

**期末项目设计报告**

|  |  |  |  |
| --- | --- | --- | --- |
| 题 目 | 基于Oracle的家庭医疗信息系统的数据库设计 | | |
| 课程 | Oracle数据库应用 | | |
| 学 院 | 信息科学与工程学院 | | |
| 专 业 | 软件工程 | 年级 | 2018级 |
| 学生姓名 | 库淑贞 | 学号 | 201810414302 |
| 指导教师 | 赵卫东 | 职称 | 副教授 |

|  |  |  |  |
| --- | --- | --- | --- |
| **评分项** | **评分标准** | **满分** | **得分** |
| 文档整体 | 文档内容详实、规范，美观大方 | 10 |  |
| 表设计 | 表，表空间设计合理，数据合理 | 20 |  |
| 用户管理 | 权限及用户分配方案设计正确 | 20 |  |
| PL/SQL设计 | 存储过程和函数设计正确 | 30 |  |
| 备份方案 | 备份方案设计正确 | 20 |  |
| **得分合计** | | |  |

2021 年 6 月 1 日

**项目内容**

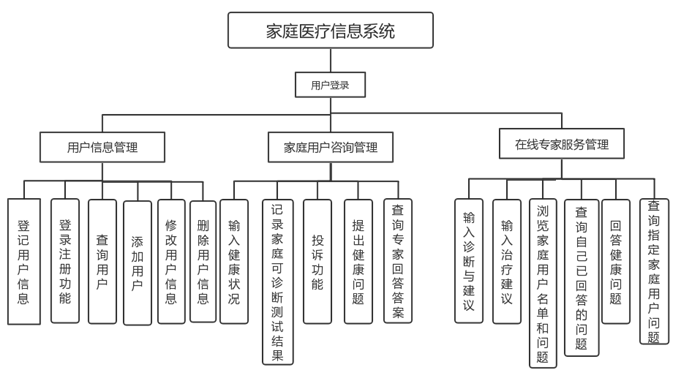
**数据库基本描述**

本系统包括了两种用户类型：家庭用户和在线专家。家庭用户通过注册进入系统登记个人信息，输入自己的健康状态，家庭可收集的诊断测试结果（如有）或向在线专家提问。进行咨询。

在线专家注册登录系统后登记个人信息，可以浏览家庭用户的名单和问题，并使用该系统输入诊断，治疗建议和回答家庭用户提出的问题。

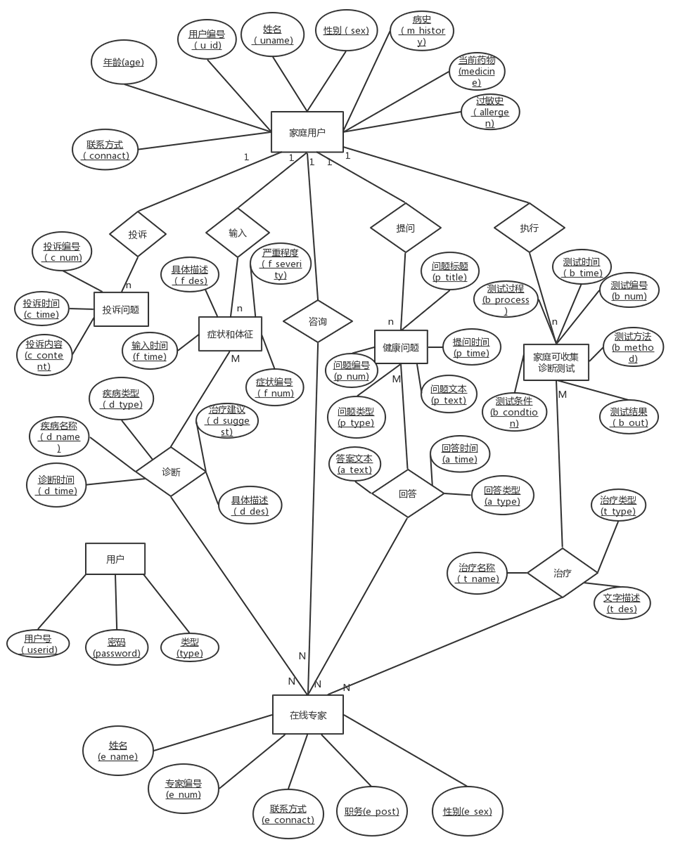
**系统设计**

**系统结构**

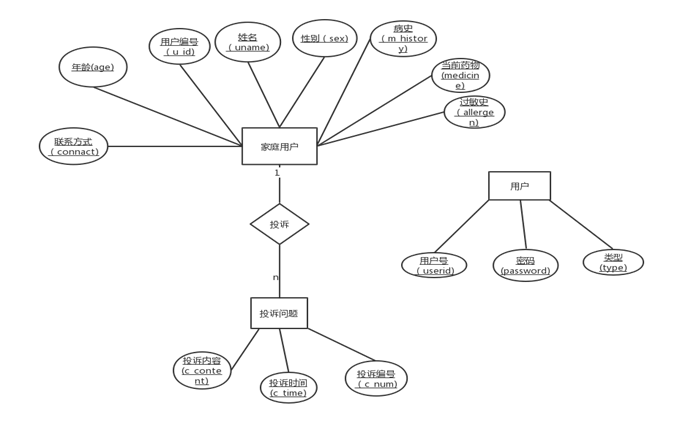


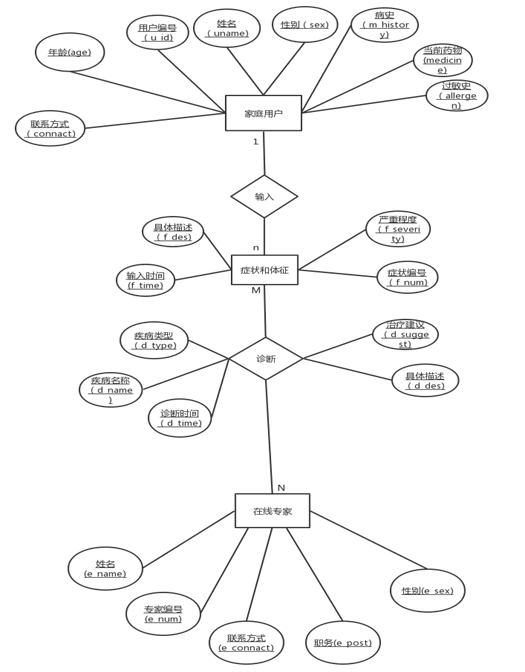
**E-R图**

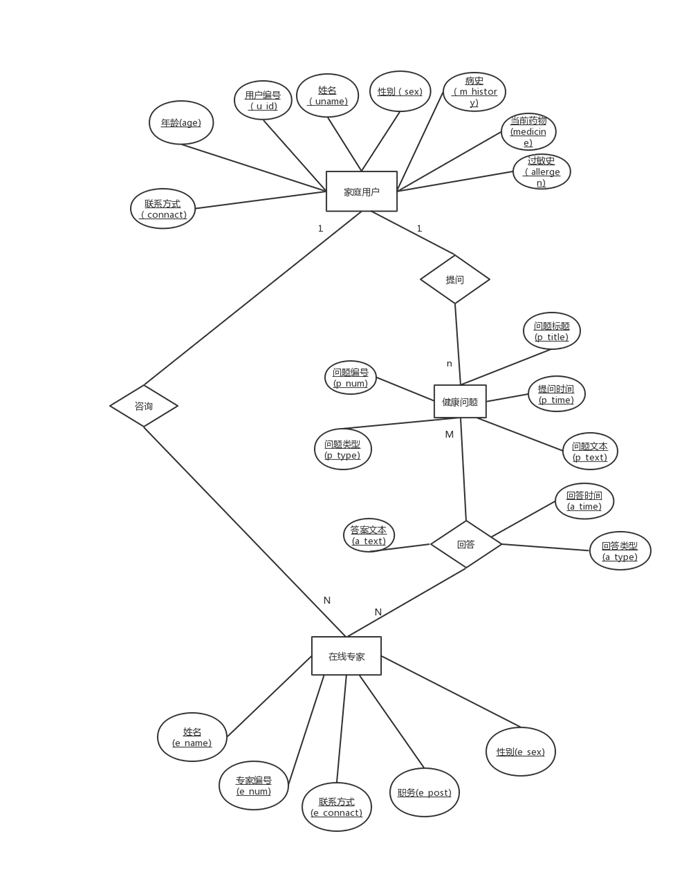
总图：

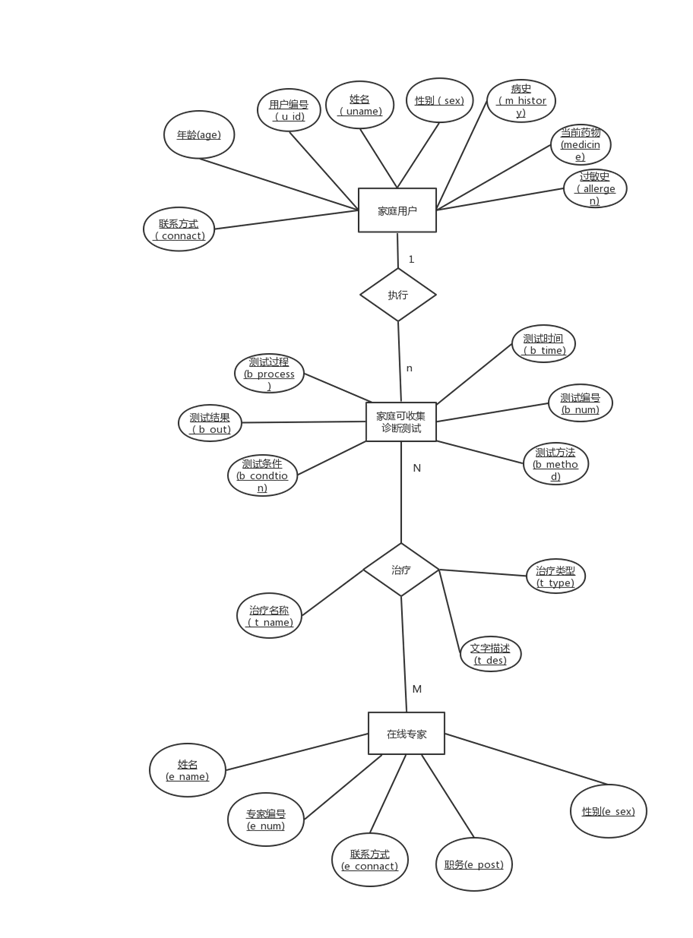


分图:









**表结构**

**关系模式：**

用户表：users(u\_id,uname,sex,number,connact,m\_history,medicine,allergen,e\_num)

专家表：expert(e\_num,e\_sex,e\_connact,e\_post)

投诉表: complaint(c\_num,u\_id,c\_time,c\_content)

症状表:features(f\_num,u\_id,f\_severity,f\_des,f\_time)

诊断表：diagnose(f\_num,e\_num,d\_name,d\_type,d\_time,d\_des,d\_suggest)

问题表:problem(p\_num,u\_id,p\_title,p\_type,p\_time,p\_text)

回答表:answer(p\_num,e\_num,a\_time,a\_text,a\_type)

测试表:beta(b\_num,u\_id,b\_time,b\_process,b\_condtino,b\_method,b\_out)

治疗表:treat(b\_num,e\_num,t\_name,t\_type,t\_des)

注册表:use\_info(useid,password,type)

**物理模型：**

用户表

| **字段名** | **数据类型** | **允许空** | **主/外键** | **检查条件** |
| --- | --- | --- | --- | --- |
| u\_id | numeric | no | 主键 |  |
| uname | varchar |  |  |  |
| sex | char |  |  | ‘男’或’女’ 默认值为’男’ |
| age | number |  |  |  |
| connact | varchar |  |  |  |
| m\_history | varchar |  |  |  |
| medicine | varchar |  |  |  |
| allergen | varchar |  |  |  |
| e\_num | numberic |  | 外键 |  |

专家表

| **字段名** | **数据类型** | **允许空** | **主/外键** | **检查条件** |
| --- | --- | --- | --- | --- |
| e\_num | numeric | no | 主键 |  |
| e\_name | varchar |  |  |  |
| e\_sex | char |  |  | ‘男’或’女’ 默认值为’男’ |
| e\_connact | varchar |  |  |  |
| e\_post | varchar |  |  |  |

投诉表

| **字段名** | **数据类型** | **允许空** | **主/外键** | **检查条件** |
| --- | --- | --- | --- | --- |
| c\_num | numeric | no | 主键 |  |
| u\_id | numeric |  | 外键 |  |
| c\_time | date |  |  |  |
| c\_content | varchar |  |  |  |

症状表

| **字段名** | **数据类型** | **允许空** | **主/外键** | **检查条件** |
| --- | --- | --- | --- | --- |
| f\_num | varchar | no | 主键 |  |
| u\_id | numeric |  | 外键 |  |
| f\_severity | char |  |  | ‘轻度’,’中度’,’严重’ |
| f\_des | varchar |  |  |  |
| f\_time | date |  |  |  |

诊断表

| **字段名** | **数据类型** | **允许空** | **主/外键** | **检查条件** |
| --- | --- | --- | --- | --- |
| f\_num | varchar |  | 外键 |  |
| e\_num | numeric |  | 外键 |  |
| d\_name | varchar |  |  |  |
| d\_type | varchar |  |  |  |
| d\_time | date |  |  |  |
| d\_des | varchar |  |  |  |
| d\_suggest | varchar |  |  |  |
| f\_num,e\_num |  |  | 主键 |  |

问题表

| **字段名** | **数据类型** | **允许空** | **主/外键** | **检查条件** |
| --- | --- | --- | --- | --- |
| p\_num | varchar | no | 主键 |  |
| u\_id | numeric |  | 外键 |  |
| p\_title | varchar |  |  |  |
| p\_type | char |  |  | ‘公开’或’私人’ 默认值为’公开’ |
| p\_time | date |  |  |  |
| p\_text | varchar |  |  |  |

回答表

| **字段名** | **数据类型** | **允许空** | **主/外键** | **检查条件** |
| --- | --- | --- | --- | --- |
| p\_num | varchar |  | 外键 |  |
| e\_num | numeric |  | 外键 |  |
| a\_time | Date |  |  |  |
| a\_text | varchar |  |  |  |
| a\_type | char |  |  | ‘公开’或’私人’ 默认值为’公开’ |
| p\_num,e\_num |  |  | 主键 |  |

测试表

| **字段名** | **数据类型** | **允许空** | **主/外键** | **检查条件** |
| --- | --- | --- | --- | --- |
| b\_num | numeric | no | 主键 |  |
| u\_id | numeric |  | 外键 |  |
| b\_time | date |  |  |  |
| b\_process | varchar |  |  |  |
| b\_condtion | varchar |  |  |  |
| b\_method | varchar |  |  |  |
| b\_out | char |  |  | ‘阳性’或’阴性’ 默认值为’阳性’ |

治疗表

| **字段名** | **数据类型** | **允许空** | **主/外键** | **检查条件** |
| --- | --- | --- | --- | --- |
| b\_num | numeric |  | 外键 |  |
| e\_num | numeric |  | 外键 |  |
| t\_name | varchar |  |  |  |
| t\_type | varchar |  |  |  |
| t\_des | varchar |  |  |  |
| b\_num,e\_num |  |  | 主键 |  |

注册表

| **字段名** | **数据类型** | **允许空** | **主/外键** | **检查条件** |
| --- | --- | --- | --- | --- |
| useid | numeric | no | 主键 |  |
| password | varchar | no |  |  |
| type | varchar |  |  | ‘专家’或‘用户’ 默认值为‘用户’ |

**功能实现**

**数据操作**

全部的数据操作都采用存储过程来进行。

**插入数据**

总共有10张基本表，创建了10个存储过程对每个基本表插入数据。

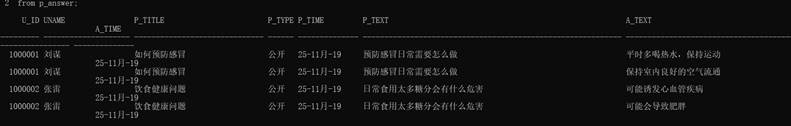
[saveSql.sql](file:///D:\Oracle\oracle\test6\saveSql.sql)

**视图**

1. 创建了从users，problem，answer表产生的一个视图p\_answer，它包括用户号，用户姓名，问题类型，问题标题，问题内容，提问时间，回答时间，回答内容。

create view p\_ answer  
as select users.u\_id, uname, p\_title, p\_type, p\_time, p\_text, a\_text, a\_time  
from users, problem, answer  
where users. u\_id=problem.u\_id and problem.p\_num=answe.p\_num;

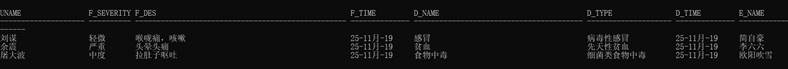
结果显示



1. 创建了从users，features，diagnose，expert表中产生一个视图f\_diagnose,他包括用户姓名，症状严重程度，症状描述，症状输入时间，疾病名称，疾病类型，诊断时间，专家姓名。

create view f\_diagnose  
as select users.uname, f\_severity, f\_des, f\_time, d\_name, d\_type, d\_time, e\_name  
from users, features, diagnose, expert  
where users.u\_id=features.u\_id and features.f\_num=diagnose.f\_num and diagnose.e\_num=expert.e\_num;

结果显示

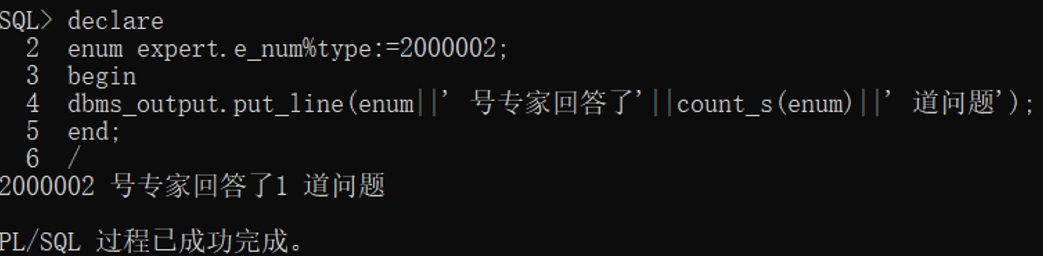


**函数**

1. 创建了一个自定义函数方便的统计对应专家编号回答了多少道问题

create or replace function count\_s(enu expert.e\_num%type)  
return number is  
rowcount number;  
begin  
select count (e\_num)  
into rowcount  
from answer  
where e\_num=enu;  
return rowcount;  
end;  
/

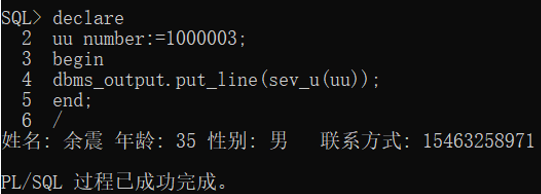
结果显示：



1. 创建了一个自定义函数通过用户编号搜索用户的个人信息

create or replace function sev\_u(uid users.u\_id%type)  
return varchar2 is  
usss users%rowtype;  
begin  
select \* into usss  
from users  
where u\_id=uid;  
return ('姓名:'||usss. uname||'年龄:' ||'usss. age'||'性别: '||usss. sex||'联系方式: '||usss. connact);  
end;  
/

结果显示：



**存储过程**

1.创建更新指定问题标号和专家编号的回答内容的存储过程

create or replace procedure update\_answer(u\_p\_num answer.p\_num%type,u\_e\_num answer.e\_num%type,u\_a\_text answe.a\_text%type)  
begin  
update answer  
set a\_text=u\_a\_text   
where answer.p\_num=u\_p\_num and answer.e\_num=u\_e\_num;  
end;  
/

2.创建查询姓李的用户的姓名的存储过程

create or replace procedure se\_users  
is  
v\_name users.uname%type;  
begin  
select uname into v\_name   
from users  
where uname like '李%';  
end;  
/

**备份方案(在SQL Server中备份)**

* bat命令备份

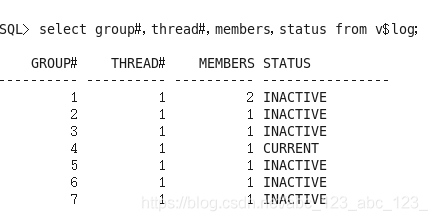
@echo off  
set path=%path%;[osql.exe]  
echo 数据库备份开始  
osql.exe -S 202.115.82.8 -U 'fourteen' -P '123' -i sqlserverbackup.sql -o 'D:\Sql Developer\sql\out.bak'  
echo 数据库备份完成  
Pause

* 创建sqlserverbackup.sql文件

DECLARE @name varchar(50)  
DECLARE @datetime char(14)  
DECLARE @path varchar(255)  
DECLARE @bakfile varchar(255)  
set @name='out'  
set @datetime=CONVERT(char(8),getdate(),112) + REPLACE(CONVERT(char(8),getdate(),108),':','')  
set @path='D:\Sql Developer\sql'  
set @bakfile=@path+''+@name+'.BAK'  
backup database @name to disk=@bakfile with name=@name  
Go

**容灾方案**

* 主库查询确认组数



* 主库增加standbylogfile：

alter database add standby logfile group 8 '/home/oracle/app/oracle/oradata/orcl/stdredo1.log' size 50m;

alter database add standby logfile group 9 '/home/oracle/app/oracle/oradata/orcl/stdredo2.log' size 50m;

alter database add standby logfile group 10 '/home/oracle/app/oracle/oradata/orcl/stdredo3.log' size 50m;

alter database add standby logfile group 11 '/home/oracle/app/oracle/oradata/orcl/stdredo4.log' size 50m;

* 主库环境开启强制归档并修改参数

ALTER DATABASE FORCE LOGGING;

alter system set LOG\_ARCHIVE\_CONFIG='DG\_CONFIG=(orcl,stdorcl)' scope=both sid='\*';

alter system set log\_archive\_dest\_1='LOCATION=/home/oracle/arch VALID\_FOR=(ALL\_LOGFILES,ALL\_ROLES) DB\_UNIQUE\_NAME=orcl' scope=spfile;

alter system set LOG\_ARCHIVE\_DEST\_2='SERVICE=stdorcl LGWR ASYNC VALID\_FOR=(ONLINE\_LOGFILES,PRIMARY\_ROLE) DB\_UNIQUE\_NAME=stdorcl' scope=both sid='\*';

alter system set fal\_client='orcl' scope=both sid='\*';

alter system set FAL\_SERVER='stdorcl' scope=both sid='\*';

alter system set standby\_file\_management=AUTO scope=both sid='\*';

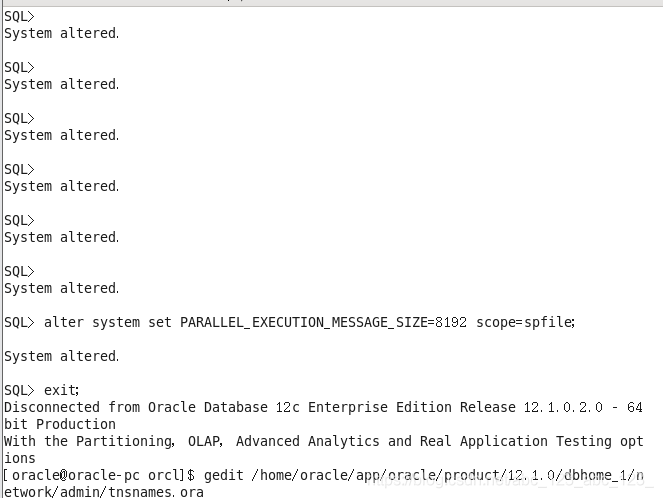
alter system set DB\_FILE\_NAME\_CONVERT='/home/oracle/app/oracle/oradata/stdorcl/','/home/oracle/app/oracle/oradata/orcl/' scope=spfile sid='\*';

alter system set LOG\_FILE\_NAME\_CONVERT='/home/oracle/app/oracle/oradata/stdorcl/','/home/oracle/app/oracle/oradata/orcl/' scope=spfile sid='\*';

alter system set log\_archive\_format='%t\_%s\_%r.arc' scope=spfile sid='\*';

alter system set remote\_login\_passwordfile='EXCLUSIVE' scope=spfile;

alter system set PARALLEL\_EXECUTION\_MESSAGE\_SIZE=8192 scope=spfile;



* 在备库oracle用户创建归档目录，数据目录并设置权限

mkdir -p /u01/app/oracle/diag/orcl

mkdir -p /u01/app/oracle/oradata/stdorcl/

mkdir -p /u01/arch

mkdir -p /u01/rman

mkdir -p /u01/app/oracle/oradata/stdorcl/pdbseed/

mkdir -p /u01/app/oracle/oradata/stdorcl/pdb/

* 备库下执行

/home/oracle/app/oracle/product/12.1.0/dbhome\_1/network/admin/tnsnames.ora 加入配置：

$sqlplus / as sysdba

shutdown immediate;

startup mount exclusive restrict;

drop database;

startup nomount

* 修改主库及备库下文件

ORCL =

(DESCRIPTION =

(ADDRESS\_LIST =

(ADDRESS = (PROTOCOL = TCP)(HOST = 192.168.1.104)(PORT = 1521)) //\*\*

)

(CONNECT\_DATA =

(SERVER = DEDICATED)

(SERVICE\_NAME = orcl)

)

)

stdorcl =

(DESCRIPTION =

(ADDRESS = (PROTOCOL = TCP)(HOST = 192.168.1.103)(PORT = 1521)) //\*\*

(CONNECT\_DATA =

(SERVER = DEDICATED)

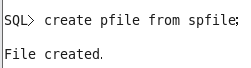
(SID = orcl)

)

)

* 在主库上生成备库的参数文件

create pfile from spfile;



* 将主库的参数文件，密码文件拷贝到备库

scp /home/oracle/app/oracle/product/12.1.0/dbhome\_1/dbs/initorcl.ora 192.168.1.103:/home/oracle/app/oracle/product/12.1.0/dbhome\_1/dbs/

scp /home/oracle/app/oracle/product/12.1.0/dbhome\_1/dbs/orapworcl 192.168.1.103:/home/oracle/app/oracle/product/12.1.0/dbhome\_1/dbs/



* 在备库增加静态监听

gedit /home/oracle/app/oracle/product/12.1.0/dbhome\_1/network/admin/listener.ora

SID\_LIST\_LISTENER =

(SID\_LIST =

(SID\_DESC =

(ORACLE\_HOME = /home/oracle/app/oracle/product/12.1.0/db\_1)

(SID\_NAME = orcl)

)

)

* 将主库复制到备库

rman target sys/123@orcl auxiliary sys/123@stdorcl

run{

allocate channel c1 type disk;

allocate channel c2 type disk;

allocate channel c3 type disk;

allocate AUXILIARY channel c4 type disk;

allocate AUXILIARY channel c5 type disk;

allocate AUXILIARY channel c6 type disk;

DUPLICATE TARGET DATABASE

FOR STANDBY

FROM ACTIVE DATABASE

DORECOVER

NOFILENAMECHECK;

release channel c1;

release channel c2;

release channel c3;

release channel c4;

release channel c5;

release channel c6;

}

输出结果省略

* 在备库上更改参数文件

gedit /home/oracle/app/oracle/product/12.1.0/dbhome\_1/dbs/initorcl.ora

orcl.\_\_data\_transfer\_cache\_size=0

orcl.\_\_db\_cache\_size=671088640

orcl.\_\_java\_pool\_size=16777216

orcl.\_\_large\_pool\_size=33554432

orcl.\_\_oracle\_base='/home/oracle/app/oracle'#ORACLE\_BASE set from environment

orcl.\_\_pga\_aggregate\_target=536870912

orcl.\_\_sga\_target=1258291200

orcl.\_\_shared\_io\_pool\_size=50331648

orcl.\_\_shared\_pool\_size=301989888

orcl.\_\_streams\_pool\_size=0

\*.\_allow\_resetlogs\_corruption=TRUE

\*.\_catalog\_foreign\_restore=FALSE

\*.audit\_file\_dest='/home/oracle/app/oracle/admin/orcl/adump'

\*.audit\_trail='db'

\*.compatible='12.1.0.2.0'

\*.control\_files='/home/oracle/app/oracle/oradata/orcl/control01.ctl','/home/oracle/app/oracle/fast\_recovery\_area/orcl/control02.ctl','/home/oracle/app/oracle/fast\_recovery\_area/orcl/control03.ctl'

\*.db\_block\_size=8192

\*.db\_domain=''

\*.db\_file\_name\_convert='/home/oracle/app/oracle/oradata/orcl/','/home/oracle/app/oracle/oradata/stdorcl/'

\*.db\_name='orcl'

\*.db\_unique\_name='stdorcl'

\*.db\_recovery\_file\_dest='/home/oracle/app/oracle/fast\_recovery\_area'

\*.db\_recovery\_file\_dest\_size=4823449600

\*.diagnostic\_dest='/home/oracle/app/oracle'

\*.dispatchers='(PROTOCOL=TCP)(dispatchers=1)(pool=on)(ticks=1)(connections=500)(sessions=1000)'

\*.enable\_pluggable\_database=true

\*.fal\_client='stdorcl'

\*.fal\_server='orcl'

\*.inmemory\_max\_populate\_servers=2

\*.inmemory\_size=157286400

\*.local\_listener=''

\*.log\_archive\_config='DG\_CONFIG=(stdorcl,orcl)'

\*.log\_archive\_dest\_1='LOCATION=/home/oracle/arch VALID\_FOR=(ALL\_LOGFILES,ALL\_ROLES) DB\_UNIQUE\_NAME=stdorcl'

\*.log\_archive\_dest\_2='SERVICE=orcl LGWR ASYNC VALID\_FOR=(ONLINE\_LOGFILES,PRIMARY\_ROLE) DB\_UNIQUE\_NAME=orcl'

\*.log\_archive\_format='%t\_%s\_%r.arc'

\*.log\_file\_name\_convert='/home/oracle/app/oracle/oradata/orcl/','/home/oracle/app/oracle/oradata/stdorcl/'

\*.max\_dispatchers=5

\*.max\_shared\_servers=20

\*.open\_cursors=400

\*.parallel\_execution\_message\_size=8192

\*.pga\_aggregate\_target=511m

\*.processes=300

\*.recovery\_parallelism=0

\*.remote\_login\_passwordfile='EXCLUSIVE'

\*.service\_names='ORCL'

\*.sga\_max\_size=1572864000

\*.sga\_target=1258291200

\*.shared\_server\_sessions=200

\*.standby\_file\_management='AUTO'

\*.undo\_tablespace='UNDOTBS1'

* 在备库增加静态监听

gedit /home/oracle/app/oracle/product/12.1.0/dbhome\_1/network/admin/listener.ora //运行

SID\_LIST\_LISTENER =

(SID\_LIST =

(SID\_DESC =

(ORACLE\_HOME = /home/oracle/app/oracle/product/12.1.0/db\_1)

(SID\_NAME = orcl)

)

)

* 然后重启备库

shudown

startup

alter database recover managed standby database disconnect;

容灾完成！

**实验总结**

通过这次独立的建立一个家庭医疗信息系统的关系数据库，从将现实世界的对象转换成实体找出联系，画出E-R图，再将E-R图转换成关系模式，再对关系模式进行优化，转化成3NF，这样极大的减少基本表的数据冗余和对数据的操作困难。然后进行建表，我才用所有的数据操作都使用存储过程来实现，这样能极大的提高效率。创建视图，函数以及存储过程来实现各种功能，还简单了解了如何对数据库进行备份和容灾方案配置。对数据库的基本操作和理解加深了很多。但是还有许多更要学习的东西。

**表中数据（部分测试使用）**

insert\_use\_info(1000001,123456,'用户');

insert\_use\_info(1000002,123457,'用户');

insert\_use\_info(1000003,123455,'用户');

insert\_use\_info(1000004,123456,'用户');

insert\_use\_info(1000005,123453,'用户');

insert\_use\_info(2000001,654321,'专家');

insert\_use\_info(2000002,654322,'专家');

insert\_use\_info(2000003,654323,'专家');

insert\_use\_info(2000004,654324,'专家');

insert\_use\_info(2000005,654325,'专家');

insert\_expert(

insert\_expert(2000001,'张丽丽','女','18225647890','临床医生');

insert\_expert(2000002,'李政','男','19256321452','儿童医生');

insert\_expert(2000003,'李六六','男','148956963','骨科医生');

insert\_expert(2000004,'欧阳吹雪','女','14526541254','全科医生');

insert\_expert(2000005,'简自豪','男','14789654125','内科医生');

）

insert\_users(1000001,'刘谋','男',28,'19825038317','肺炎','无','海鲜过敏',2000005);

insert\_users(1000002,'张雷','女',19,'19825033333','无','无','芒果过敏',2000001);

insert\_users(1000003,'余震','男',35,'15463258971','无','感冒药','无',2000003);

insert\_users(1000004,'胡大力','男',25,'19635412585','肿瘤','无','花粉过敏',2000005);

insert\_users(1000005,'屠大波','女',18,'19360533498','中耳炎','无','无',2000004);

insert\_complaint(1001,1000001,sysdate,'专家回应速度过于缓慢');

insert\_complaint(1002,1000003,sysdate,'系统网络不稳定');

insert\_features('a01',1000001,'轻微','喉咙痛，咳嗽',sysdate);

insert\_features('a02',1000003,'严重','头晕头痛',sysdate);

insert\_features('a03',1000005,'中度','拉肚子呕吐',sysdate);

insert\_features('a04',1000002,'严重','膝关节疼痛',sysdate);

insert\_features('a05',1000004,'轻微','肚子痛',sysdate);

insert\_diagnose('a01',2000005,'感冒','病毒性感冒',sysdate,'感染了最近的流感病毒所得的呼吸道感染','进行点滴输液，照医嘱服用感冒药');

insert\_diagnose('a02',2000003,'贫血','先天性贫血',sysdate,'先天性贫血导致的头晕头痛','平时减少剧烈运动,注意补充糖分');

insert\_diagnose('a03',2000004,'食物中毒','细菌类食物中毒',sysdate,'食用的食物中大肠杆菌造成了中毒','去医院进行洗胃');

insert\_problem('p001',1000001,'如何预防感冒','公开',sysdate,'预防感冒日常需要怎么做');

insert\_problem('p002',1000001,'作息时间问题','私人',sysdate,'晚上几点睡最健康');

insert\_problem('p003',1000002,'饮食健康问题','公开',sysdate,'日常食用太多糖分会有什么危害');

insert\_answer('p001',2000001,sysdate,'平时多喝热水，保持运动','公开');

insert\_answer('p001',2000002,sysdate,'保持室内良好的空气流通','私人');

insert\_answer('p003',2000005,sysdate,'可能会导致肥胖','公开');

insert\_answer('p003',2000004,sysdate,'可能诱发心血管疾病','公开');

insert\_beta(1000001,0001,sysdate,'定量筛选试验','过敏','皮肤针刺伤,效果明显','阳性');

insert\_beta(1000002,0002,sysdate,'葡萄糖糖尿病试验','糖尿病','皮肤针刺伤，效果明显','阴性');

insert\_beta(1000005,0003,sysdate,'流感A/B试验','流感','鼻标本拭子送病理学家','阳性');

insert\_treat(0001,2000005,'皮肤过敏治疗','药物治疗','外涂药物于皮肤，阻断过敏原');

insert\_treat(0002,2000004,'糖尿病治疗','注射治疗','注射胰岛素，帮助降低血糖');

insert\_treat(0003,2000001,'流感治疗','注射治疗','使用抗生素杀死病毒');