

实验一结果截图

一、实验截图

(注意截图清晰，包含完整的sql 语句和运行结果)

创建数据库和创建表格




The screenshot displays a database management interface. On the left, a tree view shows the 'employee' database with tables: departments, dept_emp, dept_manager, employees, salaries, and titles. The 'employees' table is currently selected. The main editor area shows the following SQL script:



```
1 • create database employee
2
3
4
5
6
7
8 • create table employee.dept_emp (
9     emp_no int not null,
10    dept_no char(4) not null,
11    from_date date not null,
12    to_date date not null,
13    primary key(emp_no, dept_no)
14 );
15
16 • create table employee.dept_manager (
17     emp_no int not null,
18     dept_no char(4) not null,
19     from_date date not null,
20     to_date date not null,
21     primary key(emp_no, dept_no)
22 );
23
24 • create table employee.employees (
25     emp_no int not null,
26     birth_date date not null,
27     first_name varchar(14) not null,
28     last_name varchar(16) not null,
29     gender enum('M','F') not null,
30     hire_date date not null,
31     primary key(emp_no)
32 );
33
34 • create table employee.salaries (
35     emp_no int not null,
36     salary int not null,
37     from_date date not null,
38     to_date date not null,
```


```

34 • ○ create table employee.salaries (
35     emp_no int not null,
36     salary int not null,
37     from_date date not null,
38     to_date date not null,
39     primary key(emp_no, from_date)
40 );
41
42 • ○ create table employee.titles (
43     emp_no int not null,
44     title varchar(50) not null,
45     from_date date not null,
46     to_date date,
47     primary key(emp_no, title, from_date)
48 );

```


 Table Name: Schema: **employee**
 Charset/Collation: Engine:
 Comments:

Column Name	Datatype	PK	NN	UQ	B	UN
 dept_no	CHAR(4)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
 dept_name	VARCHAR(40)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>


 Table Name: Schema: **employee**
 Charset/Collation: Engine:
 Comments:





Column Name	Datatype	PK	NN	UQ	B	UN	ZF
 emp_no	INT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
 dept_no	CHAR(4)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
 from_date	DATE	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
 to_date	DATE	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>



Table Name: dept_manager Schema: employee

Charset/Collation: utf8mb4 utf8mb4 Engine: InnoDB

Comments:

Column Name	Datatype	PK	NN	UQ	B	UN	Z
emp_no	INT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
dept_no	CHAR(4)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
from_date	DATE	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
to_date	DATE	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>



Table Name: employees Schema: employee

Charset/Collation: utf8mb4 utf8mb4 Engine: InnoDB

Comments:

Column Name	Datatype	PK	NN	UQ	B	UN
emp_no	INT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
birth_date	DATE	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
first_name	VARCHAR(14)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
last_name	VARCHAR(16)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
gender	ENUM('M', 'F')	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
hire_date	DATE	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>



Table Name: salaries Schema: employee

Charset/Collation: utf8mb4 utf8mb4 Engine: InnoDB

Comments:

Column Name	Datatype	PK	NN	UQ	B	UN	Z
emp_no	INT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
salary	INT	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
from_date	DATE	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
to_date	DATE	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>



Table Name: Schema: **employee**

Charset/Collation: Engine:

Comments:

Column Name	Datatype	PK	NN	UQ	B	UN	ZF
emp_no	INT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
title	VARCHAR(50)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
from_date	DATE	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
to_date	DATE	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

完成 14 道题目

1.

```
1 • SELECT e.emp_no,  
2     birth_date,  
3     first_name,  
4     last_name,  
5     gender,  
6     hire_date,  
7     title  
8 FROM employee.employees e,  
9     employee.titles t  
10 WHERE e.emp_no = 10029  
11        AND e.emp_no = t.emp_no;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Contents: |

emp_no	birth_date	first_name	last_name	gender	hire_date	title
10029	1956-12-13	Otmar	Herbst	M	1985-11-20	Engineer
10029	1956-12-13	Otmar	Herbst	M	1985-11-20	Senior Engineer

158 12:46:01 SELECT e.emp_no, birth_date, first_name, last... 2 row(s) returned 0.000 sec / 0.000 sec

2.

```

1 • SELECT first_name, last_name
2 FROM   employee.employees e,
3         employee.departments d,
4         employee.dept_emp de
5 WHERE  e.hire_date > 1990-01-01
6        AND d.dept_name = 'Finance'
7        AND e.gender = 'M'
8        AND e.emp_no = de.emp_no
9        AND d.dept_no = de.dept_no;

```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: |

	first_name	last_name
▶	Yinghua	Dredge
	Premal	Baek
	Ayakannu	Skrikant
	Marla	Brendel
	Chenyi	Syang
	Kazuhito	Encarnacion
	Miyeon	Macedo
	Shrikanth	Mahmud
	Pragnesh	Iisaka
	Shaw	Wendorf
	Shigehito	Kropatsch
	Marc	Hellwagner
	Ortrud	Murillo
	Takahiro	Waterhouse
	Yishai	Cannane
	Kristine	Velardi
	Lucien	Staudhammer
	Moto	Kusakari
	Uli	Keustermans
	Ennio	Trogemann
	Gretta	Baig
	Lein	Lichtman
	Xiaoqiu	Krychniak
	Subbu	Sanella
	Keung	Delgrande
	Drans	Renzini

Result 2 × Read O

Output

Action Output




#	Time	Action	Message
✓ 157	12:39:59	SELECT * FROM employee.departments LIMIT 0, 10...	10 row(s) returned
✓ 158	12:46:01	SELECT e.emp_no, birth_date, first_name, last...	2 row(s) returned
✓ 159	12:47:29	SELECT first_name, last_name FROM employee.emp...	39 row(s) returned

3.

```

1 • SELECT DISTINCT e.emp_no,
2     birth_date,
3     first_name,
4     last_name,
5     gender,
6     hire_date
7 FROM   employee.departments d,
8     employee.dept_emp de,
9     employee.employees e
10 WHERE  e.first_name like 'Ge%'
11        AND e.emp_no not in (
12            SELECT de.emp_no
13            FROM   employee.departments d,
14                employee.dept_emp de
15            WHERE  d.dept_name = 'Production'
16                AND d.dept no = de.dept no);

```

result Grid   Filter Rows: Export:  Wrap Cell Co

emp_no	birth_date	first_name	last_name	gender	hire_date
10001	1953-09-02	Georgi	Facello	M	1986-06-26
10055	1956-06-06	Georgy	Dredge	M	1992-04-27
10219	1952-05-02	Genta	Kolvik	M	1993-03-31

✓ 160 12:47:55 SELECT DISTINCT e.emp_no, birth_date, first_name... 3 row(s) returned 0.015 sec / 0.000 sec

4.

```

1 • SELECT e.emp_no,
2         birth_date,
3         first_name,
4         last_name,
5         gender,
6         hire_date
7 FROM   employee.employees e
8 WHERE  e.first_name in (
9         SELECT first_name
10        FROM   employee.employees e
11        GROUP BY e.first_name HAVING COUNT(*)>3
12      )
13 ORDER BY e.first_name ASC;

```

emp_no	birth_date	first_name	last_name	gender	hire_date
10162	1957-10-05	Florina	Eugenio	M	1991-05-01
10681	1964-12-25	Florina	Garnier	M	1992-10-25
10757	1961-01-11	Florina	Simkin	M	1986-01-06
10850	1963-01-01	Florina	Schapiro	F	1990-04-22
10031	1959-01-27	Karsten	Joslin	M	1991-09-01
10163	1952-09-17	Karsten	Szmurlo	M	1989-07-19
10611	1964-05-11	Karsten	Blokdijk	M	1989-09-16
110228	1958-12-02	Karsten	Sigstam	F	1985-08-04
10214	1962-04-14	Tadahiko	Ciolek	M	1988-02-29
10300	1960-07-12	Tadahiko	Ulpinar	F	1991-05-17
10351	1963-08-23	Tadahiko	Strehl	F	1985-03-07
10791	1954-05-27	Tadahiko	Sgarro	M	1988-07-30
10339	1957-08-27	Teunis	Liedekerke	M	1989-11-30
10663	1965-01-09	Teunis	Noriega	M	1993-01-23
10794	1962-06-26	Teunis	Bertziss	M	1992-03-29
10884	1958-03-07	Teunis	Kleiser	F	1993-04-15
10525	1963-03-29	Yuchang	Lunt	M	1990-07-08
10748	1964-04-12	Yuchang	Francisci	F	1990-08-14
10800	1961-12-15	Yuchang	Matzke	M	1993-05-20
111939	1960-03-25	Yuchang	Weedman	M	1989-07-10
NULL	NULL	NULL	NULL	NULL	NULL

employees 7 x Apply

Output

Action Output

#	Time	Action	Message
✓ 159	12:47:29	SELECT first_name, last_name FROM employee.emp...	39 row(s) returned
✓ 160	12:47:55	SELECT DISTINCT e.emp_no, birth_date, first_name...	3 row(s) returned
✓ 161	12:48:38	SELECT e.emp_no, birth_date, first_name, ...	20 row(s) returned

5.

```

1 • SELECT de.emp_no
2 FROM employee.dept_emp de,
3 employee.departments d
4 WHERE d.dept_name = 'Production'
5 AND de.emp_no in (
6     SELECT de.emp_no
7     FROM employee.dept_emp de,
8     employee.departments d
9     WHERE d.dept_name = 'Quality Management'
10    AND d.dept_no = de.dept_no
11 )
12 AND d.dept_no = de.dept_no;

```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

emp_no
10010
10029
10124
10192
10200
10307
10344
10363
10392
10422
10786
10874

162 12:49:24 SELECT de.emp_no FROM employee.dept_emp de, ... 12 row(s) returned 0.000 sec / 0.000 sec

6.

```

1 • SELECT count(DISTINCT de.emp_no)
2 FROM employee.dept_emp de
3 WHERE de.emp_no in (
4     SELECT de.emp_no
5     FROM employee.dept_emp de
6     GROUP BY de.emp_no HAVING COUNT(*)>=2
7 );

```

< Result Grid | Filter Rows: | Export: | Wrap Cell Content:



count(DISTINCT de.emp_no)
94

163 12:50:16 SELECT count(DISTINCT de.emp_no) FROM employee.dept_emp de 1 row(s) returned 0.000 sec / 0.000 sec

7.

```
1 • SELECT de.emp_no, MAX(salary)
2 FROM employee.salaries s,
3      employee.dept_emp de
4 WHERE de.dept_no = 'd003'
5        AND s.emp_no = de.emp_no
```

<

Result Grid   Filter Rows: | Export



	emp_no	MAX(salary)
▶	10005	96471

✓ 164 12:50:36 SELECT de.emp_no, MAX(salary) FROM employee.s... 1 row(s) returned 0.015 sec / 0.000 sec

8.

```
1 • SELECT first_name, last_name
2 FROM employee.employees e,
3      employee.dept_manager dm
4 WHERE dm.to_date > NOW()
5        AND dm.dept_no = 'd002'
6        AND e.emp_no = dm.emp_no
```

<

Result Grid   Filter Rows: | Export

	first_name	last_name
▶	Isamu	Legleitner



✓ 165 12:51:22 SELECT first_name, last_name FROM employee.empl... 1 row(s) returned 0.015 sec / 0.000 sec

9.

```

1 • SELECT de.dept_no, SUM(s.salary)
2 FROM employee.dept_emp de,
3      employee.salaries s
4 WHERE de.to_date > NOW()
5       AND s.to_date > NOW()
6       AND de.emp_no = s.emp_no
7 GROUP BY de.dept_no

```

Result Grid   Filter Rows: | Ex

	dept_no	SUM(s.salary)
▶	d005	12842935
	d007	10310030
	d004	11799882
	d003	2814474
	d008	3632343
	d006	3321837
	d001	2492789
	d009	3775363
	d002	2949661




✓ 166 12:51:55 SELECT de.dept_no, SUM(s.salary) FROM employee.... 9 row(s) returned 0.015 sec / 0.000 sec

10.

```

1 • SELECT d.dept_no, d.dept_name, AVG(salary)
2 FROM employee.departments d,
3      employee.dept_emp de,
4      employee.salaries s
5 WHERE de.to_date > NOW()
6       AND s.to_date > NOW()
7       AND de.emp_no = s.emp_no
8       AND de.dept_no = d.dept_no
9 GROUP BY de.dept_no
10      HAVING AVG(salary) >= 70000 AND AVG(salary) < 80000;
11

```

Result Grid   Filter Rows: | Export:  | Wrap Cell Conte

	dept_no	dept_name	AVG(salary)
▶	d002	Finance	77622.6579

✓ 167 12:52:36 SELECT d.dept_no, d.dept_name, AVG(salary) FRO... 1 row(s) returned 0.000 sec / 0.000 sec

11.

```

1 • INSERT INTO employee.departments (dept_no, dept_name)
2   VALUES('d180', 'computer');
3
4 • INSERT INTO employee.departments(dept_no, dept_name)
5   VALUES('d110', 'science');

```

✓	173	12:55:01	INSERT INTO employee.departments (dept_no, dept...	1 row(s) affected	0.015 sec
✓	174	12:55:01	INSERT INTO employee.departments(dept_no, dept...	1 row(s) affected	0.000 sec

	dept_no	dept_name
▶	d001	Marketing
	d002	Finance
	d003	Human Resources
	d004	Production
	d005	Development
	d006	Quality Management
	d007	Sales
	d008	Research
	d009	Customer Service
	d110	science
	d180	computer
*	NULL	NULL

12.

```

1 • DELETE FROM employee.departments
2   WHERE dept_no = 'd180'
3   AND dept_name = 'computer';

```

✓	176	12:57:50	DELETE FROM employee.departments WHERE dep...	1 row(s) affected	0.000 sec
---	-----	----------	---	-------------------	-----------

	dept_no	dept_name
▶	d009	Customer Service
	d005	Development
	d002	Finance
	d003	Human Resources
	d001	Marketing
	d004	Production
	d006	Quality Management
	d008	Research
	d007	Sales
	d110	science
*	NULL	NULL

13.

```

1  UPDATE employee.departments
2  SET dept_name = 'computer science'
3  WHERE dept_no = 'd110'

```

✓ 178 12:58:34 UPDATE employee.departments SET dept_name = '... 1 row(s) affected Rows matched: 1 Changed: 1 Wa... 0.015 sec

	dept_no	dept_name
▶	d110	computer science
	d009	Customer Service
	d005	Development
	d002	Finance
	d003	Human Resources
	d001	Marketing
	d004	Production
	d006	Quality Management
	d008	Research
	d007	Sales
*	NULL	NULL

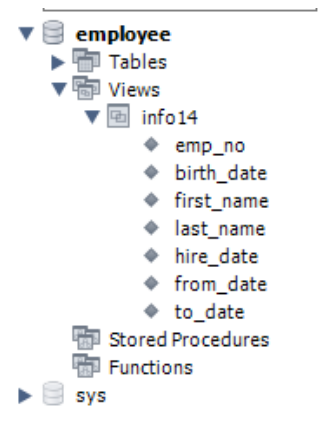
14.

```

1  ● ○ CREATE VIEW info14 AS (
2      SELECT  e.emp_no,
3              birth_date,
4              first_name,
5              last_name,
6              hire_date,
7              de.from_date,
8              de.to_date
9      FROM    employee.employees e,
10             employee.dept_emp de,
11             employee.departments d
12      WHERE   de.from_date >= '1990-01-01'
13              AND d.dept_name = 'Finance'
14              AND e.gender = 'M'
15              AND e.emp_no = de.emp_no
16              AND d.dept_no = de.dept_no
17      );

```

✓ 181 12:59:35 CREATE VIEW info14 AS (SELECT e.emp_no, birt... 0 row(s) affected 0.031 sec



Result Grid							
	emp_no	birth_date	first_name	last_name	hire_date	from_date	to_date
▶	10050	1958-05-21	Yinghua	Dredge	1990-12-25	1990-12-25	1992-11-01
	10080	1957-12-03	Premal	Baek	1985-11-19	1994-09-28	1997-07-01
	10132	1956-12-15	Ayakannu	Skrikant	1994-10-30	1997-06-18	9999-01-01
	10173	1962-10-28	Shrikanth	Mahmud	1992-03-21	1992-03-21	9999-01-01
	10177	1954-11-08	Pragnesh	Iisaka	1993-02-06	1993-02-06	1994-03-01
	10180	1956-01-29	Shaw	Wendorf	1986-02-25	1994-07-01	9999-01-01
	10186	1953-07-16	Shigehito	Kropatsch	1986-03-28	1996-12-25	1997-06-01
	10196	1954-01-27	Marc	Hellwagner	1994-11-16	1999-01-31	9999-01-01
	10263	1961-02-08	Takahiro	Waterhouse	1994-02-05	1996-12-21	9999-01-01
	10272	1954-11-30	Yishai	Cannane	1988-05-23	1997-10-23	9999-01-01
	10295	1953-01-13	Kristine	Velardi	1990-08-27	1995-06-19	9999-01-01
	10354	1961-04-21	Moto	Kusakari	1996-03-26	1999-03-03	2000-03-01
	10459	1959-09-30	Ennio	Trogemann	1997-06-28	1998-06-14	9999-01-01
	10496	1959-04-18	Gretta	Baig	1987-02-26	1995-08-29	1999-01-01
	10543	1962-10-06	Lein	Lichtman	1998-10-16	1998-11-24	9999-01-01
	10576	1957-08-23	Xiaoqiu	Krychniak	1988-05-17	1991-01-03	9999-01-01
	10589	1958-09-25	Subbu	Sanella	1994-01-26	1994-01-26	9999-01-01
	10615	1955-08-30	Keung	Delgrande	1996-03-10	1996-03-10	9999-01-01
	10657	1958-03-09	Juichirou	Kitsuregawa	1989-12-31	1992-03-26	9999-01-01
	10685	1959-01-14	Zongyan	Cusworth	1990-04-24	1995-12-27	1996-10-01
	10743	1963-12-26	Kiyomitsu	Peltason	1985-06-10	1999-11-05	9999-01-01
	10760	1963-05-27	Phillip	Dratva	1994-06-15	1997-08-22	9999-01-01
	10761	1959-04-10	Sumali	Liedekerke	1992-05-30	1992-05-30	1999-01-01
	10806	1958-10-17	Giap	Matzel	1993-04-25	1993-04-25	1994-04-01
	10827	1953-09-25	Willard	Danley	1996-11-07	1996-11-07	9999-01-01
	10855	1957-08-07	Kwangho	Reinhart	1991-08-05	1993-03-29	1997-03-01
	10873	1952-07-09	Sukumar	Wegerle	1994-03-23	1994-03-23	9999-01-01
	10901	1959-06-12	Chrisa	Narahara	1992-10-30	1998-08-05	9999-01-01

Action Output				
#	Time	Action	Message	
3	13:43:17	DROP VIEW 'employee'.info14	0 row(s) affected	
4	13:43:20	CREATE VIEW info14 AS (SELECT e.emp_no, birt...	0 row(s) affected	
5	13:43:25	SELECT * FROM employee.info14 LIMIT 0, 1000	28 row(s) returned	
6	13:44:35	SELECT * FROM employee.info14 LIMIT 0, 1000	28 row(s) returned	

二、思考题

(PPT 中的思考题)

1.

Dept_no 重复：报错主键重复

```
197 13:07:18 INSERT INTO employee.departments (dept_no, dept_name) VALUES(0100, 'computer') 1 row(s) affected 0.031 sec
198 13:07:19 INSERT INTO employee.departments (dept_no, dept_name) VALUES(0100, 'computer') Error Code: 1062, Duplicate entry '0100' for key 'departments.PRIMARY' 0.000 sec
```

Dept_name 重复：报错该列重复

```
199 13:08:45 INSERT INTO employee.departments(dept_no, dept_name) VALUES(0110, 'computer') Error Code: 1062, Duplicate entry 'computer' for key 'departments.dept_name' 0.015 sec
```

满足条件：

该数据中的 emp_no 能在 employees 表中找到元组

dept_no 能在 departments 中找到对应元组

表中不存在 emp_no 和 dept_no 都与新增数据相同的元组

新增数据各项都不为空

机制：

外键 foreign key，dept_emp 中的 emp_no，dept_no 分别为

employees 和 departments 的外键，则 dept_emp 中的外键必定在对应表中存在，且为主键。

2.

分析：

从 query stats 来看，第一种进行的时间消耗较长，第二种较少。

从 execution plan 来看，第一种 tmp table 的行数为 2990，第二种的行数为 930，比第一种大大的减少。

第二种查询把工资判断作为无关子查询，所以只用执行一次，而第一种每一行都要执行一次工资判断。

第一种：

Timing (as measured at client side):

Execution time: 0:00:0.00000000

Timing (as measured by the server):

Execution time: 0:00:0.00298490

Table lock wait time: 0:00:0.00009800

Errors:

Had Errors: NO

Warnings: 0

Rows Processed:

Rows affected: 0

Rows sent to client: 62

Rows examined: 9708

Temporary Tables:

Temporary disk tables created: 0

Temporary tables created: 1

Joins per Type:

Full table scans (Select_scan): 1

Joins using table scans (Select_full_join): 0

Joins using range search (Select_full_range_join): 0

Joins with range checks (Select_range_check): 0

Joins using range (Select_range): 0

Sorting:

Sorted rows (Sort_rows): 0

Sort merge passes (Sort_merge_passes): 0

Sorts with ranges (Sort_range): 0

Sorts with table scans (Sort_scan): 0

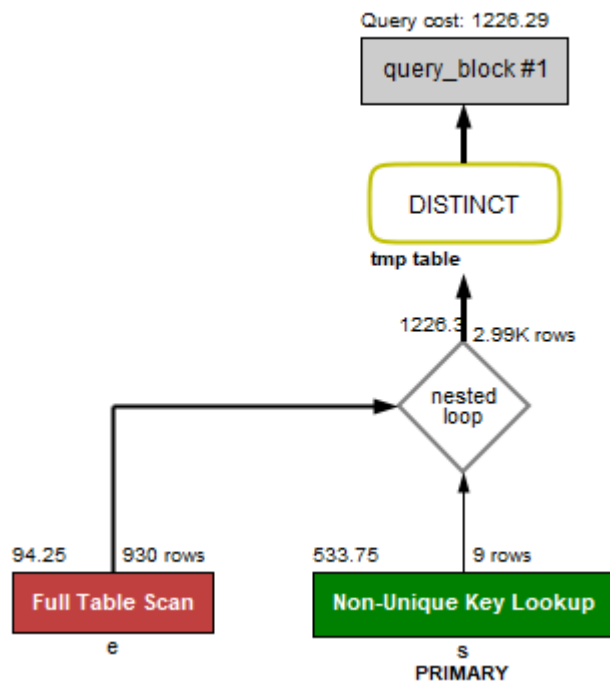
Index Usage:

No Index used

Other Info:

Event Id: 667

Thread Id: 50



第二种:

Timing (as measured at client side):

Execution time: 0:00:0.00000000

Timing (as measured by the server):

Execution time: 0:00:0.00339520

Table lock wait time: 0:00:0.00010800

Errors:

Had Errors: NO

Warnings: 0

Rows Processed:

Rows affected: 0

Rows sent to client: 62

Rows examined: 9708

Temporary Tables:

Temporary disk tables created: 0

Temporary tables created: 1

Joins per Type:

Full table scans (Select_scan): 1

Joins using table scans (Select_full_join): 0

Joins using range search (Select_full_range_join): 0

Joins with range checks (Select_range_check): 0

Joins using range (Select_range): 0

Sorting:

Sorted rows (Sort_rows): 0

Sort merge passes (Sort_merge_passes): 0

Sorts with ranges (Sort_range): 0

Sorts with table scans (Sort_scan): 0

Index Usage:

No Index used

Other Info:

Event Id: 676

Thread Id: 50

