

## 第五章

### 第五章 关于问题

#### 5.4 关于问题的问题

eg: 列出干过多个工作岗位的员工

```
select employee_id, count(*) job_ct from job_history group by employee_id having count(*) > 1;
```

eg: 重写的员工岗位查询

```
select employee_id, count(*) job_ct
from
(
select e.employee_id, e.job_id
from employees e
union all
select j.employee_id, j.job_id
from job_history j
)
group by employee_id
having count(*) > 1;
```

Note: job\_history 只包含历史数据而没有当前的数据, employees 表包含了当前的数据。

#### 5.5 关于数据的问题

eg:

@st-all

eg: 索引聚簇因子

```
select t.table_name||'.'||i.index_name idx_name,
       i.clustering_factor, t.blocks, t.num_rows
from user_indexes i, user_tables t
where i.table_name = t.table_name
       and t.table_name = 'SALES'
order by t.table_name, i.index_name;
```

#### 5.6 建立逻辑表达式

eg:

```
variable empno number
variable getall number
exec :empno := 7369;
exec :getall := 1;
select /* opt1 */ empno, ename from emp
where empno = case when :GetAll <> 1 then :empno else empno end;
@p1n opt1
```

eg:

```
select /* opt2 */ empno, ename from emp
where (:GetAll = 1) OR (empno = :empno);
@p1n opt2
```

eg:

```
exec :getall := 0;
select /* opt3 */ empno, ename from emp
```

```
where empno = CASE WHEN :GetAll <> 1 then :empno ELSE empno END;  
@pln opt3
```

```
eg:  
select /* opt4 */ empno, ename from emp  
where (:GetAll = 1) OR (empno = :empno);  
@pln opt4
```

```
eg:  
select /* opt5 */ empno, ename from emp  
where empno = NVL(:empno, empno);  
@pln opt5
```

```
eg:  
select /* opt6 */ empno, ename from emp  
where (:empno is null) OR (:empno = empno);  
@pln opt6
```

```
eg:  
exec :empno := null;  
select /* opt7 */ empno, ename from emp  
where empno = NVL(:empno, empno);  
@pln opt7
```

```
eg:  
select /* opt8 */ empno, ename from emp  
where (:empno is null) or (:empno = empno);  
@pln opt8
```

```
eg: 使用一个 union all 来处理条件逻辑  
select /* opt9 */ empno, ename from emp  
where :empno is null  
union all  
select empno, ename from emp  
where :empno = empno;  
@pln opt9
```