

**期末项目设计报告**

|  |  |  |  |
| --- | --- | --- | --- |
| 题 目 | 基于Oracle的小物件租借信息系统数据库设计 | | |
| 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 | 魏希 | 学号 | 201810414124 |
| 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 | 10 |  |
| PL/SQL设计 | 存储过程和函数设计正确 | 25 |  |
| 备份方案 | 备份方案设计正确 | 25 |  |
| 容灾方案 | DataGuard设计正确 | 10 |  |
| **得分合计** | | |  |

2019 年 11 月 18 日

目录

1. 系统概述
2. 功能需求分析

1、功能概述

1. 概念结构设计

四、系统结构设计

1逻辑模型

2数据库表

五、表空间设计

六、表、约束、索引、视图管理

七、PL/SQL语句

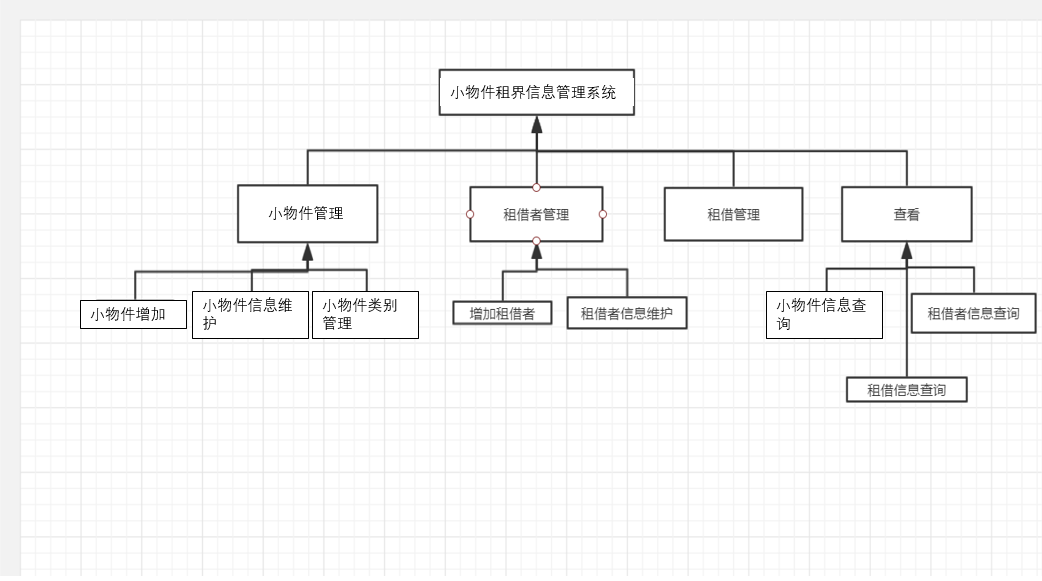
八、总结

一、系统概述

小物件租借信息管理系统是现在存在的一种为小物件拥有者提供各种类型共享的一种新型的小物件模式，目的在为那些想用某些小物件但是又无法花大价钱购买的人。本系统主要实现的是对小物件信息的管理，主要功能是为想暂时使用小物件的客户提供数据库支持，为那些共享的小物件提供有效的管理。对小物件的去向进行管理。帮助管理者和小物件客户更好的管理共享的小物件和进行良好的小物件租借体验。

二、功能需求分析

小物件租借信息系统功能模块如图所示：



2.1、功能概述：

这个小物件租借信息管理系统适用于任何想使用小物件的小物件客户和小物件提供者使用，主要任务是建立详细的租借者的信息，以及小物件的信息和种类以及对应的小物件的租借记录。方便管理员进行小物件的管理维护和跟踪。

2.1.1、管理员信息管理：管理员是本系统的管理者，具有最高权限，对小物件和租借者添加，修改和删除等功能。但管理员只能修改自己的信息，不能添加或删除自己的信息。要想添加和删除管理员必须到数据库中去操作表结构。

2.1.2、租借者信息管理：由系统管理员，给每个租借创建一个账户，账户应具有借阅者的详细信息，并且借阅者的信

息在借阅者的身份信息改变后可以修改或注销。

2.1.3、书库管理：

1）新增小物件。图书管理员对新增的入库新小物件进行试使用、分类、给出评级、价格、建库。每个小物件要有较为完整的信息，包括小物件名，小物件分类，小物件的评级，还有给出租借价格。

2）小物件信息维护。图书管理员可以对入库的小物件信息进行修改，或是删除（下架）该小物件信息。

3）小物件类别管理。新增书籍按照小物件类型进行分类。

（3）租借管理

1）小物件借阅管理。管理员根据不同级别的租借者的租借情况对其进行租借管理。

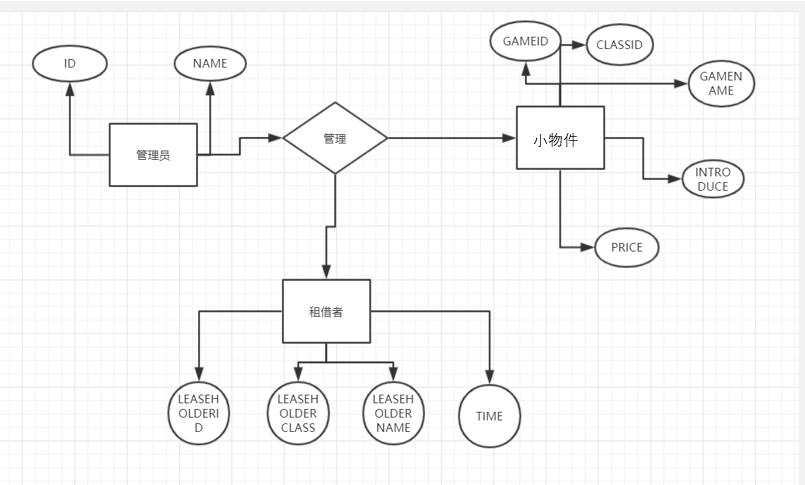
2.1.4、小物件出租管理：小物件提供商提供小物件并进行小物件维护，在租借者用它的帐号进行认证后，可以在系统里查看小物件信息和自己的租借情况，可以进行小物件的租借，在选完小物件后系统会对这个小物件的租借信息进行修改，将小物件的租借信息改为出库，并反应在租借界面上。

2.1.5、小物件归还管理：管理员可以根据租借的关键信息，查询到该租借者所有的租借信息，租借日期等，从中选择要归还的小物件，显示有关该小物件的所有租借信息，方便让管理员判断出是否过期，是否对租借者进行罚款，退还押金后并对数据库记录做出改变，，并把该小物件状态信息改为在库。

2.1.6、租借管理：管理员要能够对所有被租的小物件进行查询，了解哪些人租了哪些小物件，以及租借小物件的日期，当一个小物件租借太久后，管理员可以对租借的人员进行通知。

1. 概念结构设计

3.1 E-R图 ：



上图是该小物件租界信息管理系统总E-R图，该系统涉及的实体集有：

小物件实体集：小物件id,小物件名，小物件类别，小物件介绍，小物件价格

租借者实体集：具有属性id,级别，姓名，时间

管理员实体集：具有的属性id,姓名

一个管理员可以对多小物件进行管理管理员可以有多个，所以管理员和小物件之间的联系为m:n的联系。

一个管理员可以对多个租借者进行管理，管理员可以有多个，所以管理员和租借者之间的联系为m:n的联系。

1. 系统结构设计
2. 逻辑模型

在得出概念结构设计的E-R图后，下一步的任务就是吧概念结构设计中的E-R模型转换为具体的数据库管理系统支持的组织层数据模型，也就是导出特定的数据库管理系统可以处理的数据库逻辑结构，由于在概念结构设计中我们就得出了若干个关系模式，因此在逻辑结构设计要做的就是把各个关系模式做一遍规范化处理，以减少冗余。以下关系模式均已规范化到第三范式。

小物件表（小物件编号id，小物件类别编号，小物件名，介绍，价格）

小物件类别表（类别编号，类别名称，类别解释）

租借者表（租借者编号，租借等级，姓名，时间，已租小物件）

租借者类别表（租借等级，等级名称，最大可借小物件数，）

管理员表（管理员编号，姓名）

1. 系统数据库表

数据库总表

|  |  |
| --- | --- |
| 表名 | 中文名 |
| GAME | 小物件信息表 |
| MANGER | 管理员表 |
| LEASEHOLDER | 租借者表 |
| LEASEHOLDERCLASS | 租借等级表 |
| CLASSIDTABLE | 小物件类别表 |

小物件信息表（GAME）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 类型 | 长度 | 主/外键 | 约束 | 中文名 |
| GAMEID | VARCHAR | 20 | P | NOT NULL | 小物件id |
| CLASSID | VARCHAR | 20 | F | NOT NULL | 小物件类别编号 |
| GAMENAME | VARCHAR | 20 |  |  | 小物件名 |
| INTRODUCE | VARCHAR | 100 |  |  | 介绍 |
| PRICE | VARCHAR | 20 |  |  | 租借价格 |

管理员表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 类型 | 长度 | 主/外键 | 约束 | 中文名 |
| ID | VARCHAR | 20 | P | NOT NULL | 管理员id |
| NAME | VARCHAR | 20 |  | NOT NULL | 管理员名 |

租借者表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 类型 | 长度 | 主/外键 | 约束 | 中文名 |
| LEASEHOLDERID | VARCHAR | 20 | P | NOT NULL | 租借者id |
| LEASEHOLDERCLASS | VARCHAR | 20 | F | NOT NULL | 租借等级 |
| LEASEHOLDERNAME | VARCHAR | 20 |  |  | 租借者姓名 |
| TIME | VARCHAR | 100 |  |  | 时限 |

租借等级表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 类型 | 长度 | 主/外键 | 约束 | 中文名 |
| LEASEHOLDERCLASS | VARCHAR | 20 | P | NOT NULL | 租界等级 |
| LEASEHOLDERCLASSNAME | VARCHAR | 20 |  | NOT NULL | 租界等级名 |
| MAXBORROW | VARCHAR | 20 |  |  | 最大数 |

小物件类别表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 类型 | 长度 | 主/外键 | 约束 | 中文名 |
| CLASSID | VARCHAR | 20 | P | NOT NULL | 类别id |
| CLASSNAME | VARCHAR | 20 |  | NOT NULL | 小物件类别 |
| CLASSEXPLAIN | VARCHAR | 20 |  |  | 描述 |

1. 表空间设计

/\*第1步：创建表空间 \*\*/

CREATE TABLESPACE game\_data DATAFILE

'home:\desktop\space\game\_data.dbf' SIZE 2048m

AUTOEXTEND ON NEXT 10m;

/\*

创建表空间game\_data

\*/

create tablespace geme\_idx

datafile 'home:\desktop\space\game\_idx.dbf'

size 2048m

autoextend on next 10m;

/\*

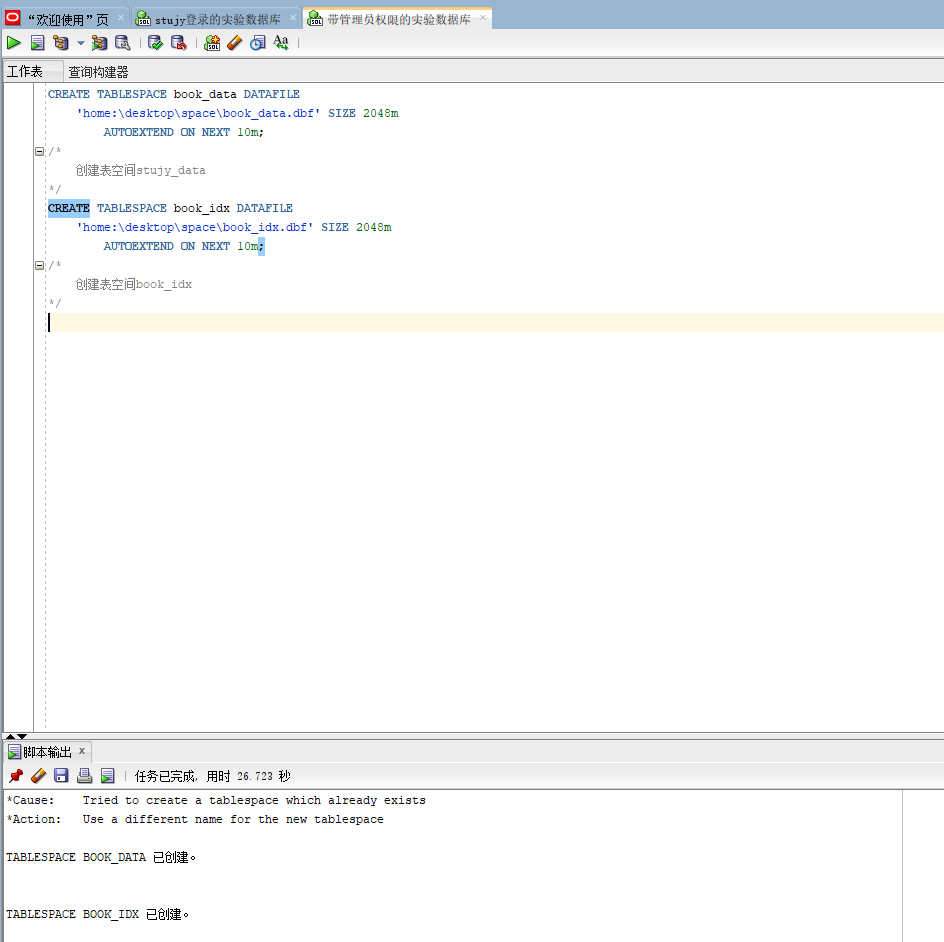
创建表空间game\_idx

\*/

/\*第2步：创建用户并指定表空间 \*/

create user gameleaseholder identified by 123456;

create user gamemanger identified by 000000;



/\*

创建用户gameleaseholder, gamemanger

\*/

create role borrow;

create role sale;

/\* 创建两个角色\*/

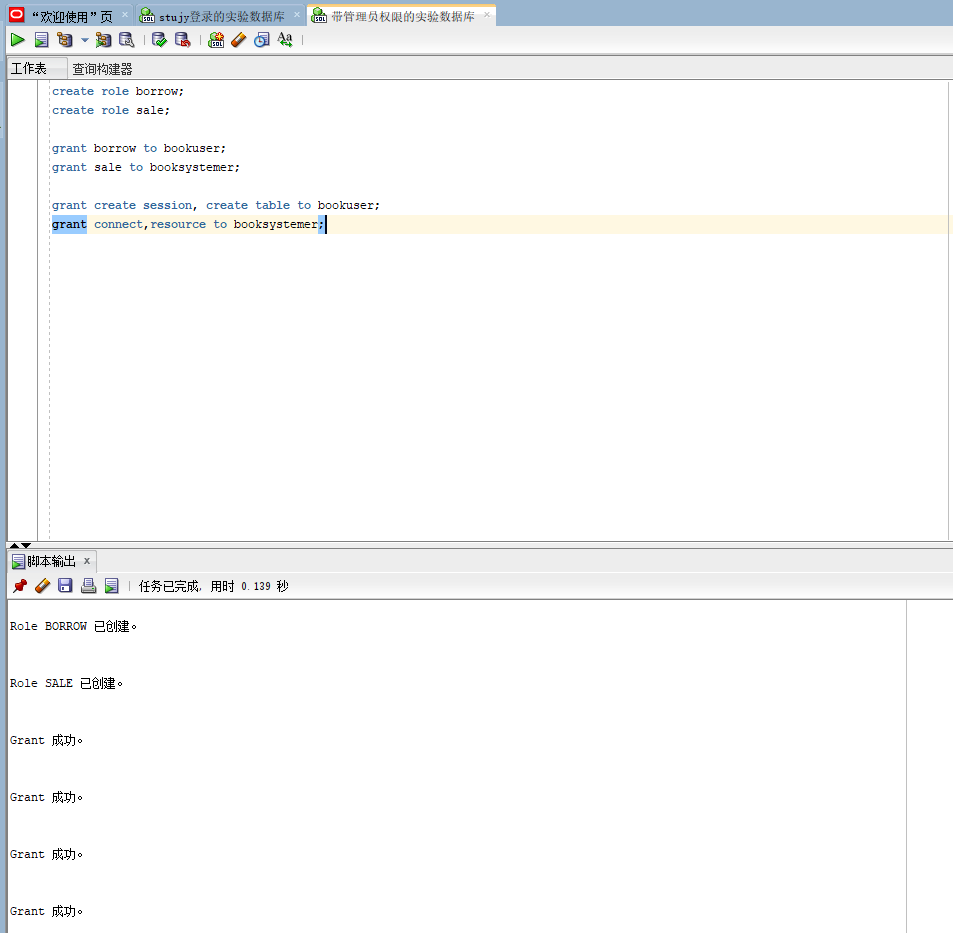
grant borrow to gameleaseholder;

grant sale to gamemanger;

/\*第4步：给用户授予权限 \*/

grant create session, create table to gameleaseholder;

grant connect,resource to gamemanger;



1. 具体设计

1.表、约束、索引、视图管理

1.创建表代码

--创建GAME表

CREATE TABLE "GAME" (

"GAMEID" VARCHAR(20) NOT NULL,

"CLASSID" VARCHAR(20) NOT NULL,

"GAMENAME" VARCHAR(20),

"INTRODUCE" VARCHAR(100),

"PRICE" VARCHAR(20),

CONSTRAINT pk\_GAME PRIMARY KEY ("GAMEID")

);

comment on table Game is '小物件基本信息表';

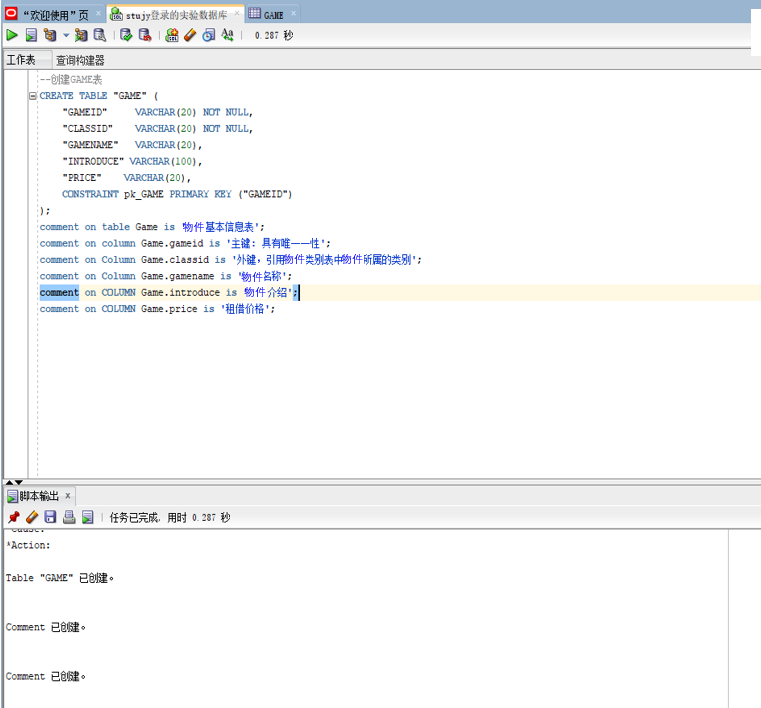
comment on column Game.gameid is '主键: 具有唯一一性';

comment on Column Game.classid is '外键，引用小物件类别表中小物件所属的类别';

comment on Column Game.gamename is '小物件名称';

comment on COLUMN Game.introduce is '小物件介绍';

comment on COLUMN Game.price is '租借价格';



--创建管理员表

CREATE TABLE "MANGER" (

"ID" VARCHAR(20) NOT NULL,

"NAME" VARCHAR(20) NOT NULL,

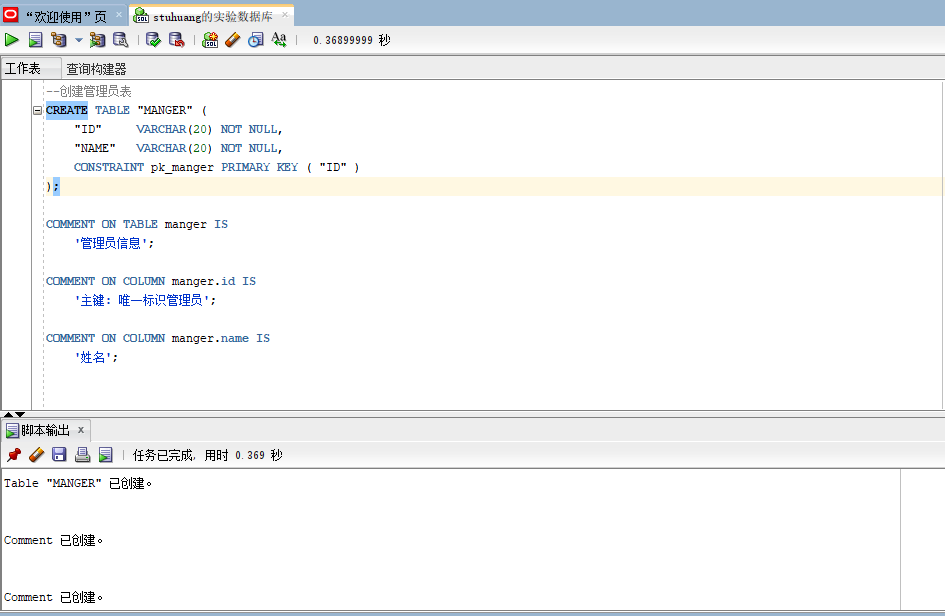
CONSTRAINT pk\_MANGER PRIMARY KEY ( "ID" )

);

comment on table Manger is '管理员信息';

comment on column Manger.ID is '主键: 唯一标识管理员';

comment on Column Manger.name is '姓名';



--创建租借者表

CREATE TABLE "LEASEHOLDER" (

"LEASEHOLDERID" VARCHAR(20) NOT NULL,

"LEASEHOLDERCLASS" VARCHAR(20) NOT NULL,

"LEASEHOLDERNAME" VARCHAR(20),

"TIME" VARCHAR(20),

CONSTRAINT pk\_LEASEHOLDER PRIMARY KEY ("LEASEHOLDERID")

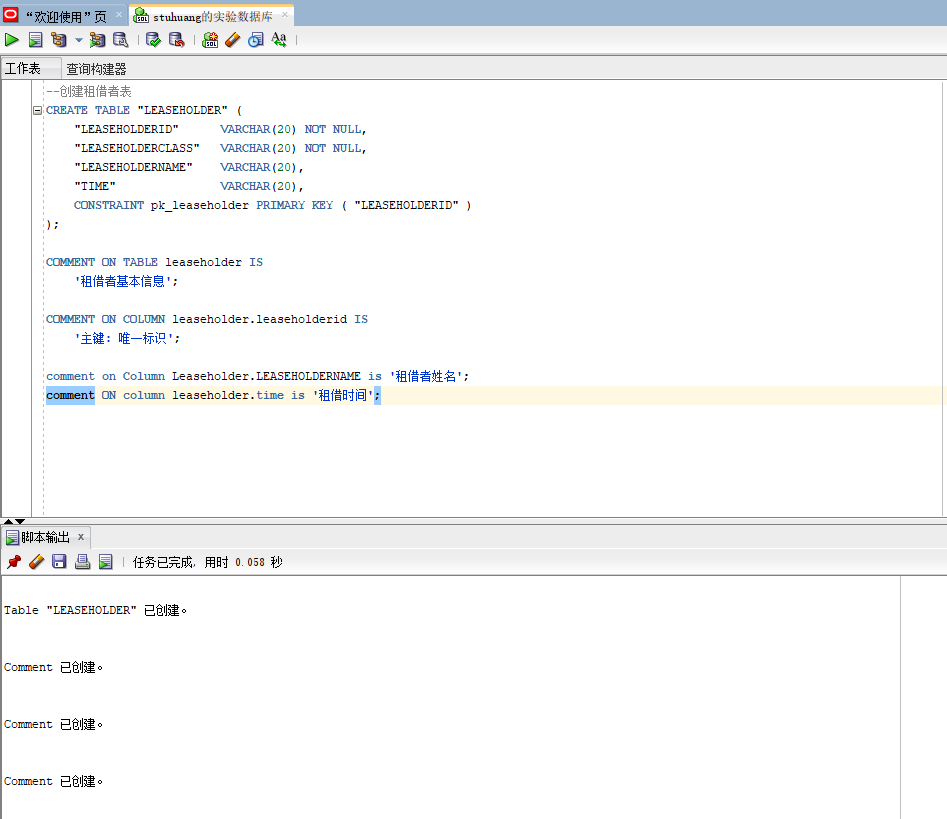
);

comment on table Leaseholder is '租借者基本信息';

comment on column Leaseholder.LEASEHOLDERID is '主键: 唯一标识';

commnent on Column Leaseholder.LEASEHOLDERNAME is '租借者姓名';

commnent on Column Leaseholder.TIME is '租借时间';



--创建租借等级表

CREATE TABLE "LEASEHOLDERCLASS" (

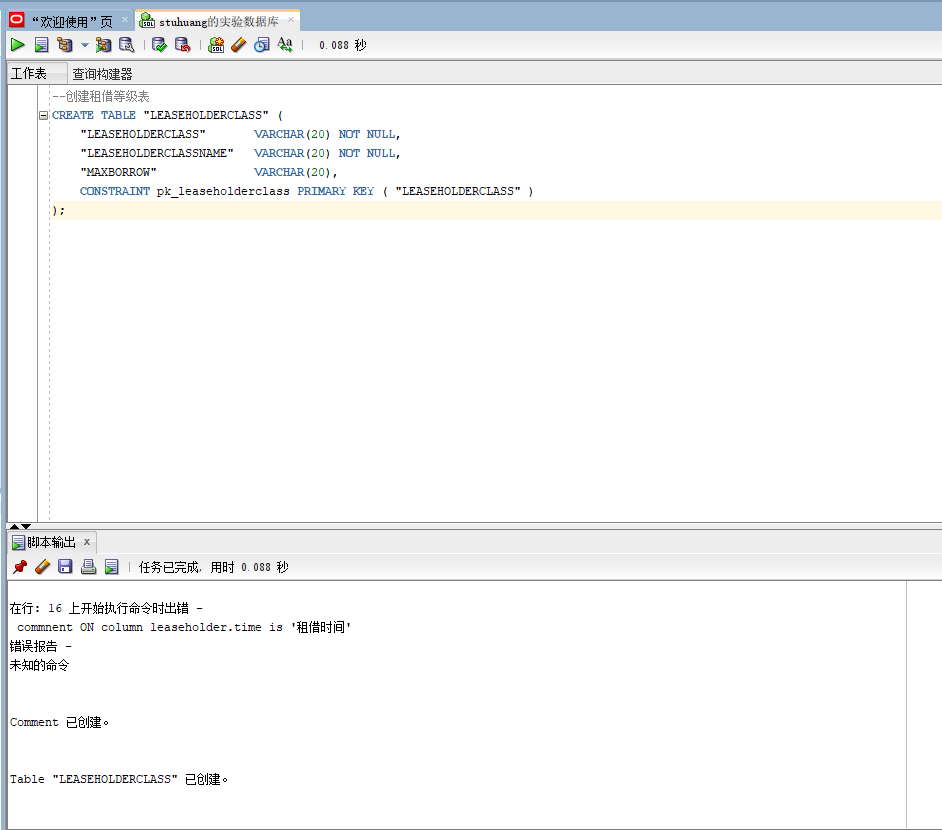
"LEASEHOLDERCLASS" VARCHAR(20) NOT NULL,

"LEASEHOLDERCLASSNAME" VARCHAR(20) NOT NULL,

"MAXBORROW" VARCHAR(20),

CONSTRAINT pk\_LEASEHOLDERCLASS PRIMARY KEY ("LEASEHOLDERCLASS")

);



--创建小物件类别表

CREATE TABLE "CLASSIDTABLE" (

"CLASSID" VARCHAR(20) NOT NULL,

"CLASSNAME" VARCHAR(20) NOT NULL,

"CLASSEXPLAIN" VARCHAR(20),

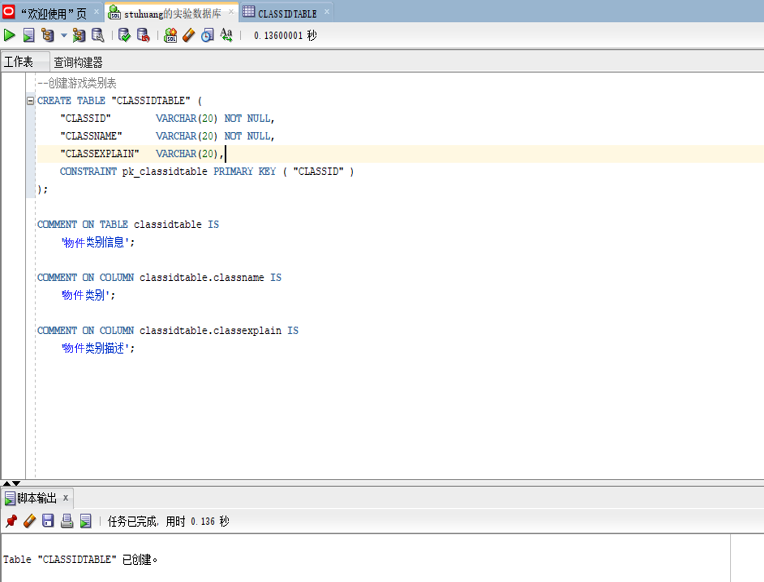
CONSTRAINT pk\_CLASSIDTABLE PRIMARY KEY ("CLASSID")

);

comment on table Classidtabe is '小物件类别信息';

comment on column Classidtabe.CLASSNAME is '小物件类别';

comment on column Classidtabe.CLASSEXPLAIN is '小物件类别描述';



1. PL/SQL语句
2. 写入数据

CREATE OR REPLACE PACKAGE write\_data IS --write\_data为包名

PROCEDURE writedata; --声明一个过程

END write\_data;

--主体

CREATE OR REPLACE PACKAGE BODY write\_data IS --包名必须一致

PROCEDURE writedata IS --实现规范中的过程

BEGIN

FOR i IN 1..10000 LOOP INSERT INTO game VALUES (

i + 1,

'1',

'pocketmonster',

'一个收集养成类小物件',

'15'

);

END LOOP;

COMMIT;

FOR i IN 1..50000 LOOP INSERT INTO manger VALUES (

i,

'stujy'

);

END LOOP;

COMMIT;

FOR i IN 1..50000 LOOP INSERT INTO leaseholder VALUES (

i,

'5',

'jiangyuan',

'30'

);

END LOOP;

COMMIT;

FOR i IN 1..50000 LOOP INSERT INTO leaseholderclass VALUES (

i,

'jiangyuan',

'5'

);

END LOOP;

COMMIT;

FOR i IN 1..50000 LOOP INSERT INTO classidtable VALUES (

i,

'nurturance',

'收集并变强类型小物件'

);

END LOOP;

COMMIT;

END writedata;

END write\_data;

1. 查看数据

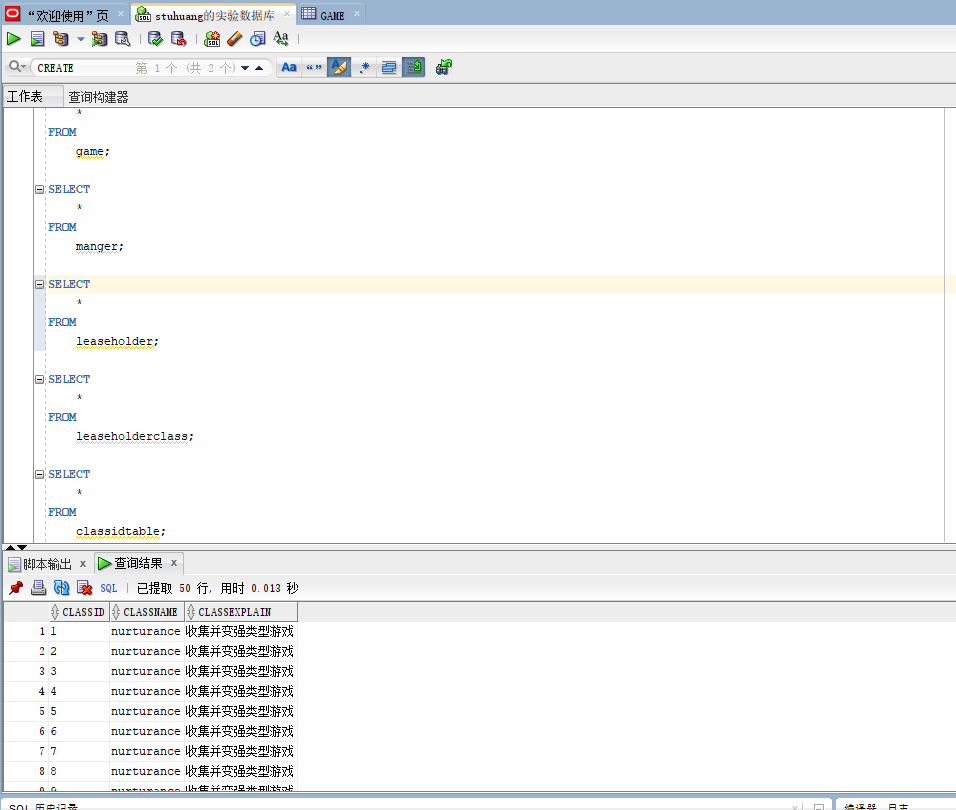
select \* from GAME;

select \* from MANGER

select \* from LEASEHOLDER;

select \* from LEASEHOLDERCLASS;

select \* from CLASSIDTABLE;



1. 总结

小物件的租借在未来的应该会时很常见的，在未来云小物件发展起向我们这类信息管理系统会是常见的，也是必须的。这次的数据库的课程设计，基本可以看作是对之前的几次实验的结合深化，这次综合项目实验就是把之前的东西结合起来，相信在后面的深入学习中将会更加的使用Oracle数据库。这次的综合项目对自己今后参与开发数据库系统积累了不少经验，在实验过程中，从建立数据开始，对数据库设计要求上有更高的认识，从需求分析，到概念设计和逻辑设计，E-R图的表示，数据字典的创建，懂得了不少有关数据库开发过程中的知识，在实验中建表，及其关系模式，关系代数的建立及理解，增强了自己在数据库中应用SQL语言的灵活性。在学习过程中，我也能过上网查了不少资料，对于建立数据库过程中遇到的问题有了更深入的记忆。