

实验四 数据库常用对象

【实验目的】

掌握索引与视图相关操作，理解索引和视图的作用。

【 实验内容 】

在实验一的基础上完成以下操作。

(1) (建立索引) 为 Score 表按课程号升序、分数降序建立索引, 索引名为 SC_GRADE。

```
create index SC_GRADE on SCORE(CNO , SCORE desc);
```

	INDEX_OWNER	INDEX_NAME	UNIQUENESS	STATUS	INDEX_TYPE	TEMPORARY	PARTITIONED
1	SYSTEM	PK_SCORE	UNIQUE	VALID	NORMAL	N	NO
2	SYSTEM	SC_GRADE	NONUNIQUE	VALID	FUNCTION-BASED NORMAL	N	NO

(2) (删除索引) 删除索引 SC_GRADE。

```
drop index SC_GRADE;
```

	INDEX_OWNER	INDEX_NAME	UNIQUENESS	STATUS	INDEX_TYPE	TEMPORARY	PARTITIONED
1	SYSTEM	PK_SCORE	UNIQUE	VALID	NORMAL	N	NO

(3) (建立视图) 建立计算机系的学生的视图 STUDENT_CS。

```
create view STUDENT_CS
```

as

```
select * from STUDENT
```

where SDEPT = 'CS';

	SNO	SNAME	SDEPT	SCLASS	SSEX	SAGE
1	96001	马小燕	CS	01	女	22
2	96002	黎明	CS	01	男	19
3	97002	李成功	CS	01	男	21
4	97004	李丽	CS	02	女	20
5	96005	司马志明	CS	02	男	19

(4) (建立视图) 建立由学号和平均成绩两个字段的视图 STUDENT_GR。

```
create view STUDENT_GR
```

as

```
select SNO , avg(SCORE) AVG_SCORE from SCORE group by SNO;
```

SNO	AVG_SCORE
1 96001	83.625
2 96002	90.166666666666666666666666666667
3 96003	80
4 96004	87
5 96005	87.25
6 97001	95.5
7 97002	91.5
8 97004	87.5

注意不要雷同

banban

<https://github.com/dream4789/Computer-learning-resources.git>

```
select * from STUDENT_CS where SAGE > 19;
```

(6) (视图查询) 利用视图 STUDENT_GR, 求平均成绩为 88 分以上的学生的学号和平均成绩。

(7) (视图更新) 利用视图 STUDENT_CS, 增加学生('96006', '张然', 'CS', '02', '男', 19)。

(8) (视图更新) 利用视图 STUDENT_CS, 将学生年龄增加 1 岁。观察其运行结果并分析原因。

(9) (视图更新) 利用视图 STUDENT GR, 将平均成绩增加 2 分。观察其运行结果并分析原因。

注意不要雷同

banban

<https://github.com/dream4789/Computer-learning-resources.git>

(10) (视图更新) 删除视图 STUDENT_CS 中学号为 ‘96006’ 的学生的全部数据。

```
delete from STUDENT_CS where SNO = '96006';
```

	SNO	SNAME	SDEPT	SCLASS	SSEX	SAGE
1	96001	马小燕	CS	01	女	23
2	96002	黎明	CS	01	男	20
3	97002	李成功	CS	01	男	22
4	97004	李丽	CS	02	女	21
5	96005	司马志明	CS	02	男	20

(11) (视图更新) 删除视图 STUDENT_GR 的全部数据。

```
delete from STUDENT_GR;
```

错误报告 -
SQL 错误: ORA-01732: 此视图的数据操纵操作非法
01732. 00000 - "data manipulation operation not legal on this view"
*Cause:
*Action:

(12) (删除视图) 删除视图 STUDENT_CS 和 STUDENT_GR。

```
drop view STUDENT_GR;
```

```
drop view STUDENT_CS;
```

View STUDENT_CS已删除。

View STUDENT_GR已删除。

(13) 为 Score 表创建共有同义词 Score_Syn，并通过其查询 Score 表信息。

```
create public synonym Score_Syn for SCORE;
```

SCN_ID	I_TIMESTAMP
SCORE	
SNO	

Public synonym SCORE_SYN 已创建。

	SNO	CNO	SCORE
1	96001	001	77.5
2	96001	003	89
3	96001	004	86
4	96001	005	82
5	96002	001	88
6	96002	003	92.5
7	96002	006	90
8	96005	004	92
9	96005	005	90
10	96005	006	89
11	96005	007	78
12	96003	001	69
13	97001	001	96
14	97001	008	95
15	96004	001	87
16	96003	003	91
17	97002	003	91

(14) 创建表 T1 (No (Number), Name (Varchar2)), 序列 T1_No_Seq (范围: 100000-999999), 并借助 T1_No_Seq 自增主键的方式在表 T1 中插入 3-5 条记录。

注意不要雷同

banban

<https://github.com/dream4789/Computer-learning-resources.git>

```

create table T1
(
    No number(6) primary key,
    Name varchar2(20)
);

```

```

create sequence T1_NO_Seq
increment by 1
start with 100000
maxvalue 999999
cache 50;

```

```

insert into T1 values(T1_NO_Seq.nextval, '张三');
insert into T1 values(T1_NO_Seq.nextval, '李四');
insert into T1 values(T1_NO_Seq.nextval, '王五');
insert into T1 values(T1_NO_Seq.nextval, '王五');
select * from T1;
select * from T1_NO_Seq;

```

Table T1 已创建。

Sequence T1_NO_SEQ 已创建。

	NO	NAME
1	100000	张三
2	100001	李四
3	100002	王五
4	100003	王五