# SQL194 知乎问答最大连续回答问题天数大于等于3天的用户及其对应等级

## 抛真题

描述

现有某乎问答创作者信息表author\_tb如下(其中author\_id表示创作者编号、author\_level表示创作者级别，共1-6六个级别、sex表示创作者性别)：

|  |  |  |
| --- | --- | --- |
| **author\_id** | **author\_level** | **sex** |
| 101 | 6 | m |
| 102 | 1 | f |
| 103 | 1 | m |
| 104 | 3 | m |
| 105 | 4 | f |
| 106 | 2 | f |
| 107 | 2 | m |
| 108 | 5 | f |
| 109 | 6 | f |
| 110 | 5 | m |

创作者回答情况表answer\_tb如下（其中answer\_date表示创作日期、author\_id指创作者编号、issue\_id指回答问题编号、char\_len表示回答字数）：

|  |  |  |  |
| --- | --- | --- | --- |
| **answer\_date** | **author\_id** | **issue\_id** | **char\_len** |
| 2023-11-01 | 101 | E001 | 150 |
| 2023-11-01 | 101 | E002 | 200 |
| 2023-11-01 | 102 | C003 | 50 |
| 2023-11-01 | 103 | P001 | 35 |
| 2023-11-01 | 104 | C003 | 120 |
| 2023-11-01 | 105 | P001 | 125 |
| 2023-11-01 | 102 | P002 | 105 |
| 2023-11-02 | 101 | P001 | 201 |
| 2023-11-02 | 110 | C002 | 200 |
| 2023-11-02 | 110 | C001 | 225 |
| 2023-11-02 | 110 | C002 | 220 |
| 2023-11-03 | 101 | C002 | 180 |
| 2023-11-04 | 109 | E003 | 130 |
| 2023-11-04 | 109 | E001 | 123 |
| 2023-11-05 | 108 | C001 | 160 |
| 2023-11-05 | 108 | C002 | 120 |
| 2023-11-05 | 110 | P001 | 180 |
| 2023-11-05 | 106 | P002 | 45 |
| 2023-11-05 | 107 | E003 | 56 |

请你统计最大连续回答问题的天数大于等于3天的用户及其等级（若有多条符合条件的数据，按author\_id升序排序），以上例子的输出结果如下：

|  |  |  |
| --- | --- | --- |
| **author\_id** | **author\_level** | **days\_cnt** |
| 101 | 6 | 3 |

|  |
| --- |
| drop database if exists db\_1;  create database db\_1;  use db\_1;  drop table if exists author\_tb;  CREATE TABLE author\_tb  (  author\_id int(10) NOT NULL,  author\_level int(10) NOT NULL,  sex char(10) NOT NULL  );  INSERT INTO author\_tb  VALUES  (101, 6, 'm'),  (102, 1, 'f'),  (103, 1, 'm'),  (104, 3, 'm'),  (105, 4, 'f'),  (106, 2, 'f'),  (107, 2, 'm'),  (108, 5, 'f'),  (109, 6, 'f'),  (110, 5, 'm');  drop table if exists answer\_tb;  CREATE TABLE answer\_tb  (  answer\_date date NOT NULL,  author\_id int(10) NOT NULL,  issue\_id char(10) NOT NULL,  char\_len int(10) NOT NULL  );  INSERT INTO answer\_tb  VALUES  ('2021-11-1', 101, 'E001', 150),  ('2021-11-1', 101, 'E002', 200),  ('2021-11-1', 102, 'C003', 50),  ('2021-11-1', 103, 'P001', 35),  ('2021-11-1', 104, 'C003', 120),  ('2021-11-1', 105, 'P001', 125),  ('2021-11-1', 102, 'P002', 105),  ('2021-11-2', 101, 'P001', 201),  ('2021-11-2', 110, 'C002', 200),  ('2021-11-2', 110, 'C001', 225),  ('2021-11-2', 110, 'C002', 220),  ('2021-11-3', 101, 'C002', 180),  ('2021-11-4', 109, 'E003', 130),  ('2021-11-4', 109, 'E001', 123),  ('2021-11-5', 108, 'C001', 160),  ('2021-11-5', 108, 'C002', 120),  ('2021-11-5', 110, 'P001', 180),  ('2021-11-5', 106, 'P002', 45),  ('2021-11-5', 107, 'E003', 56);  select \* from author\_tb;  select \* from answer\_tb; |

## 分析

## 实现

## 小结

# 力扣-2173. 最多连胜的次数

https://leetcode.cn/problems/longest-winning-streak/description/

## 抛真题

表: Matches

+-------------+------+

| Column Name | Type |

+-------------+------+

| player\_id | int |

| match\_day | date |

| result | enum |

+-------------+------+

(player\_id, match\_day) 是该表的主键（具有唯一值的列的组合）。

每一行包括了：参赛选手 id、 比赛时间、 比赛结果。

比赛结果（result）的枚举类型为 ('Win', 'Draw', 'Lose')。

选手的 连胜数 是指连续获胜的次数，且没有被平局或输球中断。

编写解决方案来计算每个参赛选手最多的连胜数。

结果可以以 任何顺序 返回。

结果格式如下例所示：

示例 1:

输入:

Matches 表:

+-----------+------------+--------+

| player\_id | match\_day | result |

+-----------+------------+--------+

| 1 | 2022-01-17 | Win |

| 1 | 2022-01-18 | Win |

| 1 | 2022-01-25 | Win |

| 1 | 2022-01-31 | Draw |

| 1 | 2022-02-08 | Win |

| 2 | 2022-02-06 | Lose |

| 2 | 2022-02-08 | Lose |

| 3 | 2022-03-30 | Win |

+-----------+------------+--------+

输出:

+-----------+----------------+

| player\_id | longest\_streak |

+-----------+----------------+

| 1 | 3 |

| 2 | 0 |

| 3 | 1 |

+-----------+----------------+

解释:

Player 1:

从 2022-01-17 到 2022-01-25, player 1连续赢了三场比赛。

2022-01-31, player 1 平局.

2022-02-08, player 1 赢了一场比赛。

最多连胜了三场比赛。

Player 2:

从 2022-02-06 到 2022-02-08, player 2 输了两场比赛。

最多连赢了0场比赛。

Player 3:

2022-03-30, player 3 赢了一场比赛。

最多连赢了一场比赛。

进阶: 如果我们想计算最长的连续不输的次数（即获胜或平局），你将如何调整？

|  |
| --- |
| drop database if exists db\_1;  create database db\_1;  use db\_1;  Create table If Not Exists Matches (player\_id int, match\_day date, result ENUM('Win', 'Draw', 'Lose'));  Truncate table Matches;  insert into Matches (player\_id, match\_day, result) values ('1', '2022-01-17', 'Win');  insert into Matches (player\_id, match\_day, result) values ('1', '2022-01-18', 'Win');  insert into Matches (player\_id, match\_day, result) values ('1', '2022-01-25', 'Win');  insert into Matches (player\_id, match\_day, result) values ('1', '2022-01-31', 'Draw');  insert into Matches (player\_id, match\_day, result) values ('1', '2022-02-08', 'Win');  insert into Matches (player\_id, match\_day, result) values ('2', '2022-02-06', 'Lose');  insert into Matches (player\_id, match\_day, result) values ('2', '2022-02-08', 'Lose');  insert into Matches (player\_id, match\_day, result) values ('3', '2022-03-30', 'Win');  select \* from matches;  # todo 题目要求: 选手的 连胜数 是指连续获胜的次数，且没有被平局或输球中断。  # todo 编写解决方案来计算每个参赛选手最多的连胜数。  # todo 结果可以以 任何顺序 返回。 |

## 分析

## 实现

## 小结

# 拼多多数据分析面试题：连续3次为球队得分的球员名单

## 抛真题

两支篮球队进行了激烈的比赛，比分交替上升。比赛结束后，你有一个两队分数的明细表（名称为“分数表”）。表中记录了球队、球员号码、球员姓名、得分分数及得分时间。现在球队要对比赛中表现突出的球员进行奖励。

问题：请你写一个SQL语句，统计出连续3次为球队得分的球员名单。

准备工作

|  |
| --- |
| *-- 创建分数表，并为列名增加注释* CREATE TABLE tb\_score (  team\_name VARCHAR(50) COMMENT '球队名称',  player\_id INT COMMENT '球员ID',  player\_name VARCHAR(50) COMMENT '球员姓名',  score INT COMMENT '得分',  score\_time DATETIME COMMENT '得分时间' );  *-- 插入数据以满足条件* INSERT INTO tb\_score (team\_name, player\_id, player\_name, score, score\_time) VALUES ('洛杉矶湖人队', 23, '勒布朗·詹姆斯', 3, '2023-12-25 10:00:00'), ('洛杉矶湖人队', 23, '勒布朗·詹姆斯', 3, '2023-12-25 10:15:00'), ('洛杉矶湖人队', 23, '勒布朗·詹姆斯', 1, '2023-12-25 10:30:00'), ('洛杉矶湖人队', 3, '安东尼·戴维斯', 2, '2023-12-25 10:32:00'), ('洛杉矶湖人队', 23, '勒布朗·詹姆斯', 3, '2023-12-25 10:45:00'), ('洛杉矶湖人队', 23, '勒布朗·詹姆斯', 3, '2023-12-25 11:00:00'), ('洛杉矶湖人队', 23, '勒布朗·詹姆斯', 2, '2023-12-25 11:15:00'), ('洛杉矶湖人队', 23, '勒布朗·詹姆斯', 2, '2023-12-25 11:30:00'), ('金州勇士队', 30, '斯蒂芬·库里', 1, '2023-12-25 10:10:00'), ('金州勇士队', 30, '斯蒂芬·库里', 1, '2023-12-25 10:25:00'), ('金州勇士队', 30, '斯蒂芬·库里', 1, '2023-12-25 10:40:00'), ('金州勇士队', 11, '克莱·汤普森', 2, '2023-12-25 10:45:00'), ('金州勇士队', 30, '斯蒂芬·库里', 2, '2023-12-25 10:55:00'), ('金州勇士队', 30, '斯蒂芬·库里', 2, '2023-12-25 11:10:00'), ('金州勇士队', 30, '斯蒂芬·库里', 3, '2023-12-25 11:25:00'), ('金州勇士队', 30, '斯蒂芬·库里', 3, '2023-12-25 11:40:00'), ('金州勇士队', 30, '斯蒂芬·库里', 3, '2023-12-25 11:55:00');  select *\** from tb\_score; |

## 分析

## 实现

## 小结

# 美团面试题-查询2023年每个⽉，连续2天都有登陆的⽤⼾名单。

## 抛真题

有⼀张 ''⽤⼾登陆记录表"，包含两个字段：⽤⼾id、⽇期。

|  |  |
| --- | --- |
| user\_id | login\_date |
| 1 | 2023-01-03 |
| 1 | 2023-01-04 |
| 1 | 2023-01-05 |
| 1 | 2023-01-08 |
| 1 | 2023-01-09 |
| 1 | 2023-02-16 |
| 1 | 2023-02-17 |
| 1 | 2023-02-27 |
| 2 | 2023-01-10 |
| 2 | 2023-01-11 |
| 2 | 2023-03-08 |
| 2 | 2023-03-09 |
| 3 | 2023-02-08 |
| 3 | 2023-02-09 |
| 3 | 2023-02-10 |
| 3 | 2023-02-15 |
| 3 | 2023-03-09 |
| 3 | 2023-03-19 |

【问题】查询2023年每个⽉，连续2天都有登陆的⽤⼾名单。

|  |  |
| --- | --- |
| **y\_month** | **user\_id** |
| **2023-01** | **1** |
| **2023-01** | **2** |
| **2023-02** | **1** |
| **2023-02** | **3** |
| **2023-03** | **2** |

|  |
| --- |
| -- todo 1 准备工作  drop database if exists db\_1;  create database db\_1;  use db\_1;  create table tb\_user\_login(  user\_id varchar(32),  login\_date varchar(32)  )  ;  insert into db\_1.tb\_user\_login  values  ('1', '2023-01-03'),  ('1', '2023-01-04'),  ('1', '2023-01-05'),  ('1', '2023-01-08'),  ('1', '2023-01-09'),  ('1', '2023-02-16'),  ('1', '2023-02-17'),  ('1', '2023-02-27'),  ('2', '2023-01-10'),  ('2', '2023-01-11'),  ('2', '2023-03-08'),  ('2', '2023-03-09'),  ('3', '2023-02-08'),  ('3', '2023-02-09'),  ('3', '2023-02-10'),  ('3', '2023-02-15'),  ('3', '2023-03-09'),  ('3', '2023-03-19')  ;  select \* from db\_1.tb\_user\_login;  # todo 【问题】查询2023年每个⽉，连续2天都有登陆的⽤⼾名单。 |

## 分析

## 实现

## 小结

# 小鹏面试题: 小鹏汽车充电每辆车连续快充最大次数

## 抛真题

需求: 小鹏汽车充电每辆车连续快充最大次数

输入

|  |  |  |
| --- | --- | --- |
| id | charge\_time | charge\_type |
| XP1001 | 2023/11/20 08:45:00 | 快充 |
| XP1001 | 2023/11/21 20:45:00 | 快充 |
| XP1001 | 2023/11/22 08:45:00 | 快充 |
| XP1001 | 2023/11/23 08:45:00 | 慢充 |
| XP1001 | 2023/11/25 08:45:00 | 快充 |
| XP1002 | 2023/11/25 08:45:00 | 快充 |
| XP1002 | 2023/11/25 12:45:00 | 快充 |
| XP1002 | 2023/11/25 23:45:00 | 慢充 |
| XP1003 | 2023/11/25 23:45:00 | 慢充 |
| XP1003 | 2023/11/26 23:45:00 | 慢充 |

输出

|  |  |
| --- | --- |
| id | cnts |
| XP1001 | 3 |
| XP1002 | 2 |
| XP1003 | 0 |

|  |
| --- |
| -- 小鹏充电  drop database if exists db\_1;  create database if not exists db\_1;  use db\_1;  CREATE TABLE charging\_data (  id VARCHAR(50),  charge\_time DATETIME,  charge\_type VARCHAR(10)  );  INSERT INTO charging\_data (id, charge\_time, charge\_type)  VALUES  ('XP1001', '2023-11-20 08:45:00', '快充'),  ('XP1001', '2023-11-21 20:45:00', '快充'),  ('XP1001', '2023-11-22 08:45:00', '快充'),  ('XP1001', '2023-11-23 08:45:00', '慢充'),  ('XP1001', '2023-11-25 08:45:00', '快充'),  ('XP1002', '2023-11-25 08:45:00', '快充'),  ('XP1002', '2023-11-25 12:45:00', '快充'),  ('XP1002', '2023-11-25 23:45:00', '慢充'),  ('XP1003', '2023-11-25 23:45:00', '慢充'),  ('XP1003', '2023-11-26 23:45:00', '慢充')  ;  # todo 需求: 小鹏汽车充电每辆车连续快充最大次数 |

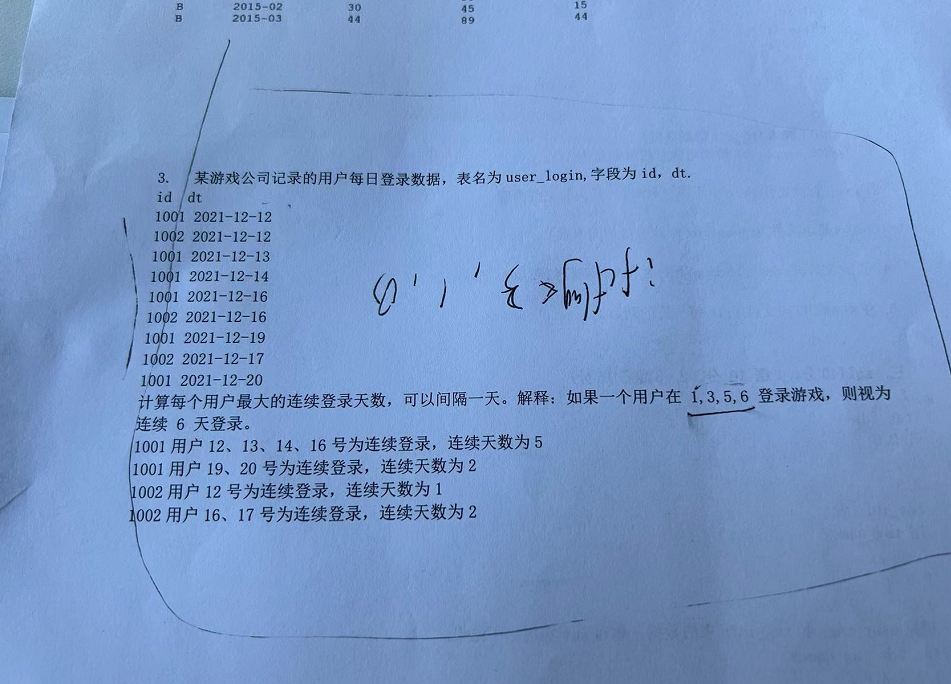
## 分析

## 实现

## 小结

# 指南针面试题-间断连续真题

## 抛真题



3. 某游戏公司记录的用户每日登录数据,表名为user\_login,字段为id,dt.

id dt

1001 2021-12-12

1002 2021-12-12

1001 2021-12-13

1001 2021-12-14

1001 2021-12-16

1002 2021-12-16

1001 2021-12-19

1002 2021-12-17

1001 2021-12-20

计算每个用户最大的连续登录天数,可以间隔一天。

解释:如果一个用户在1,3,5,6登录游戏,则视为连续6天登录。

1001用户12、13、14、16号为连续登录,连续天数为5

1001用户19、20号为连续登录,连续天数为2

1002用户12号为连续登录,连续天数为1

1002用户16、17号为连续登录,连续天数为2

|  |
| --- |
| drop database if exists db\_1;  create database db\_1;  use db\_1;  create table if not exists user\_login(  id int,  dt varchar(32)  );  insert into user\_login  values  (1001, '2021-12-12'),  (1002, '2021-12-12'),  (1001, '2021-12-13'),  (1001, '2021-12-14'),  (1001, '2021-12-16'),  (1002, '2021-12-16'),  (1001, '2021-12-19'),  (1002, '2021-12-17'),  (1001, '2021-12-20')  ;  select \* from user\_login; |

## 分析

## 实现

## 小结