

《数据库系统实验》

实验报告

题目	实验 7
姓名	宋渝杰
学号	18340146
班级	计科六班

一、实验环境：

win10+MySQL 8.0

二、实验内容与完成情况：

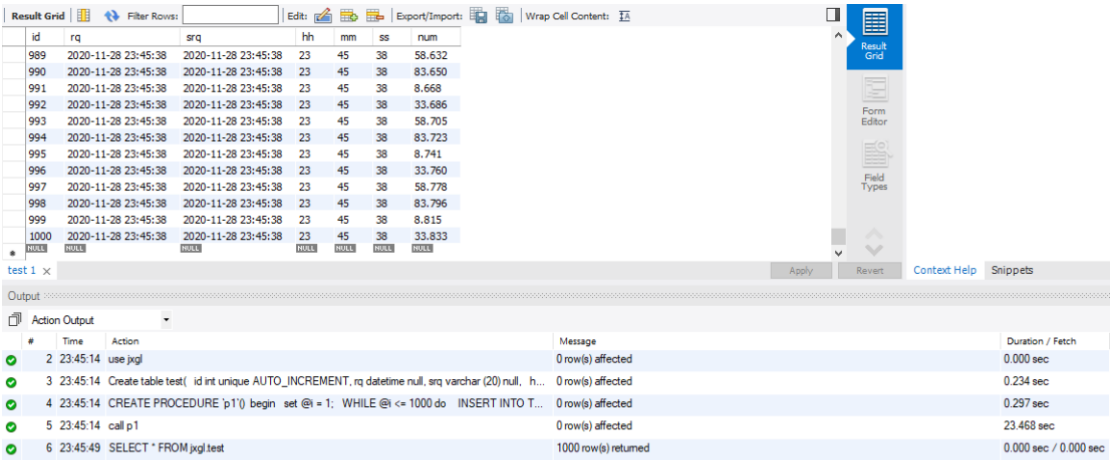
第 109 页“实验内容与要求”第 3 题：

先创建数据库和输入 1000 条测试数据，代码如下：

```
create database jxgl;
use jxgl;
Create table test(
    id int unique AUTO_INCREMENT,
    rq datetime null,
    srq varchar (20) null,
    hh smallint null,
    mm smallint null,
    ss smallint null,
    num numeric (12,3),
    primary key (id)
)AUTO_INCREMENT= 1 engine=MyISAM;
delimiter //
CREATE PROCEDURE `p1`()
begin
    set @i = 1;
    WHILE @i <= 1000 do
        INSERT INTO TEST(RQ,SRQ,HH,MM,SS,NUM)
        VALUES (NOW(),NOW(),HOUR(NOW()),
            MINUTE (NOW()),SECOND(NOW()),RAND(@i) * 100);
        set @i = @i+1;
    END WHILE;
End//
```

delimiter ;

运行结果如下：



id	rq	srq	hh	mm	ss	num
989	2020-11-28 23:45:38	2020-11-28 23:45:38	23	45	38	58.632
990	2020-11-28 23:45:38	2020-11-28 23:45:38	23	45	38	83.650
991	2020-11-28 23:45:38	2020-11-28 23:45:38	23	45	38	8.668
992	2020-11-28 23:45:38	2020-11-28 23:45:38	23	45	38	33.686
993	2020-11-28 23:45:38	2020-11-28 23:45:38	23	45	38	58.705
994	2020-11-28 23:45:38	2020-11-28 23:45:38	23	45	38	83.723
995	2020-11-28 23:45:38	2020-11-28 23:45:38	23	45	38	8.741
996	2020-11-28 23:45:38	2020-11-28 23:45:38	23	45	38	33.760
997	2020-11-28 23:45:38	2020-11-28 23:45:38	23	45	38	58.778
998	2020-11-28 23:45:38	2020-11-28 23:45:38	23	45	38	83.796
999	2020-11-28 23:45:38	2020-11-28 23:45:38	23	45	38	8.815
1000	2020-11-28 23:45:38	2020-11-28 23:45:38	23	45	38	33.833

#	Time	Action	Message	Duration / Fetch
2	23:45:14	use jxgl	0 row(s) affected	0.000 sec
3	23:45:14	Create table test(id int unique AUTO_INCREMENT, rq datetime null, srq varchar (20) null, h...	0 row(s) affected	0.234 sec
4	23:45:14	CREATE PROCEDURE `p1`() begin set @i = 1; WHILE @i <= 1000 do INSERT INTO T...	0 row(s) affected	0.297 sec
5	23:45:14	call p1	0 row(s) affected	23.468 sec
6	23:45:49	SELECT * FROM jxgl.test	1000 row(s) returned	0.000 sec / 0.000 sec

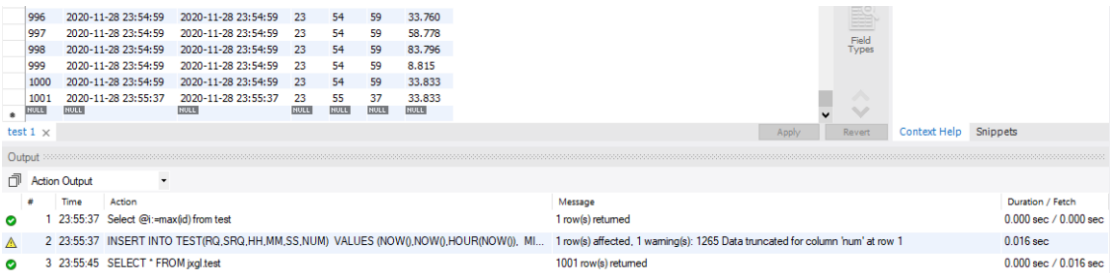
一. 未建立索引时的操作

(1) 单记录插入

代码：

```
Select @i:=max(id) from test;  
INSERT INTO TEST(RQ,SRQ,HH,MM,SS,NUM)  
VALUES (NOW(),NOW(),HOUR(NOW()),  
MINUTE (NOW()),SECOND(NOW()),RAND(@i) * 100);
```

结果：



id	rq	srq	hh	mm	ss	num
996	2020-11-28 23:54:59	2020-11-28 23:54:59	23	54	59	33.760
997	2020-11-28 23:54:59	2020-11-28 23:54:59	23	54	59	58.778
998	2020-11-28 23:54:59	2020-11-28 23:54:59	23	54	59	83.796
999	2020-11-28 23:54:59	2020-11-28 23:54:59	23	54	59	8.815
1000	2020-11-28 23:54:59	2020-11-28 23:54:59	23	54	59	33.833
1001	2020-11-28 23:55:37	2020-11-28 23:55:37	23	55	37	33.833

#	Time	Action	Message	Duration / Fetch
1	23:55:37	Select @i:=max(id) from test	1 row(s) returned	0.000 sec / 0.000 sec
2	23:55:37	INSERT INTO TEST(RQ,SRQ,HH,MM,SS,NUM) VALUES (NOW(),NOW(),HOUR(NOW()), MI...	1 row(s) affected, 1 warning(s): 1265 Data truncated for column 'num' at row 1	0.016 sec
3	23:55:45	SELECT * FROM jxgl.test	1001 row(s) returned	0.000 sec / 0.016 sec

(2) 查询所有记录，按 id 排序

代码：

```
select* from test order by id;
```

结果：

	id	rq	srq	hh	mm	ss	num
▶	1	2020-11-28 23:54:36	2020-11-28 23:54:36	23	54	36	40.540
	2	2020-11-28 23:54:36	2020-11-28 23:54:36	23	54	36	65.559
	3	2020-11-28 23:54:36	2020-11-28 23:54:36	23	54	36	90.577
	4	2020-11-28 23:54:36	2020-11-28 23:54:36	23	54	36	15.595
	5	2020-11-28 23:54:36	2020-11-28 23:54:36	23	54	36	40.614
	6	2020-11-28 23:54:36	2020-11-28 23:54:36	23	54	36	65.632
	7	2020-11-28 23:54:36	2020-11-28 23:54:36	23	54	36	90.650
	8	2020-11-28 23:54:36	2020-11-28 23:54:36	23	54	36	15.669
	9	2020-11-28 23:54:36	2020-11-28 23:54:36	23	54	36	40.687
	10	2020-11-28 23:54:36	2020-11-28 23:54:36	23	54	36	65.705
	11	2020-11-28 23:54:36	2020-11-28 23:54:36	23	54	36	90.723
	12	2020-11-28 23:54:36	2020-11-28 23:54:36	23	54	36	15.742
	13	2020-11-28 23:54:36	2020-11-28 23:54:36	23	54	36	40.760

(3) 查询所有记录，按 mm 排序

代码：

```
select* from test order by mm;
```

结果：因为每个数据的 mm 都一样，因此结果也和上图相同

	id	rq	srq	hh	mm	ss	num
▶	1	2020-11-28 23:54:36	2020-11-28 23:54:36	23	54	36	40.540
	2	2020-11-28 23:54:36	2020-11-28 23:54:36	23	54	36	65.559
	3	2020-11-28 23:54:36	2020-11-28 23:54:36	23	54	36	90.577
	4	2020-11-28 23:54:36	2020-11-28 23:54:36	23	54	36	15.595
	5	2020-11-28 23:54:36	2020-11-28 23:54:36	23	54	36	40.614
	6	2020-11-28 23:54:36	2020-11-28 23:54:36	23	54	36	65.632
	7	2020-11-28 23:54:36	2020-11-28 23:54:36	23	54	36	90.650
	8	2020-11-28 23:54:36	2020-11-28 23:54:36	23	54	36	15.669
	9	2020-11-28 23:54:36	2020-11-28 23:54:36	23	54	36	40.687
	10	2020-11-28 23:54:36	2020-11-28 23:54:36	23	54	36	65.705
	11	2020-11-28 23:54:36	2020-11-28 23:54:36	23	54	36	90.723
	12	2020-11-28 23:54:36	2020-11-28 23:54:36	23	54	36	15.742
	13	2020-11-28 23:54:36	2020-11-28 23:54:36	23	54	36	40.760

(4) 单记录查询

代码：

```
select id from test where id=51;
```

结果：

(3) 查询所有记录，按 id 排序

代码：

```
select* from test order by id;
```

结果：

	id	rq	srq	hh	mm	ss	num
▶	1	2020-11-28 23:54:36	2020-11-28 23:54:36	23	54	36	40.540
	2	2020-11-28 23:54:36	2020-11-28 23:54:36	23	54	36	65.559
	3	2020-11-28 23:54:36	2020-11-28 23:54:36	23	54	36	90.577
	4	2020-11-28 23:54:36	2020-11-28 23:54:36	23	54	36	15.595
	5	2020-11-28 23:54:36	2020-11-28 23:54:36	23	54	36	40.614
	6	2020-11-28 23:54:36	2020-11-28 23:54:36	23	54	36	65.632
	7	2020-11-28 23:54:36	2020-11-28 23:54:36	23	54	36	90.650
	8	2020-11-28 23:54:36	2020-11-28 23:54:36	23	54	36	15.669
	9	2020-11-28 23:54:36	2020-11-28 23:54:36	23	54	36	40.687
	10	2020-11-28 23:54:36	2020-11-28 23:54:36	23	54	36	65.705
	11	2020-11-28 23:54:36	2020-11-28 23:54:36	23	54	36	90.723
	12	2020-11-28 23:54:36	2020-11-28 23:54:36	23	54	36	15.742
	13	2020-11-28 23:54:36	2020-11-28 23:54:36	23	54	36	40.760

(4) 查询所有记录，按 mm 排序

代码：

```
select* from test order by mm;
```

结果：

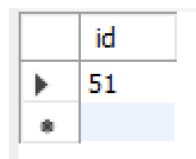
	id	rq	srq	hh	mm	ss	num
▶	1002	2020-11-29 00:07:37	2020-11-29 00:07:37	0	7	37	58.851
	1	2020-11-28 23:54:36	2020-11-28 23:54:36	23	54	36	40.540
	2	2020-11-28 23:54:36	2020-11-28 23:54:36	23	54	36	65.559
	3	2020-11-28 23:54:36	2020-11-28 23:54:36	23	54	36	90.577
	4	2020-11-28 23:54:36	2020-11-28 23:54:36	23	54	36	15.595
	5	2020-11-28 23:54:36	2020-11-28 23:54:36	23	54	36	40.614
	6	2020-11-28 23:54:36	2020-11-28 23:54:36	23	54	36	65.632
	7	2020-11-28 23:54:36	2020-11-28 23:54:36	23	54	36	90.650
	8	2020-11-28 23:54:36	2020-11-28 23:54:36	23	54	36	15.669
	9	2020-11-28 23:54:36	2020-11-28 23:54:36	23	54	36	40.687
	10	2020-11-28 23:54:36	2020-11-28 23:54:36	23	54	36	65.705
	11	2020-11-28 23:54:36	2020-11-28 23:54:36	23	54	36	90.723
	12	2020-11-28 23:54:36	2020-11-28 23:54:36	23	54	36	15.742

(5) 单记录查询

代码：

```
select id from test where id=51;
```

结果：



	id
▶	51
✱	

耗时对比：

插入：0.016s

按 id 排序：0.016s

按 mm 排序：0.000s

查询：0.000s

三、实验心得：

本次实验内容比较多，难度略大，而且一个奇怪的点是，一开始按照课本插入 80000 条数据，但是总会产生超时报错，多次测试之后发现超过 30s 就会停止，而在我的设备上插入 1000 条需要 23s，可能是设备的问题，因此在本次实验中也只能插入 1000 条数据进行数据的初始化和之后的实验。