) PRIMARY KEY,

ename char varying(10),

job char varying(9),

mgr int8(4),

hiredate DATE,

sal numeric(7,2),

comm numeric(7,2),

deptno int2(50) REFERENCES DEPT );

--创建奖金表

CREATE TABLE bonus (

enamE char varying(10) ,

job char varying(9) ,

sal int8,

comm int8 ) ;

--创建工资等级表

CREATE TABLE salgrade (

grade int,

losal int8,

hisal int8 );

--向dept表中插入数据

INSERT INTO dept VALUES (10,'ACCOUNTING','NEW YORK');

INSERT INTO dept VALUES (20,'RESEARCH','DALLAS');

INSERT INTO dept VALUES (30,'SALES','CHICAGO');

INSERT INTO dept VALUES (40,'OPERATIONS','BOSTON');

--向emp表中插入数据

INSERT INTO emp VALUES (7369,'SMITH','CLERK',7902,'1980-12-17',800,NULL,20);

INSERT INTO emp VALUES (7499,'ALLEN','SALESMAN',7698,'1981-02-20',1600,300,30);

INSERT INTO emp VALUES (7521,'WARD','SALESMAN',7698,'1981-02-22',1250,500,30);

INSERT INTO emp VALUES (7566,'JONES','MANAGER',7839,'1981-04-02',2975,NULL,20);

INSERT INTO emp VALUES (7654,'MARTIN','SALESMAN',7698,'1981-09-28',1250,1400,30);

INSERT INTO emp VALUES (7698,'BLAKE','MANAGER',7839,'1981-05-01',2850,NULL,30);

INSERT INTO emp VALUES (7782,'CLARK','MANAGER',7839,'1981-09-06',2450,NULL,10);

INSERT INTO emp VALUES (7788,'SCOTT','ANALYST',7566,'1987-04-19',3000,NULL,20);

INSERT INTO emp VALUES (7839,'KING','PRESIDENT',NULL,'1981-11-17',5000,NULL,10);

INSERT INTO emp VALUES (7844,'TURNER','SALESMAN',7698,'1981-09-08',1500,0,30);

INSERT INTO emp VALUES (7876,'ADAMS','CLERK',7788,'1987-05-25',1100,NULL,20);

INSERT INTO emp VALUES (7900,'JAMES','CLERK',7698,'1981-03-12',950,NULL,30);

INSERT INTO emp VALUES (7902,'FORD','ANALYST',7566,'1981-03-12',3000,NULL,20);

INSERT INTO emp VALUES (7934,'MILLER','CLERK',7782,'1982-01-23',1300,NULL,10);

--向salgrade表中插入数据

INSERT INTO salgrade VALUES (1,700,1200);

INSERT INTO salgrade VALUES (2,1201,1400);

INSERT INTO salgrade VALUES (3,1401,2000);

INSERT INTO salgrade VALUES (4,2001,3000);

INSERT INTO salgrade VALUES (5,3001,9999);

--事务提交

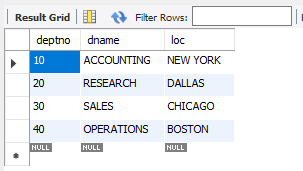
COMMIT;

-----------------------------------------------------------------------------------------------

操作：查询dept表的全部数据

SELECT \* FROM dept;

查询结果如下：



查询结果与建库脚本中的内容一致。