

**期末项目设计报告**

|  |  |  |  |
| --- | --- | --- | --- |
| 题 目 | 基于Oracle的个人博客项目的数据库设计 | | |
| 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 | 陈祥 | 学号 | 201810414307 |
| 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

目录

[期末考核要求 2](#_Toc10551)

[1.项目简介 3](#_Toc131)

[2.功能分析 3](#_Toc13240)

[3.E-R图设计 4](#_Toc28220)

[4.类图设计 5](#_Toc1010)

[5.数据库设计 5](#_Toc28155)

[5.1用户表 5](#_Toc32147)

[5.2 分类表 6](#_Toc14374)

[5.3 好友表 6](#_Toc5390)

[5.4 博文表 7](#_Toc15971)

[5.5 评论表 7](#_Toc4202)

[5.6 标签表 8](#_Toc17841)

[6.项目流程 8](#_Toc9079)

[6.1创建表空间 8](#_Toc21060)

[6.2创建表 8](#_Toc27316)

[6.3添加数据 11](#_Toc31566)

[6.4设计权限及用户分配方案,两类角色，两个用户 15](#_Toc8816)

[6.5存储过程和函数 16](#_Toc11812)

[6.6自动备份方案（exp命令备份） 16](#_Toc9198)

[6.6.1新建 backup.bat脚本 17](#_Toc25427)

[6.6.2创建 windows任务计划 17](#_Toc16520)

[7.项目总结 21](#_Toc7594)

**期末考核要求**

* 自行设计一个信息系统的数据库项目，自拟某项目名称。
* 设计项目涉及的表及表空间使用方案。至少5张表和5万条数据，两个表空间。
* 设计权限及用户分配方案。至少两类角色，两个用户。
* 在数据库中建立一个程序包，在包中用PL/SQL语言设计一些存储过程和函数，实现比较复杂的业务逻辑，用模拟数据进行执行计划分析。
* 设计自动备份方案或则手工备份方案。

# 1.项目简介

项目名称：博客系统

项目介绍：基于Oracle的个人博客系统数据库设计。

涉及角色/用户：普通用户、游客

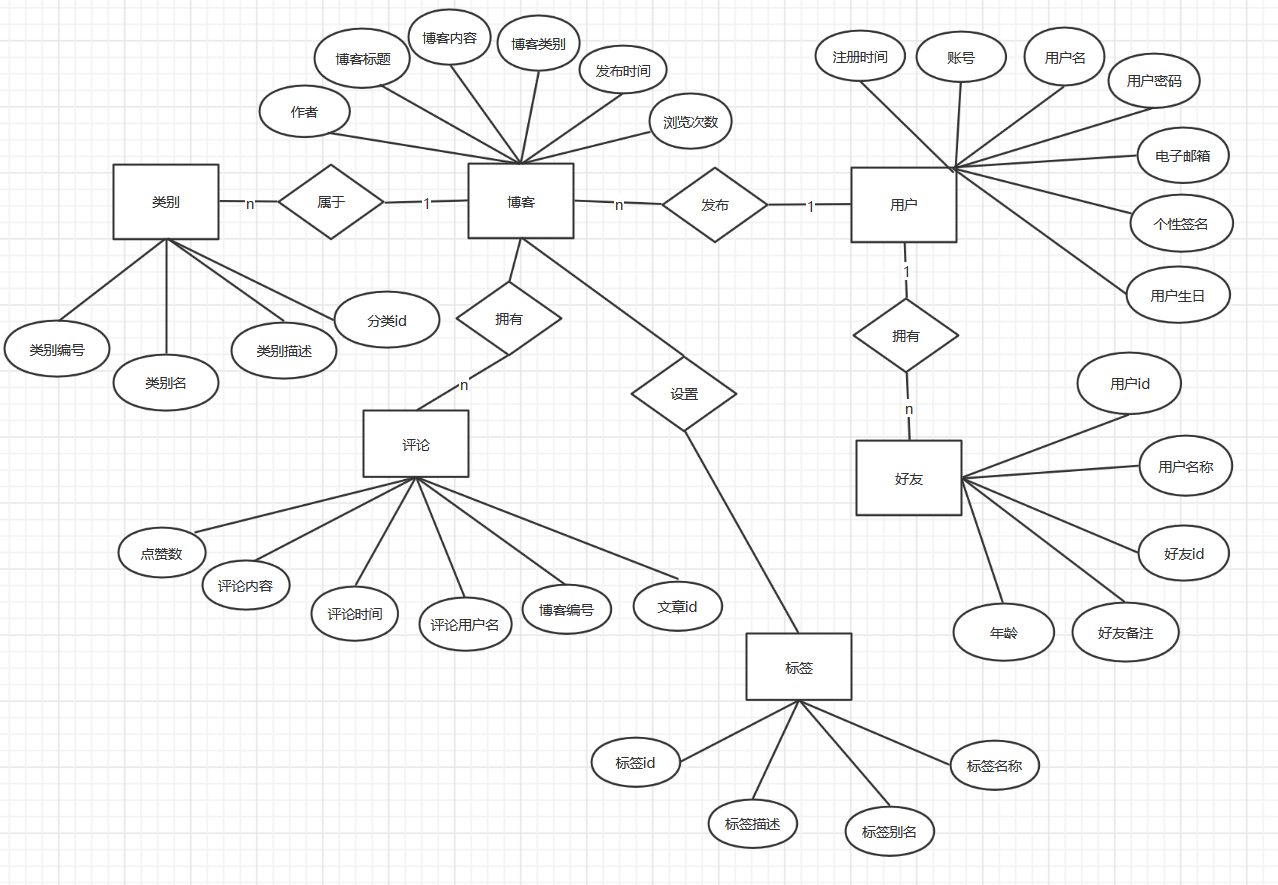
涉及表：用户表、游客表、朋友表、文章标签表、文章分类表、内容表、评论表

# 2.功能分析

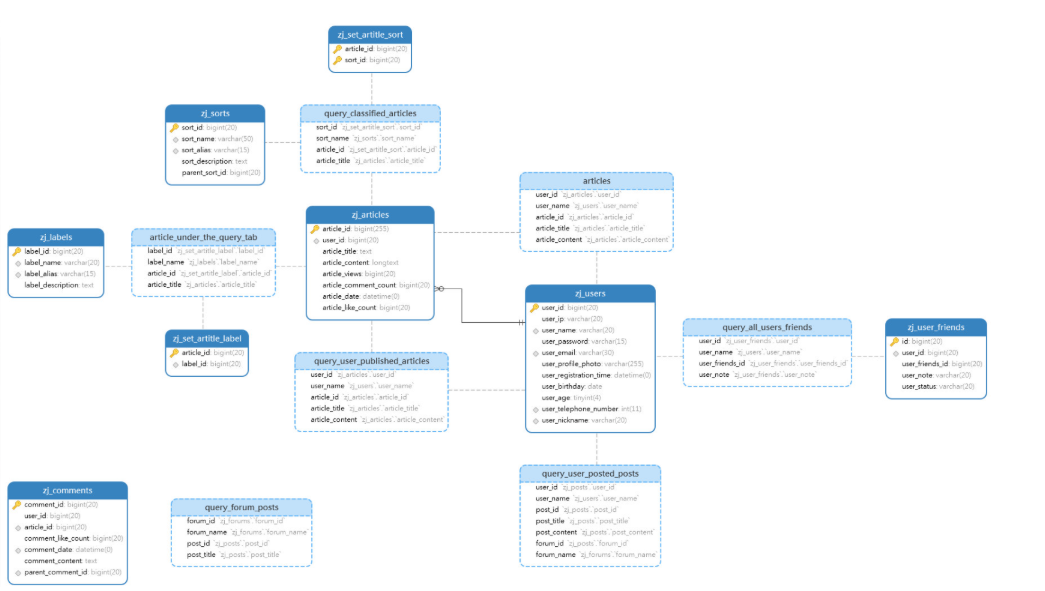
系统主要完成以下几方面的功能：

* 用户管理：用户的注册和登录，登录后可以发表博文和评论，点赞。
* 博文管理：用户可以在网站中对自己的博文进行增删改查。
* 评论管理：用户可以评论博文和回复其他用户的评论。
* 分类管理：添加和删除分类，给文章设置分类。
* 标签管理：添加和删除标签，给文章设置标签。

# 3.E-R图设计



# 4.类图设计



# 5.数据库设计

## 5.1用户表



## 5.2 分类表



## 5.3 好友表



## 5.4 博文表



## 5.5 评论表



## 5.6 标签表



# 6.项目流程

## 6.1创建表空间

CREATE TABLESPACE Users02

DATAFILE

'/home/oracle/app/oracle/oradata/orcl/pdborcl/date01.dbf'

SIZE 100M AUTOEXTEND ON NEXT 256M MAXSIZE UNLIMITED,

'/home/oracle/app/oracle/oradata/orcl/pdborcl/date02.dbf'

SIZE 100M AUTOEXTEND ON NEXT 256M MAXSIZE UNLIMITED

EXTENT MANAGEMENT LOCAL SEGMENT SPACE MANAGEMENT AUTO;

## 6.2创建表

-- user表的创建

create table manage

(

user\_id number(20) primary key,

user\_ip VARCHAR2(20 BYTE） not null,

user\_name VARCHAR2(100 BYTE） not null,

user\_password VARCHAR2(20 BYTE）not null,

user\_email VARCHAR2(100 BYTE） not null,

user\_profile\_photo VARCHAR2(255 BYTE） not null,

user\_registration\_time datetime not null,

user\_birthday date not null,

user\_age number(20) not null,

user\_telephone\_number number(20) not null,

user\_nickname VARCHAR2(100 BYTE）

)

tablespace date01

pctfree 10

initrans 1

maxtrans 255

storage

(

initial 64K

next 1M

minextents 1

maxextents unlimited

);

-- Friends表的创建

create table Friends

(

id number(20) primary key,

user\_id number(20) not null,

user\_friends\_id number(20) not null,

user\_note VARCHAR2(100 BYTE） not null,

user\_status VARCHAR2(100 BYTE）

)

tablespace date01

pctfree 10

initrans 1

maxtrans 255

storage

(

initial 64K

next 1M

minextents 1

maxextents unlimited

);

-- 分类表的创建

create table sort

(

sort\_id number(20) primary key,

sort\_name VARCHAR2(100 BYTE） not null,

sort\_alias VARCHAR2(100 BYTE） not null,

sort\_description VARCHAR2(255 BYTE） not null,

parrent\_sort\_id number(20)

)

tablespace date02

pctfree 10

initrans 1

maxtrans 255

storage

(

initial 64K

next 1M

minextents 1

maxextents unlimited

);

-- 博文表的创建

create table article

(

article\_id number(20) primary key,

user\_id number(20) not null,

article\_title VARCHAR2(255 BYTE） not null,

article\_content VARCHAR2(100 BYTE） not null,

article\_views number(20) not null,

article\_comment\_count number(20) not null,

artcle\_date DATETIME not null,

article\_like\_count number(20)

)

tablespace date02

pctfree 10

initrans 1

maxtrans 255

storage

(

initial 64K

next 1M

minextents 1

maxextents unlimited

);

-- 评论表的创建

create table comment

(

comment\_id number(20) primary key,

user\_id number(20) not null,

comment\_id number(20) not null,

comment\_like\_count number(20) not null,

comment\_date DATETIME not null,

comment\_content VARCHAR2(255 BYTE） not null,

parent\_comment\_id number(20)

)

tablespace date02

pctfree 10

initrans 1

maxtrans 255

storage

(

initial 64K

next 1M

minextents 1

maxextents unlimited

);

-- 标签表的创建

create table label

(

label\_id number(20) primary key,

label\_name VARCHAR2(255 BYTE） not null,

label\_alias VARCHAR2(100 BYTE） not null,

label\_description VARCHAR2(255 BYTE） not null

)

tablespace date02

pctfree 10

initrans 1

maxtrans 255

storage

(

initial 64K

next 1M

minextents 1

maxextents unlimited

);

## 6.3添加数据

-- user表的数据插入20000条

declare

i int;

user\_id number(20) primary key;

user\_ip VARCHAR2(20 BYTE） not null;

user\_name VARCHAR2(100 BYTE） not null;

user\_password VARCHAR2(20 BYTE）not null;

user\_email VARCHAR2(100 BYTE） not null;

user\_profile\_photo VARCHAR2(255 BYTE） not null;

user\_registration\_time datetime not null;

user\_birthday date not null;

user\_age number(20) not null;

user\_telephone\_number number(20) not null;

user\_nickname VARCHAR2(100 BYTE）;

begin

i:=1;

while i<=20000

loop

user\_id:=i;

user\_ip:=i;

user\_name:= 'user'|| i;

user\_password := '123'|| i;

user\_email := '111222333'|| i;

user\_profile\_photo :='photo'||i;

user\_registration\_time :='time'||i;

user\_birthday :='birthday'||i;

user\_age := '18'|| i;

user\_telephone\_number := '1808494'|| i;

user\_nickname := 'fengsheng'||i

insert into user(USER\_ID,USER\_ID,USER\_NAME,USER\_PASSWORD,USER\_EMAIL,USER\_PROFILE\_PHOTO,USER\_REGISTRATION\_TIME,USER\_BIRTHDAY,USER\_AGE,USER\_TELEPHONT\_NUMBER,USER\_NICKNAME) values (user\_id,user\_ip,user\_name,user\_password,user\_email,user\_profile\_photo,user\_registration\_time,user\_birthday,user\_age,user\_telephone\_number,user\_nickname);

i:=i+1;

end loop;

commit;

end;

/

-- friends表的数据插入20000条

declare

i int;

id number(20) primary key;

user\_id number(20) not null;

user\_friends\_id number(20) not null;

user\_note VARCHAR2(100 BYTE） not null;

user\_status VARCHAR2(100 BYTE);

begin

i:=1;

while i<=20000

loop

id:=i;

user\_id:='用户id'||i;

user\_friends\_id:= '好友id'|| i;

user\_note := '备注'|| i;

user\_status := '状态'|| i;

insert into user(ID,USER\_ID,USER\_FRIENDS\_ID,USER\_NOTE,USER\_STATUS) values (id,user\_id,user\_friends\_id,user\_note,user\_status);

i:=i+1;

end loop;

commit;

end;

/

-- 分类表的数据插入20000条

declare

i int;

sort\_id number(20) primary key6;

sort\_name VARCHAR2(100 BYTE） not null;

sort\_alias VARCHAR2(100 BYTE） not null;

sort\_description VARCHAR2(255 BYTE） not null;

parrent\_sort\_id number(20);

begin

i:=1;

while i<=20000

loop

sort\_id:=i;

sort\_name:='分类名称'||i;

sort\_alias:= '分类别名'|| i;

sort\_description := '分类描述'|| i;

parrent\_sort\_id := '父分类id'|| i;

insert into user(ID,USER\_ID,USER\_FRIENDS\_ID,USER\_NOTE,USER\_STATUS) values (id,user\_id,user\_friends\_id,user\_note,user\_status);

i:=i+1;

end loop;

commit;

end;

/

-- 博文表的数据插入10000条

declare

i int;

article\_id number(20) primary key;

user\_id number(20) not null;

article\_title VARCHAR2(255 BYTE） not null;

article\_content VARCHAR2(100 BYTE） not null;

article\_views number(20) not null;

article\_comment\_count number(20) not null;

artcle\_date DATETIME not null;

article\_like\_count number(20);

begin

i:=1;

while i<=10000

loop

article\_id:=i;

user\_id:=i;

article\_title:='博文标题'||i;

article\_content:='博文内容'||i;

article\_views:= i;

article\_comment\_count := i;

artcle\_date:=to\_date('2019-7-2','yyyy-mm-dd')+(i mod 60);

article\_like\_count:= i;

insert into user(ARTICLE\_ID,USER\_ID,ARTICLE\_TITLE,ARTICLE\_CONTENT,ARTICLE\_VIEWS,ARTICLE\_COMMENT\_COUNTARTICLE\_DATE,ARTICLE\_LIKE\_COUNT) values (article\_id,user\_id,article\_title,article\_content,article\_views,article\_comment\_count,artcle\_date,article\_like\_count);

i:=i+1;

end loop;

commit;

end;

/

-- 评论表的数据插入10000条

declare

i int;

comment\_id number(20) primary key;

user\_id number(20) not null;

comment\_id number(20) not null;

comment\_like\_count number(20) not null;

comment\_date DATETIME not null;

comment\_content VARCHAR2(255 BYTE） not null;

parent\_comment\_id number(20);

begin

i:=1;

while i<=10000

loop

comment\_id:=i;

user\_id:=i;

comment\_id:=i;

comment\_like\_count:=i;

comment\_date:=to\_date('2020-7-12','yyyy-mm-dd')+(i mod 60);

comment\_content:='评论内容'||i;

parent\_comment\_id:=i;

insert into user(COMMENT\_ID,USER\_ID,COMMENT\_ID,COMMENT\_LIKE\_COUNT,COMMENT\_DATE,COMMENT\_CONTENT,PARENT\_COMMENT\_ID) values (comment\_id,user\_id,comment\_id,comment\_like\_count,comment\_date,comment\_content,parent\_comment\_id);

i:=i+1;

end loop;

commit;

end;

/

-- 标签表的数据插入10000条

declare

i int;

label\_id number(20) primary key;

label\_name VARCHAR2(255 BYTE） not null;

label\_alias VARCHAR2(100 BYTE） not null;

label\_description VARCHAR2(255 BYTE） not null;

begin

i:=1;

while i<=10000

loop

label\_id:=i;

label\_name:='标签名'||i;

label\_alias:='标签别名'||i;

label\_description:='标签描述'i;

insert into user(LABEL\_ID,LABEL\_NAME,LABEL\_ALIAS,LABEL\_DESCRIPTION) values (label\_id,label\_name,label\_alias,label\_description);

i:=i+1;

end loop;

commit;

end;

/

## 6.4设计权限及用户分配方案,两类角色，两个用户

-- 创建角色

CREATE ROLE Visitor;

CREATE ROLE User;

-- 授权角色

GRANT connect,resource,dba, create table,create view,create trigger, create procedure,create sequence TO user

GRANT connect,resource, create table,create view,create trigger, create procedure,create sequence TO visitor;

-- 创建用户

CREATE Visitor visitor IDENTIFIED BY 123 DEFAULT TABLESPACE date01;

-- 创建用户

CREATE USER user IDENTIFIED BY 123 DEFAULT TABLESPACE date01,date02;

-- 分配角色给用户

GRANT Visitor TO visitor;

GRANT User TO user;

## 6.5存储过程和函数

create or replace PACKAGE BODY MyPack IS

FUNCTION Get\_SaleAmount(V\_DEPARTMENT\_ID NUMBER) RETURN NUMBER

AS

N NUMBER(20,2);

BEGIN

SELECT SUM(O.TRADE\_RECEIVABLE) into N FROM ORDERS O,EMPLOYEES E

WHERE O.EMPLOYEE\_ID=E.EMPLOYEE\_ID AND E.DEPARTMENT\_ID =V\_DEPARTMENT\_ID;

RETURN N;

END;

PROCEDURE GET\_EMPLOYEES(V\_EMPLOYEE\_ID NUMBER)

AS

LEFTSPACE VARCHAR(2000);

begin

--通过LEVEL判断递归的级别

LEFTSPACE:=' ';

--使用游标

for v in

(SELECT LEVEL,EMPLOYEE\_ID,NAME,MANAGER\_ID FROM employees

START WITH EMPLOYEE\_ID = V\_EMPLOYEE\_ID

CONNECT BY PRIOR EMPLOYEE\_ID = MANAGER\_ID)

LOOP

DBMS\_OUTPUT.PUT\_LINE(LPAD(LEFTSPACE,(V.LEVEL-1)\*4,' ')||

V.EMPLOYEE\_ID||' '||v.NAME);

END LOOP;

END;

END MyPack;

## 6.6自动备份方案（exp命令备份）

6.6.1新建 backup.bat脚本

@echo off

set curdate=%date:~0,4%%date:~5,2%%date:~8,2%

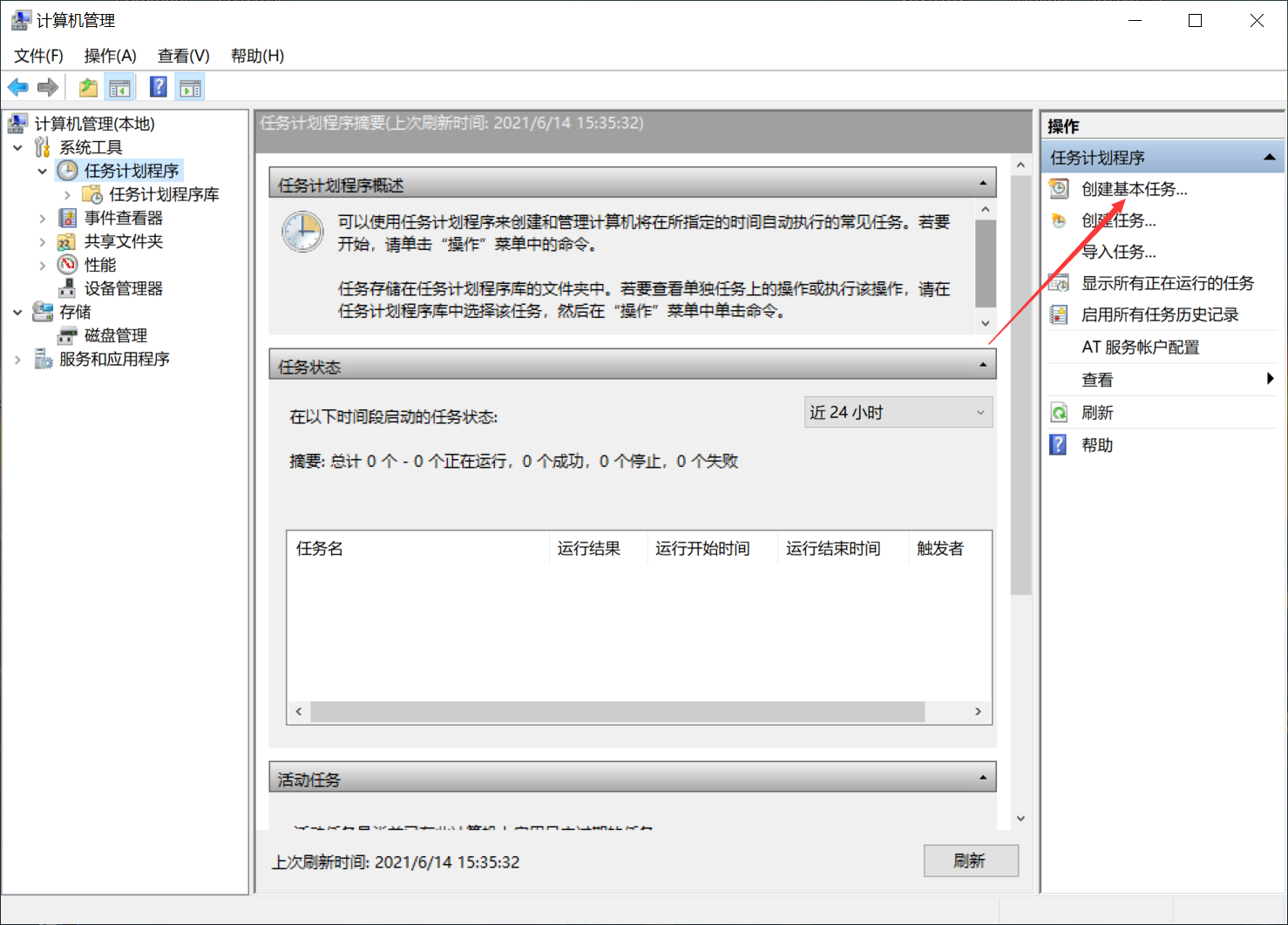
set backuppath=D:\backup

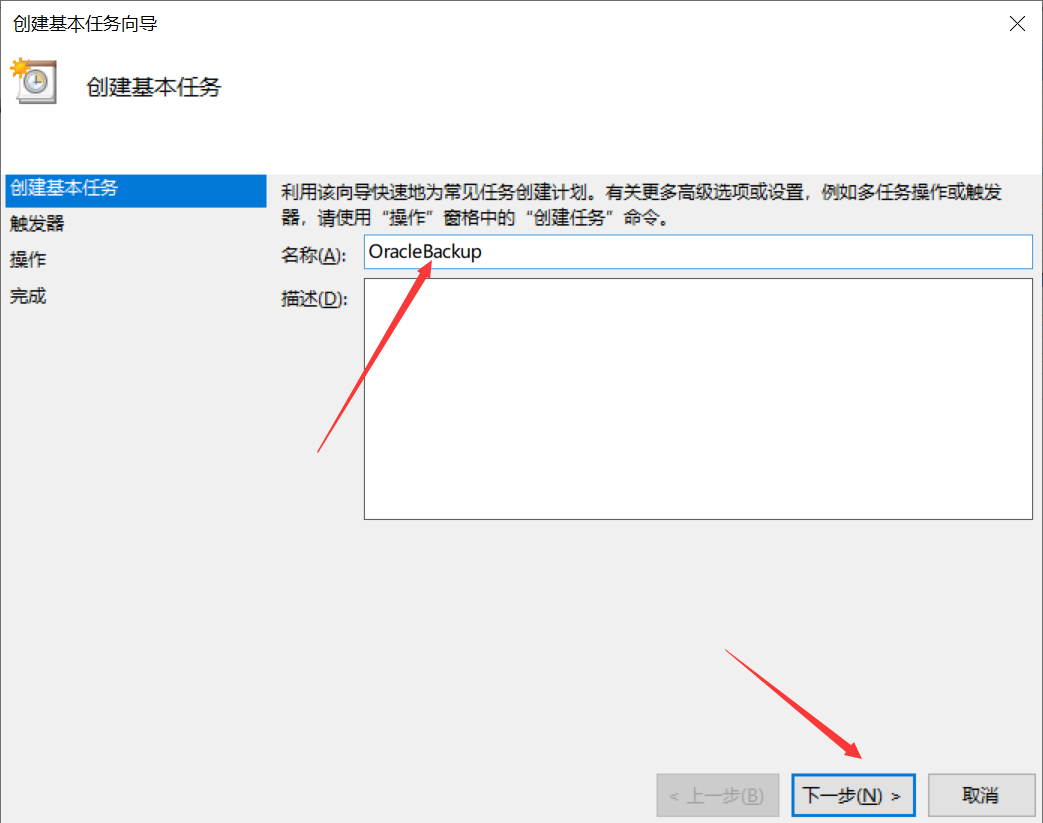
md %backuppath%

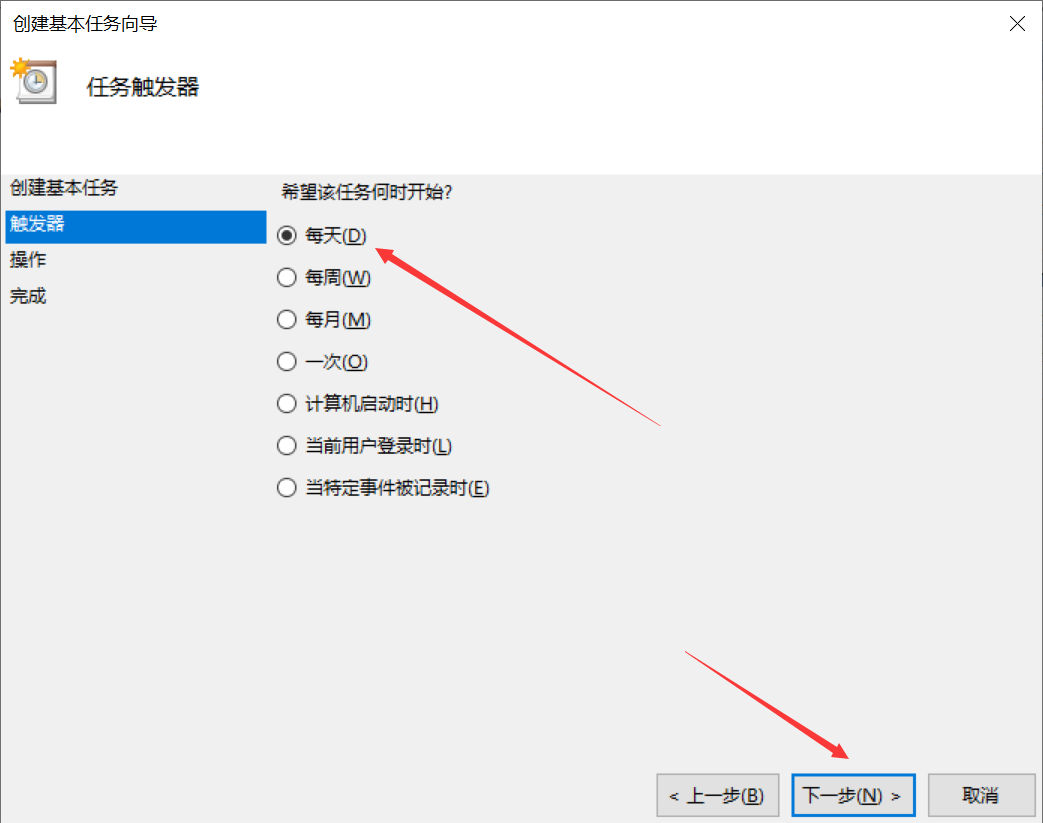
exp scott/tiger@127.0.0.1/orcl file=%backuppath%\%curdate%.dmp owner(scott) log=%backuppath%\%curdate%.log

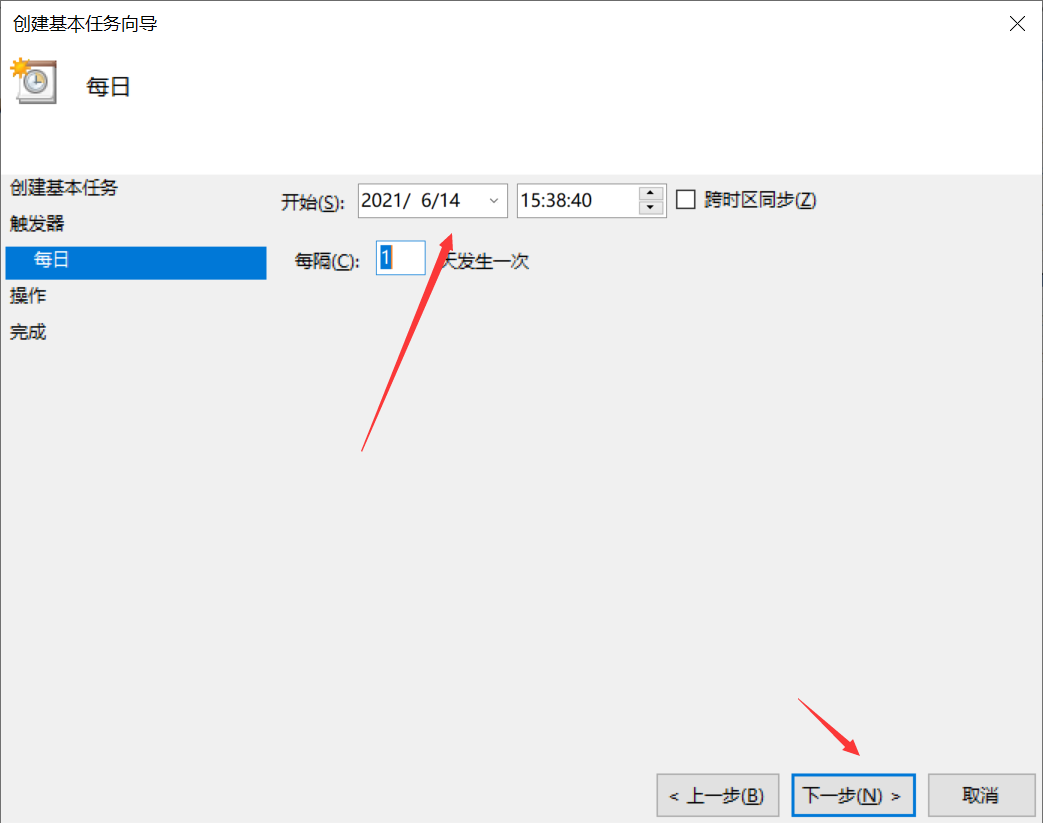
forfiles /p "D:\backup" /s /m \*.dmp /d -30 /c "cmd /c del @file"

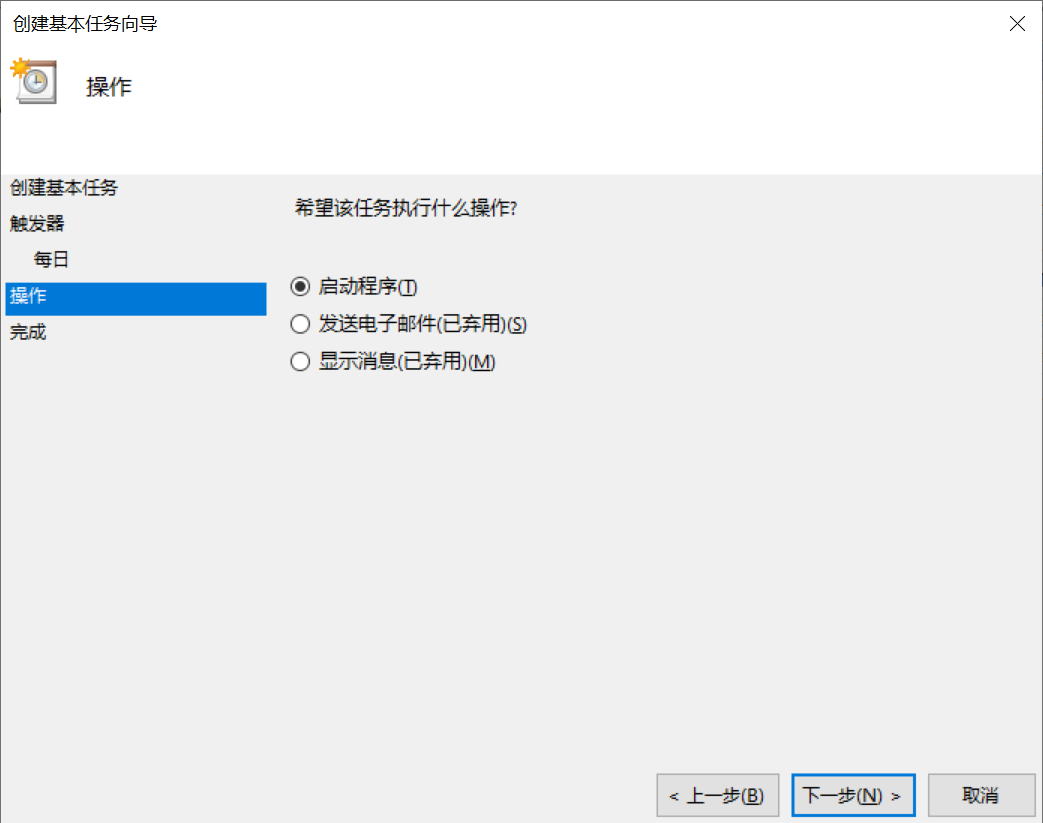
6.6.2创建 windows任务计划

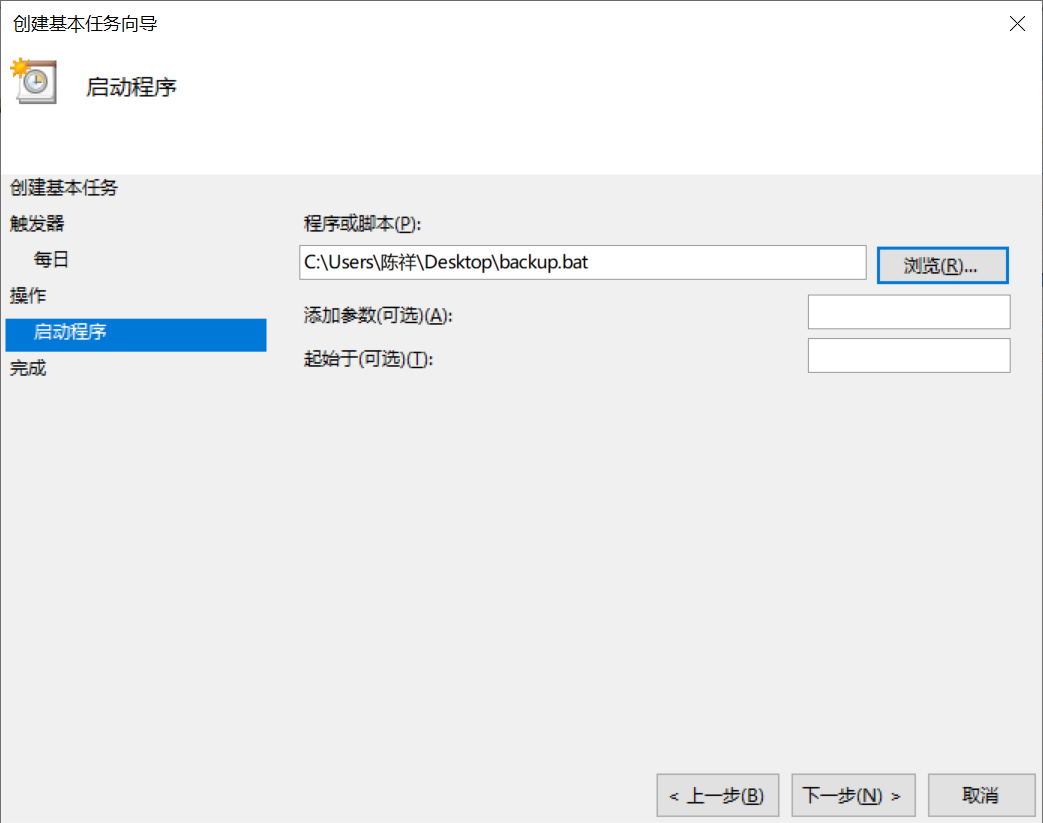


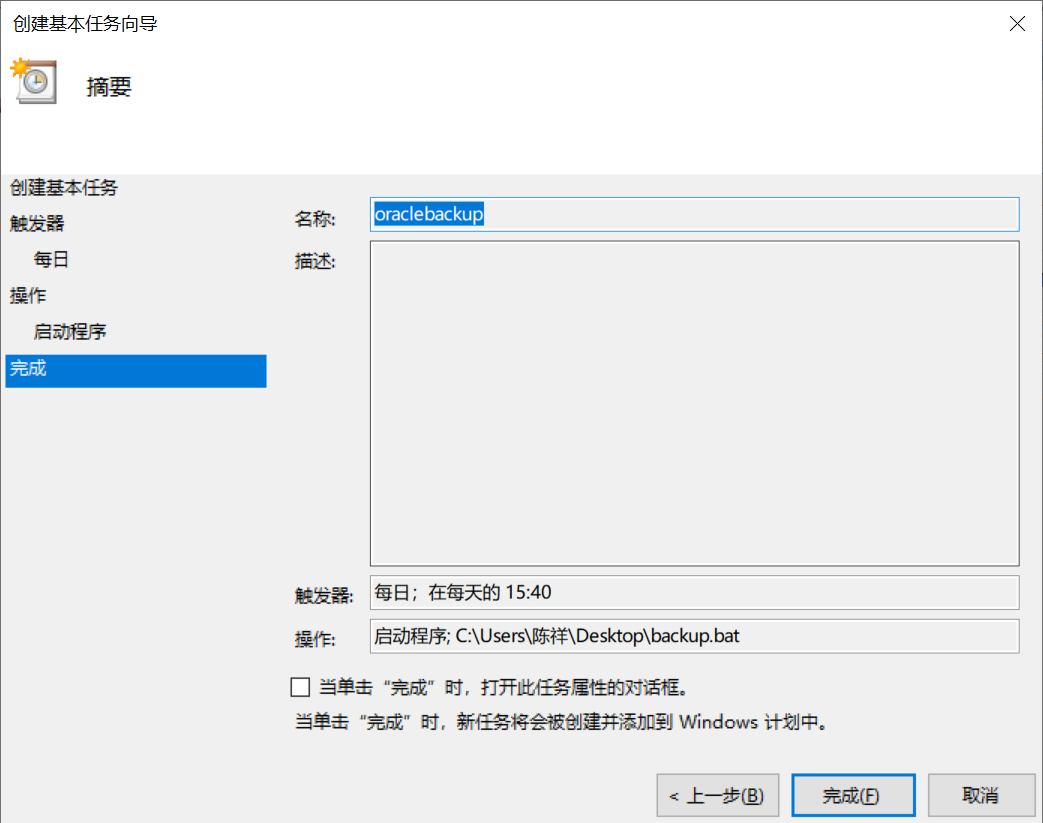












# 7.项目总结

本次大作业对这学期的知识总结以及掌握的考验，在这本次项目中，我设计了一个个人博客的数据库项目，运用到了表及表空间使用方案,创建了2个表空间6张表，插入了7万条数据，在权限设计的用户分配方案阶段，我设计了两类角色，游客和用户，用户有更高级的权限，而游客仅仅能够访问别人的文章，和评论的功能。在关于备份的设计上，我采用了exp命令备份方案，通过编写脚本和进行设置，直接进行自动备份。

本次项目作为所有所学知识点的综合应用，一路上遇到许多的困难，都只能慢慢搜索资料解决，对于我而言还是一个较大的挑战，在备份的设计在经过很多学习后，采用了exp命令方案，按照网上的教程编写脚本，设置任务等。掌握了很多新的知识。完成这个项目也让我收益颇丰，感觉到了完成项目的成就感。