

查询

1. 数据查询

(1) 查询所有列

```
SQL> create table table1  
2 (id int, name varchar2(20));
```

表已创建。

```
SQL> insert into table1  
2 values(1,'张三');
```

已创建 1 行。

```
SQL> select * from table1;
```

ID	NAME
1	张三

(2) 查询特定列

```
SQL> desc table1;  
名称
```

是否为空? 类型

ID	NUMBER(38)
NAME	VARCHAR2(20)

```
SQL> select name from table1;
```

NAME
张三

(3) 排除重复行

```

SQL> insert into table1
  2 values('1','张三');

已创建 1 行。

SQL> insert into table1
  2 values('2','李四');

已创建 1 行。

SQL> select * from table1;

      ID NAME
-----
      1 张三
      1 张三
      2 李四

SQL> select distinct * from table1;

      ID NAME
-----
      2 李四
      1 张三

```

(4) 字符串连接操作

```

SQL> select name ||''''||'s 编号 is ' || id from table1;

NAME||''''||'S编号IS' ||ID
-----
张三's 编号 is 1
张三's 编号 is 1
李四's 编号 is 2

```

注意：连接符号“||”用于连接两个字符串。四个连续的单引号是为了可以输出一个单引号。

(5) 使用列别名

```
SQL> select id*5,name from table1;
```

ID*5	NAME
5	张三
5	张三
10	李四

```
SQL> select id*5 m_id,name from table1;
```

M_ID	NAME
5	张三
5	张三
10	李四

(6) 查询条件

```
SQL> select * from table1
2 where name like '张%';
```

ID	NAME
1	张三
1	张三

```
SQL> insert into table1
2 values(5,'李大仁');
```

已创建 1 行。

```
SQL> insert into table1
2 values(6,'李维嘉');
```

已创建 1 行。

```
SQL> select * from table1;
```

ID	NAME
1	张三
1	张三
2	李四
4	王五
5	李大仁
6	李维嘉

已选择6行。

```
SQL> select * from table1
  2  where name like '李大_';
```

ID	NAME
5	李大仁

```
SQL> select * from table1
  2  where name like '李%';
```

ID	NAME
2	李四
5	李大仁
6	李维嘉

```
SQL> select * from table1
  2  where name like '李__';
```

ID	NAME
5	李大仁
6	李维嘉

2. 多表查询与子查询

(1) 相等连接查询

本实验首先要创建表 `table2`，然后使其中一列的意义与 `table1` 中某列的意义相同，这样就可以实现相等连接查询。

```
SQL> create table table2
  2 (id int, telephone char(15));
```

表已创建。

```
SQL> insert into table2
  2 values(1,'13453172987');
```

已创建 1 行。

```
SQL> insert into table2
  2 values(2,'13041025689');
```

已创建 1 行。

```
SQL> insert into table2
  2 values(3,'13869874236');
```

已创建 1 行。

```
SQL> select * from table2;
```

ID	TELEPHONE
1	13453172987
2	13041025689
3	13869874236

```
SQL> select * from table1;
```

ID	NAME
1	张三
1	张三
2	李四

```
SQL> select * from table2;
```

ID	TELEPHONE
1	13453172987
2	13041025689
3	13869874236

```
SQL> select * from table1;
```

ID	NAME
1	张三
1	张三
2	李四

```
SQL> select table1.name,table2.telephone from table1,table2
2 where table1.id = table2.id;
```

NAME	TELEPHONE
张三	13453172987
张三	13453172987
李四	13041025689

这条语句实现了两张表的相等连接查询，两张表中 id 相同的数据被显示在结果中。

(2) 不等连接查询

```
SQL> insert into table1
2 values(4,'王五');
```

已创建 1 行。

```
SQL> select distinct name,telephone from table1 t1,table2 t2
2 where t1.id in t2.id;
```

NAME	TELEPHONE
张三	13453172987
李四	13041025689

在查询结果中，table1 表中 id 出现在 table2 表的，就会出现在查询结果中。

(3) 子查询

当第一个查询结果被当成第二个查询的一部分出现在第二个查询的条件中时，第一个查询就是子查询。

```

SQL> select * from table1;

      ID NAME
-----
      1 张三
      1 张三
      2 李四
      4 王五

SQL> select * from table2;

      ID TELEPHONE
-----
      1 13453172987
      2 13041025689
      3 13869874236

SQL> select id,name from table1
      2 where id > (select id from table2 where id > 2);

      ID NAME
-----
      4 王五

```

可以看出，子查询的结果为 id=3，因此满足 id>3 的表 table1 中的内容只有 id=4 这行记录。

```

SQL> select id from table1
      2 union
      3 select id from table2
      4 ;

      ID
-----
      1
      2
      3
      4

SQL> select id from table1
      2 intersect
      3 select id from table2
      4 ;

      ID
-----
      1
      2

```

```
SQL> select id from table1
2 union all
3 select id from table2;
```

ID
1
1
2
4
1
2
3

已选择7行。

子查询也可通过 union, intersect, minus, union all 来实现。

```
SQL> select id from table1
2 minus
3 select id from table2;
```

ID
4

```
SQL> select id from table1
2 group by id
3 order by id desc;
```

ID
6
5
4
2
1

3. 使用短语对数据进行完整性控制

(1) 修改已有的表的主键：


```
SQL> alter table table1
  2  add constraint id_pk primary key(id);
add constraint id_pk primary key(id)
```

×

第 2 行出现错误:

ORA-02437: 无法验证 (SYS.ID_PK) - 违反主键

```
SQL> select * from table1;
```

ID	NAME
1	张三
1	张三
2	李四
4	王五
5	李大仁
6	李维嘉

已选择6行。

因为 id 有重复行，主键要唯一。

```
SQL> delete from table1 where id=1;
```

已删除2行。

```
SQL> select * from table1;
```

ID	NAME
2	李四
4	王五
5	李大仁
6	李维嘉

```
SQL> insert into table1
  2  values(1,'张三 ');
```

已创建 1 行。

```
SQL> select * from table1 order by id ;
```

ID	NAME
1	张三
2	李四
4	王五
5	李大仁
6	李维嘉

```
SQL> alter table table1  
2 add constraint id_pk primary key(id);
```

表已更改。

- (2) 创建表时指定主键:

```
SQL> create table table3  
2 (id int not null primary key,  
3 name varchar2(20));
```

表已创建。

- (3) 为表中字段添加唯一值约束:

```
SQL> alter table table1  
2 add constraint name_u unique(name);
```

表已更改。

- (4) 还可以为表中字段设置 check 约束

```
SQL> alter table table1  
2 add constraint id_c check  
3 (id<20);
```

表已更改。

```
SQL> insert into table1  
2 values(21,'sd');  
insert into table1  
*  
第 1 行出现错误:  
ORA-02290: 违反检查约束条件 (SYS.ID_C)
```

```
SQL> insert into table1
  2  values(19,'王王');
```

已创建 1 行。

```
SQL> select * from table1;
```

ID	NAME
2	李四
4	王五
5	李大仁
6	李维嘉
1	张三
19	王王

已选择6行。

补充-修改表的某列名称:

```
alter table table1 rename column dddd to ssss;
```

```
SQL> alter table table1
  2  rename column id to stuID;
```

表已更改。

```
SQL> select * from table1;
```

STUID	NAME
2	李四
4	王五
5	李大仁
6	李维嘉
1	张三