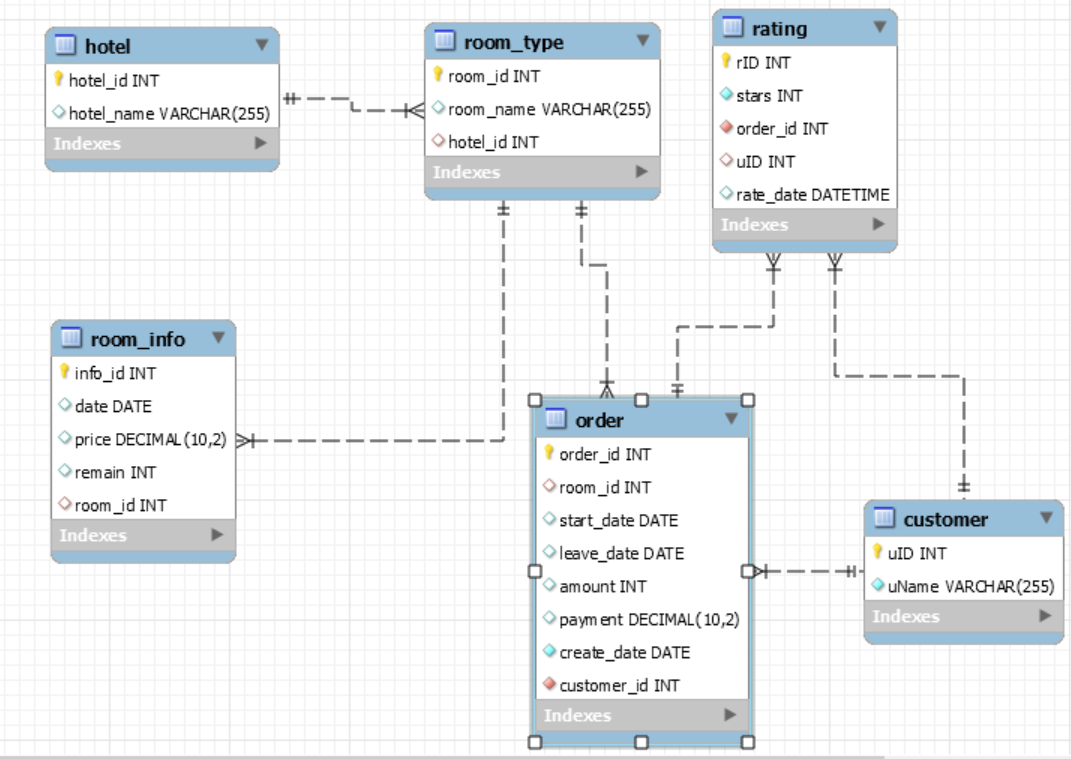
* customer(uID,uName)
* hotel(hotel\_id,hotel\_name)
* order(order\_id,room\_id,start\_date,leave\_date,amount,payment,create\_date,custom\_id)
* rating(rID,stars,order\_id,uID,rate\_date)
* room\_info(info\_id,date,price,remain,room\_id)
* room\_type(room\_id,room\_name,hotel\_id)



**查询所有房型的具体信息，包括room\_id, room\_name, hotel\_id。**

SELECT \* FROM room\_type

**查询所有酒店名称中包含“希尔顿”的酒店，返回酒店名称和酒店id。**

SELECT hotel\_name,hotel\_id FROM hotel WHERE hotel\_name LIKE '%希尔顿%'

**查询订单总价在10000元及以上的所有订单详情，包括订单编号、酒店编号、房型编号及居住时长。**

SELECT order\_id,hotel\_id,room\_id, leave\_date-start\_date+1 AS time FROM hotel NATURAL JOIN `order` NATURAL JOIN room\_type WHERE payment>=10000

**查询所有房型的订单情况，包括房型编号，房型名称，订单编号、价格。**

SELECT room\_type.room\_id,room\_type.room\_name,`order`.order\_id,`order`.payment FROM `order` RIGHT JOIN room\_type ON `order`.room\_id=room\_type.room\_id

**创建启悦酒店的订单视图。**

CREATE VIEW qiyue\_hotel AS SELECT `order`.\* FROM `order` NATURAL JOIN hotel NATURAL JOIN room\_type WHERE hotel\_name LIKE '%启悦%'

**在订单表的总价字段上 创建降序的普通索引。索引名为orderpayment。**

CREATE UNIQUE INDEX orderpayment ON `order`(payment DESC)

**查询所有酒店2020-11-14所有房型的平均价格并从低到高排序。**

SELECT hotel\_id,hotel\_name,AVG(price) AS average\_price FROM hotel NATURAL JOIN room\_type NATURAL JOIN room\_info WHERE date LIKE '2020-11-14' GROUP BY hotel\_id ORDER BY average\_price

**从订单表中统计一个酒店在指定日期的各种房型的预订情况。例如统计希尔顿大酒店2020-11-14当天各个房型预定情况，返回酒店名，房型，预定数量。**

SELECT hotel\_name,room\_name,amount FROM hotel NATURAL JOIN room\_type NATURAL JOIN `order` WHERE hotel\_name LIKE '希尔顿大酒店' AND start\_date<='2020-11-14' AND start\_date>='2020-11-14'

**查找同时评价了2次及以上的用户信息。**

SELECT uID,uName FROM customer WHERE uID NOT IN (SELECT uID FROM rating GROUP BY uID HAVING COUNT(uID)=1)

**查询评价过所有总统套房的顾客姓名。**

SELECT DISTINCT uName FROM customer,rating,`order`,room\_type WHERE customer.uID = rating.uID AND rating.order\_id=order.order\_id AND room\_type.room\_id=order.room\_id AND room\_name="总统套房" GROUP BY rating.uID HAVING COUNT(DISTINCT room\_type.room\_id)>=(SELECT COUNT(\*) FROM room\_type WHERE room\_name="总统套房")

**指定时间区间和每天要预定的房间数量，查询满足条件（时间区间，将预定房间数）的房型及其平均价格，并按平均价格从低到高进行排序。查询结果应包含酒店，房型及平均价格信息。例如预定11.14-16日每天房间数量4间**

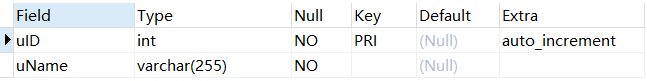
SELECT hotel\_name,room\_name,AVG(price) AS average\_price FROM hotel NATURAL JOIN room\_type NATURAL JOIN room\_info WHERE date<='2020-11-16' AND date>='2020-11-14' GROUP BY room\_id HAVING min(remain)>=4 ORDER BY average\_price

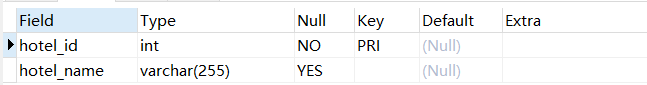
**完成预订房间，包括更新房型信息和创建订单。例如订单为预订11月14号-15号 4号房型 4间。**

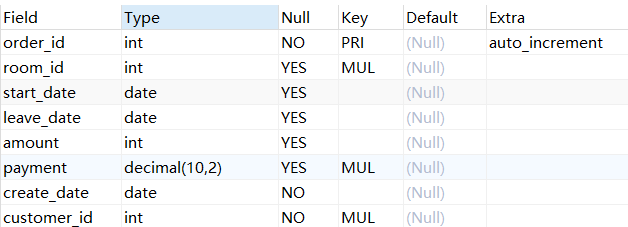
INSERT INTO `order` VALUES (null,4, '2020-11-14', '2020-11-15',4,4\*(SELECT price FROM room\_info WHERE room\_id=4 AND date='2020-11-14'), '2020-11-01',201901);

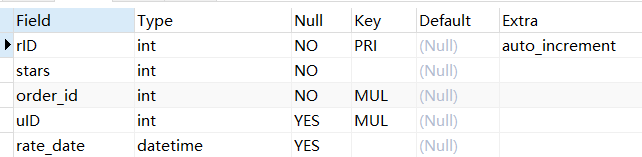
UPDATE room\_info SET remain=remain-4 WHERE room\_id=4 AND date BETWEEN '2020-11-14' AND '2020-11-15';

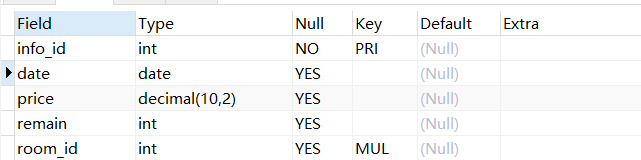
附加题

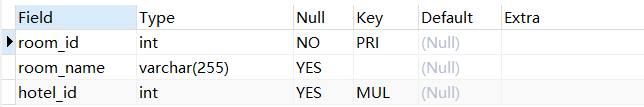












**请用存储过程完成第8题，第11题。**

**请用触发器完成第12题。**

**请设计应用程序（java、python等，编程语言不限）实现11题第12题进行房型查找及房间预订。**

**请编写GUI实现附加题第3题。**