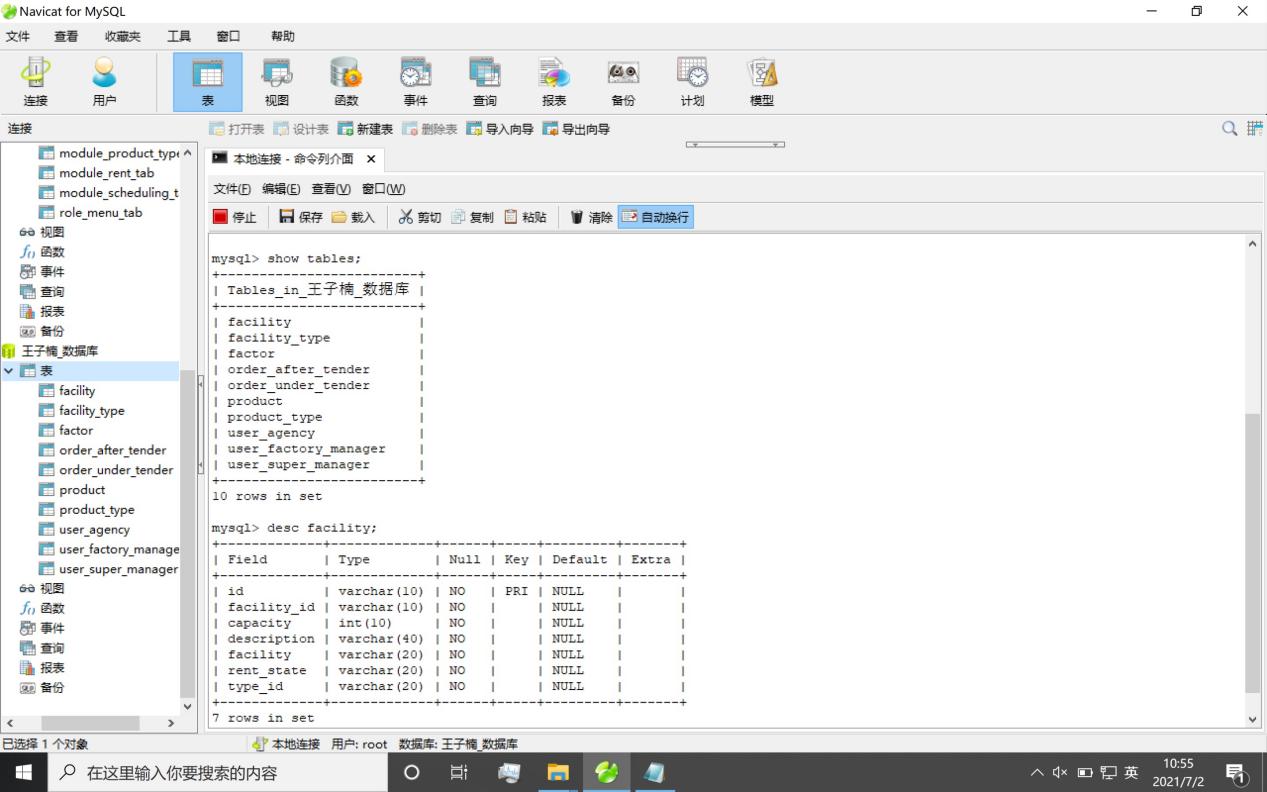
DDL: 针对自己设计的数据表结构写出创建表的SQL语句以及运行截图。 选择某个表对其进行列的增加、删除、修改等操作，写出SQL以及运行截图。

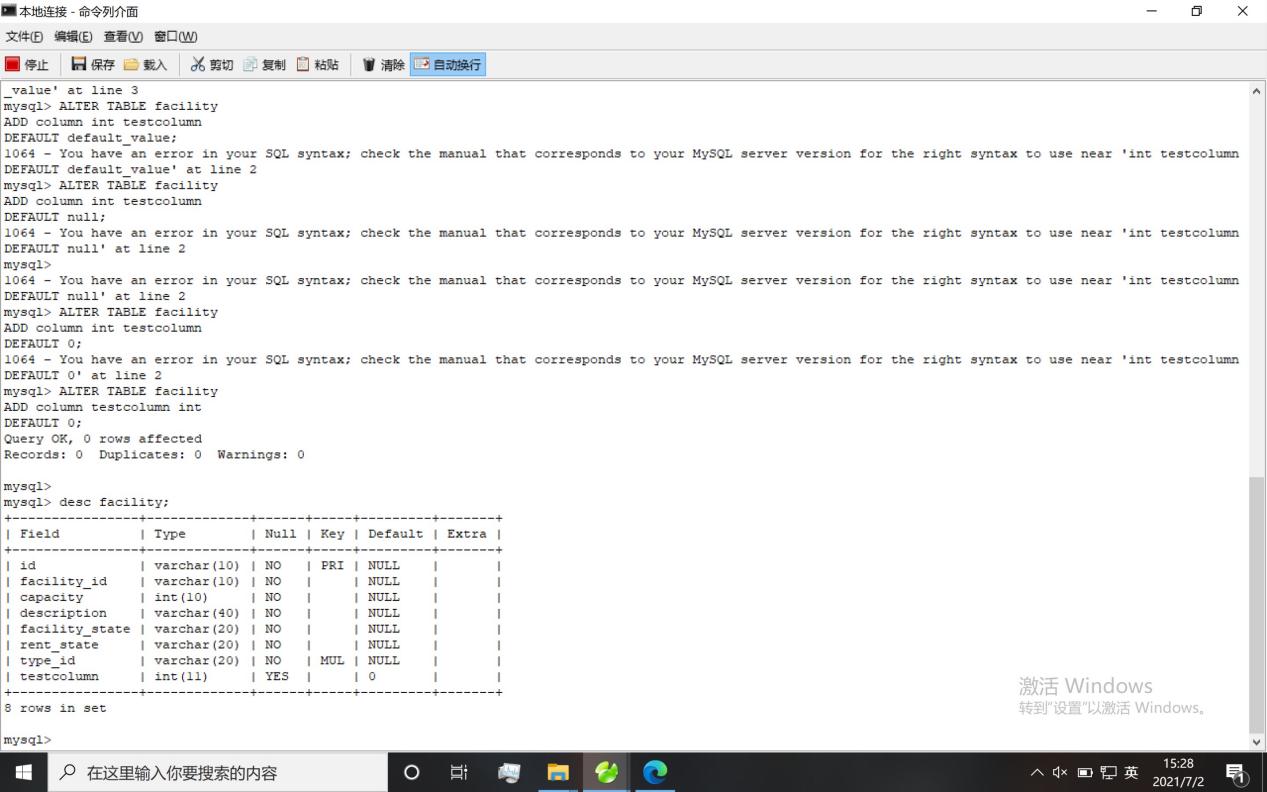
1. 创建表：
   1. sql语句：
2. */\**
3. Navicat MySQL Data Transfer
4. Source Server         : 本地连接
5. Source Server Version : 80016
6. Source Host           : localhost:3306
7. Source Database       : 王子楠\_数据库
8. Target Server Type    : MYSQL
9. Target Server Version : 80016
10. File Encoding         : 65001
11. Date: 2021-07-02 11:30:42
12. \*/
13. SET FOREIGN\_KEY\_CHECKS=0;
14. *-- ----------------------------*
15. *-- Table structure for `facility`*
16. *-- ----------------------------*
17. DROP TABLE IF EXISTS `facility`;
18. CREATE TABLE `facility` (
19. `id` varchar(10) NOT NULL,
20. `facility\_id` varchar(10) NOT NULL,
21. `capacity` int(10) NOT NULL,
22. `description` varchar(40) NOT NULL,
23. `facility\_state` varchar(20) CHARACTER SET utf8 COLLATE utf8\_general\_ci NOT NULL,
24. `rent\_state` varchar(20) NOT NULL,
25. `type\_id` varchar(20) NOT NULL,
26. PRIMARY KEY (`id`),
27. KEY `type\_id` (`type\_id`),
28. CONSTRAINT `type\_id` FOREIGN KEY (`type\_id`) REFERENCES `facility\_type` (`id`)
29. ) ENGINE=InnoDB DEFAULT CHARSET=utf8;
30. *-- ----------------------------*
31. *-- Records of facility*
32. *-- ----------------------------*
33. *-- ----------------------------*
34. *-- Table structure for `facility\_type`*
35. *-- ----------------------------*
36. DROP TABLE IF EXISTS `facility\_type`;
37. CREATE TABLE `facility\_type` (
38. `id` varchar(10) NOT NULL,
39. `fac\_type` varchar(20) NOT NULL,
40. PRIMARY KEY (`id`)
41. ) ENGINE=InnoDB DEFAULT CHARSET=utf8;
42. *-- ----------------------------*
43. *-- Records of facility\_type*
44. *-- ----------------------------*
45. *-- ----------------------------*
46. *-- Table structure for `factor`*
47. *-- ----------------------------*
48. DROP TABLE IF EXISTS `factor`;
49. CREATE TABLE `factor` (
50. `id` varchar(10) NOT NULL,
51. `factor\_id` varchar(10) NOT NULL,
52. `name` varchar(40) NOT NULL,
53. `capacity` int(10) NOT NULL,
54. `factor\_state` varchar(20) NOT NULL,
55. PRIMARY KEY (`id`)
56. ) ENGINE=InnoDB DEFAULT CHARSET=utf8;
57. *-- ----------------------------*
58. *-- Records of factor*
59. *-- ----------------------------*
60. *-- ----------------------------*
61. *-- Table structure for `order\_after\_tender`*
62. *-- ----------------------------*
63. DROP TABLE IF EXISTS `order\_after\_tender`;
64. CREATE TABLE `order\_after\_tender` (
65. `id` varchar(20) CHARACTER SET utf8 COLLATE utf8\_general\_ci NOT NULL,
66. `name` varchar(20) NOT NULL,
67. `number` int(10) NOT NULL,
68. `del\_date` datetime NOT NULL,
69. `state` varchar(20) NOT NULL,
70. `rec\_id` varchar(10) NOT NULL,
71. `rec\_place` varchar(40) NOT NULL,
72. `rec\_mobile` varchar(20) NOT NULL,
73. PRIMARY KEY (`id`),
74. KEY `rec\_id\_after` (`rec\_id`),
75. KEY `prod\_name\_after` (`name`),
76. CONSTRAINT `prod\_name\_after` FOREIGN KEY (`name`) REFERENCES `product` (`name`),
77. CONSTRAINT `rec\_id\_after` FOREIGN KEY (`rec\_id`) REFERENCES `user\_agency` (`account`) ON DELETE RESTRICT ON UPDATE RESTRICT
78. ) ENGINE=InnoDB DEFAULT CHARSET=utf8;
79. *-- ----------------------------*
80. *-- Records of order\_after\_tender*
81. *-- ----------------------------*
82. *-- ----------------------------*
83. *-- Table structure for `order\_under\_tender`*
84. *-- ----------------------------*
85. DROP TABLE IF EXISTS `order\_under\_tender`;
86. CREATE TABLE `order\_under\_tender` (
87. `id` varchar(10) NOT NULL,
88. `name` varchar(20) NOT NULL,
89. `number` int(10) NOT NULL,
90. `del\_date` datetime NOT NULL,
91. `tend\_date` datetime NOT NULL,
92. `state` varchar(20) NOT NULL,
93. `rec\_id` varchar(10) NOT NULL,
94. `rec\_place` varchar(40) NOT NULL,
95. `rec\_mobile` varchar(20) NOT NULL,
96. PRIMARY KEY (`id`),
97. KEY `rec\_id\_under` (`rec\_id`),
98. KEY `prod\_name\_under` (`name`),
99. CONSTRAINT `prod\_name\_under` FOREIGN KEY (`name`) REFERENCES `product` (`name`),
100. CONSTRAINT `rec\_id\_under` FOREIGN KEY (`rec\_id`) REFERENCES `user\_agency` (`account`)
101. ) ENGINE=InnoDB DEFAULT CHARSET=utf8;
102. *-- ----------------------------*
103. *-- Records of order\_under\_tender*
104. *-- ----------------------------*
105. *-- ----------------------------*
106. *-- Table structure for `product`*
107. *-- ----------------------------*
108. DROP TABLE IF EXISTS `product`;
109. CREATE TABLE `product` (
110. `id` varchar(10) NOT NULL,
111. `prod\_id` varchar(10) NOT NULL,
112. `name` varchar(20) NOT NULL,
113. `type` varchar(10) NOT NULL,
114. `specification` varchar(20) NOT NULL,
115. `description` varchar(20) NOT NULL,
116. PRIMARY KEY (`id`),
117. KEY `type\_id\_product` (`type`),
118. KEY `name` (`name`),
119. CONSTRAINT `type\_id\_product` FOREIGN KEY (`type`) REFERENCES `product\_type` (`id`)
120. ) ENGINE=InnoDB DEFAULT CHARSET=utf8;
121. *-- ----------------------------*
122. *-- Records of product*
123. *-- ----------------------------*
124. *-- ----------------------------*
125. *-- Table structure for `product\_type`*
126. *-- ----------------------------*
127. DROP TABLE IF EXISTS `product\_type`;
128. CREATE TABLE `product\_type` (
129. `id` varchar(10) NOT NULL,
130. `prod\_id` varchar(20) NOT NULL,
131. KEY `id` (`id`)
132. ) ENGINE=InnoDB DEFAULT CHARSET=utf8;
133. *-- ----------------------------*
134. *-- Records of product\_type*
135. *-- ----------------------------*
136. *-- ----------------------------*
137. *-- Table structure for `user\_agency`*
138. *-- ----------------------------*
139. DROP TABLE IF EXISTS `user\_agency`;
140. CREATE TABLE `user\_agency` (
141. `account` varchar(10) CHARACTER SET utf8 COLLATE utf8\_general\_ci NOT NULL,
142. `password` varchar(50) NOT NULL,
143. `name` varchar(20) NOT NULL,
144. `mobile` varchar(20) DEFAULT NULL,
145. `type` int(10) NOT NULL,
146. PRIMARY KEY (`account`)
147. ) ENGINE=InnoDB DEFAULT CHARSET=utf8;
148. *-- ----------------------------*
149. *-- Records of user\_agency*
150. *-- ----------------------------*
151. *-- ----------------------------*
152. *-- Table structure for `user\_factory\_manager`*
153. *-- ----------------------------*
154. DROP TABLE IF EXISTS `user\_factory\_manager`;
155. CREATE TABLE `user\_factory\_manager` (
156. `account` varchar(20) NOT NULL,
157. `password` varchar(50) NOT NULL,
158. `name` varchar(20) NOT NULL,
159. `mobile` varchar(20) DEFAULT NULL,
160. `type` int(10) NOT NULL,
161. PRIMARY KEY (`account`)
162. ) ENGINE=InnoDB DEFAULT CHARSET=utf8;
163. *-- ----------------------------*
164. *-- Records of user\_factory\_manager*
165. *-- ----------------------------*
166. *-- ----------------------------*
167. *-- Table structure for `user\_super\_manager`*
168. *-- ----------------------------*
169. DROP TABLE IF EXISTS `user\_super\_manager`;
170. CREATE TABLE `user\_super\_manager` (
171. `account` varchar(20) NOT NULL,
172. `password` varchar(50) NOT NULL,
173. `name` varchar(20) NOT NULL,
174. `mobile` varchar(20) DEFAULT NULL,
175. `type` int(10) NOT NULL,
176. PRIMARY KEY (`account`)
177. ) ENGINE=InnoDB DEFAULT CHARSET=utf8;
178. *-- ----------------------------*
179. *-- Records of user\_super\_manager*
180. *-- ----------------------------*
     1. 运行截图：



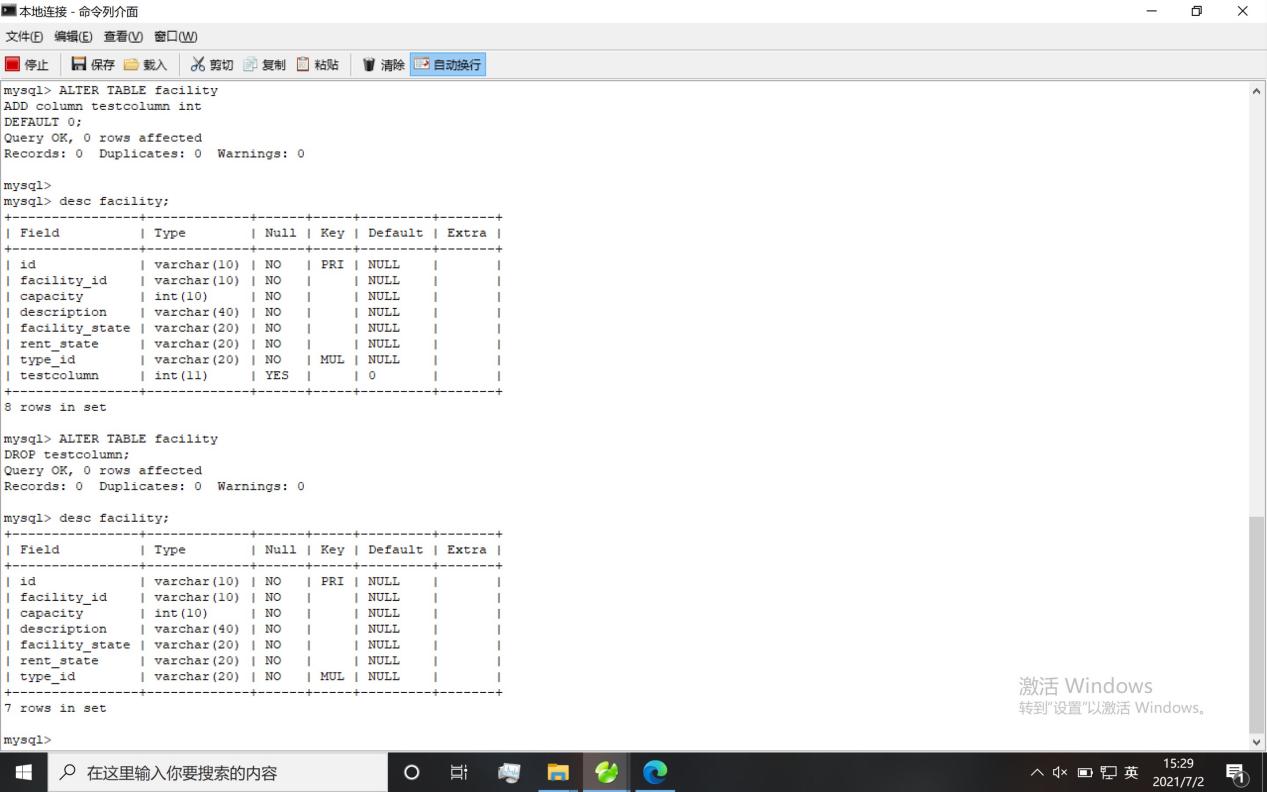
1. 修改表

选择facility对其进行列的增加、删除、修改等操作

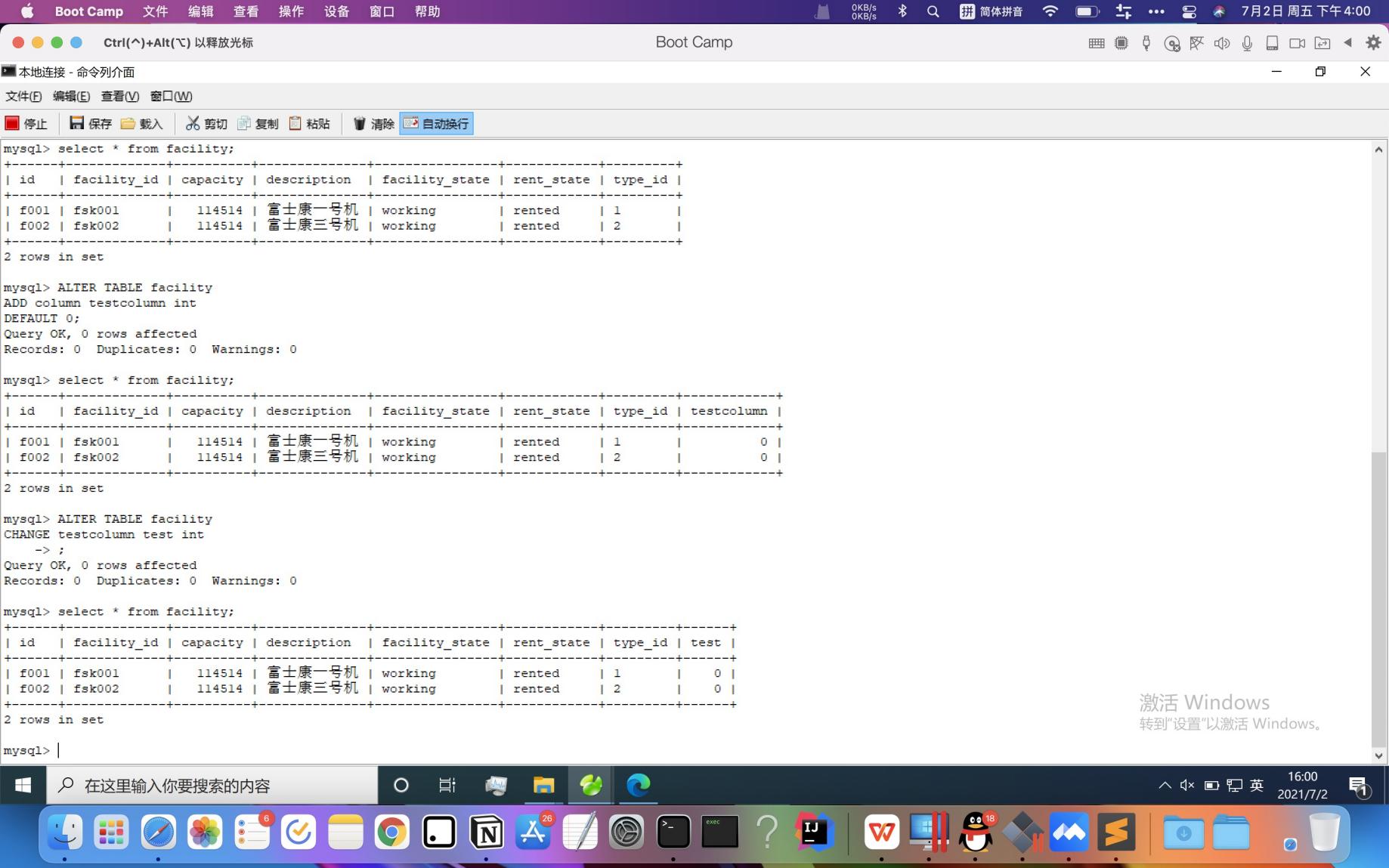
* + - 1. 增加列：



* + - 1. 删除列：



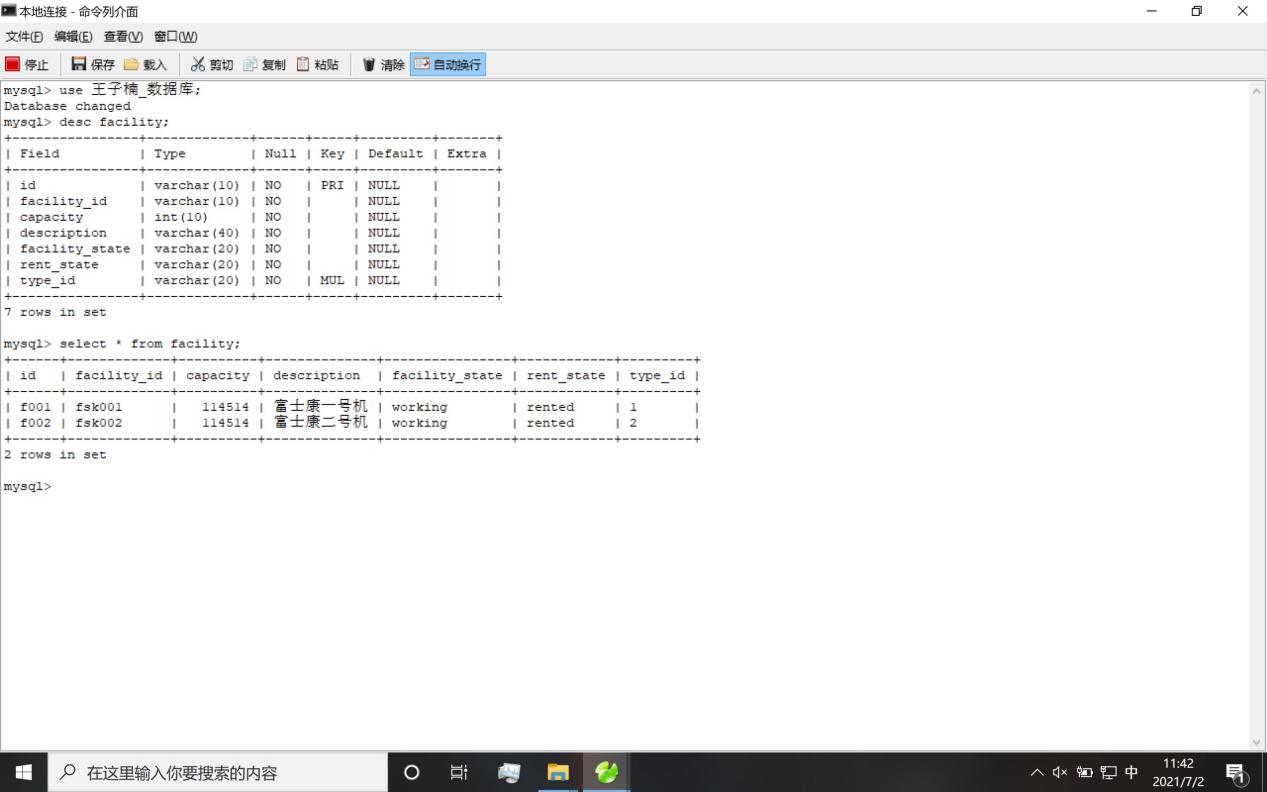
* + 1. 修改列



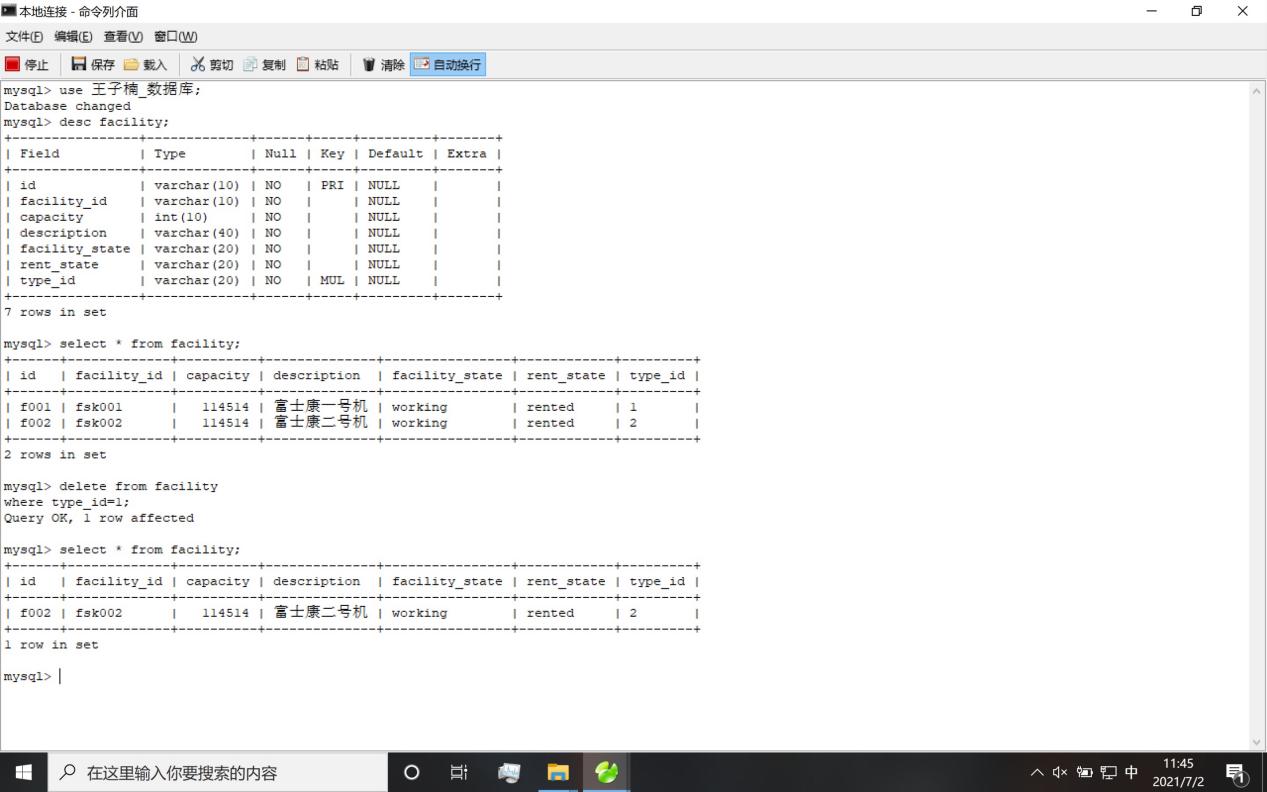
DML 在数据表中手动添加合理的数据。 选择某个表对其进行添加数据操作(使用多种方式添加数据) 删除数据以及修改数据操作，写出SQL以及运行截图

选择facility对其进数据的增加、删除、修改等操作

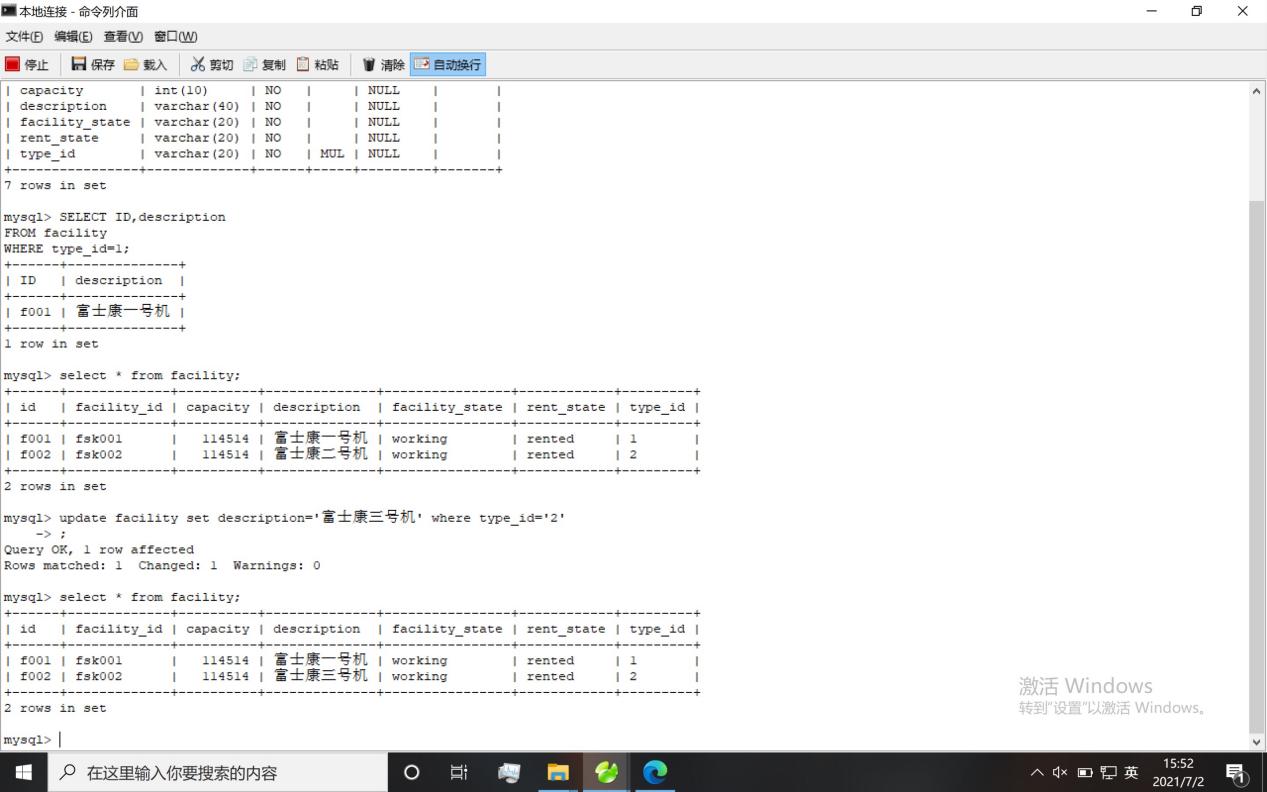
1. 增加



2. 删除



1. 修改



DQL（以下练习如果自己的表符合可以用自己的SQL，如果不符合请使用提供的SQL） 条件查询: 1.条件查询就是在查询时给出WHERE子句，在WHERE子句中可以使用一些运算符及关键字。 2.条件查询运行符及关键字：=(等于) 、 !=(不等于) 、<>(不等于) 、<(小于)、<=(小于等于)、 >(大于) 、>=(大于等于) 自选一张表进行条件查询，写出SQL以及运行截图。 请写出针对以下问题的SQL语句（每一问必须用一条SQL语句实现，但该SQL语句可以包含子查询）。

1. 查询所有空闲状态的设备信息；
2. SELECT ID,devicename
3. FROM module\_device\_tab
4. WHERE devicestatus = 2;
5. 给定工厂名称，列出所有该工厂租用的设备；
6. SELECT
7. module\_rent\_tab.deviceid
8. FROM
9. module\_rent\_tab natural join module\_factory\_tab
10. WHERE
11. module\_factory\_tab.factoryname = "云工厂1" and
12. module\_rent\_tab.factoryid=module\_factory\_tab.ID;
13. 根据设备ID查询设备基本信息；
14. SELECT \*
15. FROM module\_device\_tab
16. WHERE ID='2135224e4e624aa38dc011d5c316187a';

4. 根据设备ID查询此设备所有租赁记录信息；

1. SELECT ID,begindate,enddate,length,deviceid,factoryid
2. FROM module\_rent\_tab
3. WHERE deviceid='2135224e4e624aa38dc011d5c316187a';
4. 根据设备ID查询设备当前的产能信息，包括产品名称，产品类型，设备类型，设备名称，租用工厂名称；
5. SELECT
6. module\_product\_tab.productname,
7. module\_product\_type\_tab.typename,
8. module\_device\_type\_tab.typename,
9. module\_device\_tab.devicename,
10. module\_factory\_tab.factoryname
11. FROM
12. module\_scheduling\_tab natural join
13. module\_product\_tab natural join
14. module\_product\_type\_tab natural join
15. module\_device\_tab natural join
16. module\_device\_type\_tab natural join
17. module\_device\_product\_tab natural join
18. module\_factory\_tab
19. WHERE
20. module\_device\_product\_tab.deviceid='2135224e4e624aa38dc011d5c316187a' and
21. module\_product\_type\_tab.id=module\_product\_tab.typeid and
22. module\_device\_product\_tab.deviceid=module\_device\_tab.ID and
23. module\_device\_tab.typeid=module\_device\_type\_tab.ID and
24. module\_device\_product\_tab.factoryid=module\_factory\_tab.id;
25. 根据工厂名称查询工厂自由设备基本信息，包括设备类型，设备名称；
26. SELECT
27. module\_device\_tab.factoryid,
28. module\_factory\_tab.factoryname,
29. module\_device\_type\_tab.typename,
30. module\_device\_tab.devicename
31. FROM
32. module\_device\_tab natural join
33. module\_factory\_tab natural join
34. module\_device\_type\_tab
35. WHERE
36. factoryname='云工厂1' and
37. rentstatus = 0 and
38. module\_device\_tab.typeid=module\_device\_type\_tab.ID and
39. module\_device\_tab.factoryid=module\_factory\_tab.ID;
40. 根据租用ID,查询设备产能信息，包括设备id，设备名称，设备类型，产品类型，产品名称，产能；
41. SELECT
42. module\_rent\_tab.deviceid,
43. module\_device\_tab.devicename,
44. module\_device\_type\_tab.typename,
45. module\_product\_type\_tab.typename,
46. module\_product\_tab.productname,
47. module\_device\_product\_tab.capacity
48. FROM
49. module\_device\_tab natural join
50. module\_device\_type\_tab natural join
51. module\_product\_tab natural join
52. module\_rent\_tab natural join
53. module\_product\_type\_tab natural join
54. module\_device\_product\_tab
55. WHERE
56. module\_rent\_tab.ID='2135224e4e624aa38dc011d5c316187a' and
57. module\_rent\_tab.deviceid=module\_device\_tab.id and
58. module\_device\_tab.typeid=module\_device\_type\_tab.ID and
59. module\_device\_product\_tab.deviceid=module\_device\_tab.id and
60. module\_device\_product\_tab.productid=module\_product\_tab.id and
61. module\_product\_tab.typeid=module\_product\_type\_tab.id;
62. 根据工厂ID，查询该工厂下，自由设备且未配置产能的设备信息
63. SELECT
64. module\_device\_tab.ID,
65. module\_device\_tab.devicename,
66. module\_device\_tab.describe
67. FROM
68. module\_factory\_tab natural join
69. module\_device\_tab
70. WHERE
71. module\_factory\_tab.ID="4a4a8bed08c54566a226eed2b090308b" and
72. module\_device\_tab.factoryid=module\_factory\_tab.ID and
73. rentstatus=0 and
74. devicestatus<>1;
75. 根据产品名称，查询整个系统该产品的产能情况。
76. SELECT
77. sum(module\_device\_product\_tab.capacity)
78. FROM
79. module\_product\_tab natural join
80. module\_device\_product\_tab
81. WHERE
82. module\_product\_tab.productname='Iphone xs手机壳3' and
83. module\_device\_product\_tab.productid=module\_product\_tab.id;
84. 查询所有产能中心中未被租用且正常运行的设备
85. SELECT
86. devicename
87. FROM
88. module\_device\_tab
89. WHERE
90. module\_device\_tab.devicestatus=1 and
91. module\_device\_tab.rentstatus<>1;