（1）编写函数func\_rand\_ID，可随机生成长度为18位的仿身份证号码，第1-6位为0-9之间的数字，第7-14位为19600101到20200101之间的某个有效日期，第15-17位为0-9之间的数字，第18位为0-9之间的数字或大写X。

DROP FUNCTION IF EXISTS func\_rand\_ID;

CREATE FUNCTION func\_rand\_ID() RETURNS VARCHAR(18)

BEGIN

DECLARE result VARCHAR(18) DEFAULT '';

DECLARE first\_num VARCHAR(6) DEFAULT '';

DECLARE birthday VARCHAR(8) DEFAULT '';

DECLARE second\_num VARCHAR(3) DEFAULT '';

DECLARE last\_num VARCHAR(1) DEFAULT '';

DECLARE charset VARCHAR(11) DEFAULT '';

DECLARE interday INT DEFAULT 0;

DECLARE numset varchar(10) DEFAULT '';

DECLARE i INT DEFAULT 0;

DECLARE j INT DEFAULT 0;

SET numset = "0123456789";

SET charset = "0123456789X";

#处理1-6位置上

WHILE i<6

SET first\_num = CONCAT(first\_num,substring(numset,FLOOR(1+RAND()\*10),1));

SET i = i + 1;

END while;

#SET first\_num = rand()\*100000;

#处理生日

SET interday = floor(RAND()\*21915);

SET birthday = DATE\_ADD("1960-01-01",INTERVAL interday day);

#处理15-17位置上

WHILE j<3

SET second\_num = CONCAT(second\_num,substring(numset,FLOOR(1+RAND()\*10),1));

SET j = j + 1;

END while;

#SET second\_num = floor(RAND()\*100);

#处理最后一位

SET last\_num = substring(charset,FLOOR(1+RAND()\*11),1);

#合并结果

result = CONCAT(result, first\_num, birthday, second\_num, last\_num);

RETURN result;

END;

SELECT func\_rand\_ID();

/\*

第1-6位为0-9之间的数字，

第7-14位为19600101到20200101之间的某个有效日期，

20200101-19600101 60年

第15-17位为0-9之间的数字，

第18位为0-9之间的数字或大写X。

\*/

（2）使用下面的语句新建一张测试表testUser，编写存储过程createTestCases，往testUser表中插入100条测试数据，其中username是随机生成的长度为8的字符串（符号可包括a-z、A-Z、0-9），email由函数func\_rand\_email生成，telephone由函数func\_rand\_telnum生成，ucode由函数func\_rand\_ID生成。

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

-- Table structure for testuser

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

DROP TABLE IF EXISTS `testuser`;

CREATE TABLE `testuser` (

`uid` int(11) NOT NULL AUTO\_INCREMENT,

`username` varchar(20) DEFAULT NULL,

`email` varchar(150) DEFAULT NULL,

`telephone` varchar(20) DEFAULT NULL,

`ucode` varchar(20) DEFAULT NULL,

PRIMARY KEY (`uid`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

DROP PROCEDURE IF EXISTS createTestCases;

CREATE PROCEDURE createTestCases ()

BEGIN

DECLARE i INT DEFAULT 0;

DECLARE id INT DEFAULT 0;

DECLARE name VARCHAR(8) DEFAULT '';

DECLARE mail VARCHAR() DEFAULT '';

DECLARE phone VARCHAR() DEFAULT '';

DECLARE code VARCHAR(18) DEFAULT '';

DECLARE charset VARCHAR(70);

SET charset='abcdefghijklmnopqrseuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789';

WHILE i < 100

SELECT i INTO id;#uid

SELECT func\_rand\_email() into mail;#email

SELECT func\_rand\_telnum() into phone;#telephone

SELECT func\_rand\_ID() into code;#ucode

#生成username

WHILE i<length DO

SET name = concat(name,substring(charset,floor(1+RAND()\*62),1));

SET i=i+1;

END WHILE;

INSERT

END WHILE;

END;

/\*

其中username是随机生成的长度为8的字符串

（符号可包括a-z、A-Z、0-9）

\*/

（3）在employee表中有引用EMP\_ID字段的SUPERIOR\_EMP\_ID字段，代表上级领导编号，行长没有上级领导，因此其SUPERIOR\_EMP\_ID的值为NULL，其余员工的SUPERIOR\_EMP\_ID值均不为NULL。创建存储过程getEmpStructure，在employee表上利用递归CTE实现员工层级的计算，行长层级为1，副行长为2，依此类推，该存储过程输出员工编号（EMP\_ID）、员工姓名（LAST\_NAME和FIRST\_NAME拼接）、层级（命名为LEVEL）、职位名称（TITLE）。

（4）创建存储过程updateCloseDate，该存储过程更新account表中的关闭日期（CLOSE\_DATE），根据产品编号做不同的更新操作，要求使用游标：

a.对产品编号对应类型为存款的账户（即产品编号PRODUCT\_CD对应PRODUCT\_TYPE\_CD为ACCOUNT），如开户日期（OPEN\_DATE）在2015-01-01之前（含）的，设置其关闭日期（CLOSE\_DATE）为开户日期加20年，否则为开户日期加30年；

b.对产品编号对应类型为贷款的账户（即产品编号PRODUCT\_CD对应PRODUCT\_TYPE\_CD为LOAN），如可用余额（AVAIL\_BALANCE）少于100000的（含），设置其关闭日期（CLOSE\_DATE）为开户日期加20年，否则为开户日期加30年；

c.对产品编号对应类型为保险的账户（即产品编号PRODUCT\_CD对应PRODUCT\_TYPE\_CD为INSURANCE），设置其关闭日期（CLOSE\_DATE）为开户日期加15年。

（5）在acc\_transaction上定义触发器t\_newTransaction，当往acc\_transaction中插入一条数据时，依据账户编号（ACCOUNT\_ID）更新account表中对应账户的可用余额（AVAIL\_BALANCE）和最后活跃日期（LAST\_ACTIVITY\_DATE）：

a.如果插入数据的交易类型编码（TXN\_TYPE\_CD）为CD、TT、IC、LI则设置可用余额为当前可用余额加上交易金额、最后活跃日期为当前日期

b.如果插入数据的交易类型编码（TXN\_TYPE\_CD）为CW、TF则设置可用余额为当前可用余额减去交易金额、最后活跃日期为当前日期；如当前可用余额减去交易金额小于0，则撤销对acc\_transaction此条数据的插入，同时输出提示“余额不足Insufficient Balance”。

（6）在officer上定义触发器t\_insertOfficer，当往officer中新插入一条单位联系人信息时候，检查对应的客户编号（CUST\_ID）字段，如该客户编号对应的单位联系人信息已存在，先将原有单位联系人的结束日期（END\_DATE）修改为当前插入日期后再插入；如该客户编号对应的单位联系人信息不存在则直接插入。