《数据库概论》实验一：用SQL进行数据操作实验报告

姓名: 傅小龙 学号: 191220029 联系方式: 1830970417(QQ)

一、实验环境

操作系统: Windows 10 (19042.1288)

Mysql Workbench 8.0 CE

mysql-8.0.19-winx64

二、实验过程

任务1:

问题描述：有多少物种 `species` 的描述 `description` 中含有单词"this"？查询以如下形式返回：`(speciesCount)`。

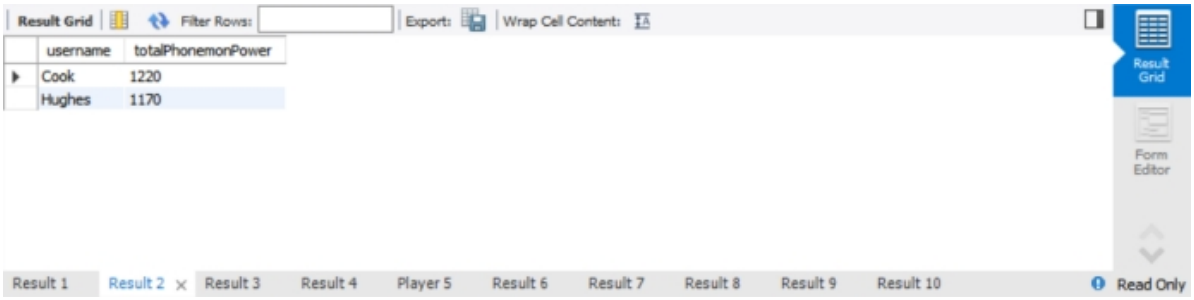
```
select count(*) speciesCount
from Species
where description like '%this%';
```

任务2:

问题描述：玩家 `player'Cook'` 将与玩家 `player'Hughes'` 作战。对于两个玩家，显示他们的用户名 `username` 和他们各自拥有的 `Phonemon` 的总能量。查询以如下形式返回：`(username, totalPhonemonPower)`。

思路：在 `Player, Phonemon` 的联结表中找到 `username` 为 `Cook` 与 `Hughes` 的元组，按 `username` 分组求 `power` 的和。

```
select Player.username, SUM(Phonemon.power)totalPhonemonPower
from Player, Phonemon
where (username = 'Cook' OR username = 'Hughes') and Player.id = Phonemon.player
group by username;
```



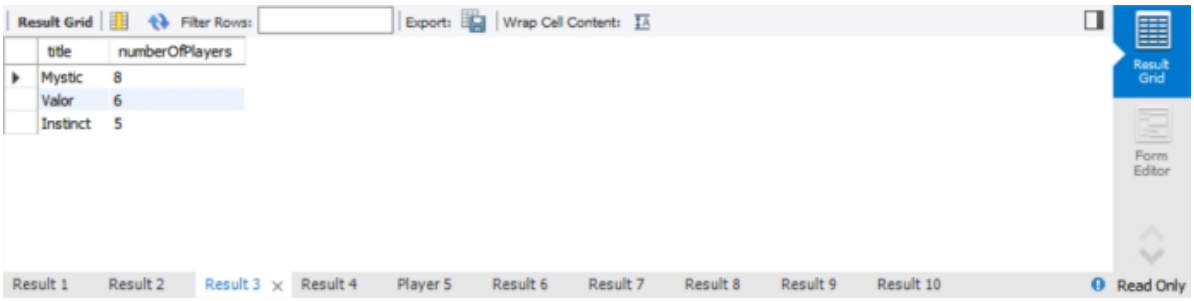
username	totalPhonemonPower
Cook	1220
Hughes	1170

任务3：

问题描述：每一个队伍team有多少名成员 `player`？按照玩家数量降序列出队伍名称title和玩家数量。查询以如下形式返回：`(title, numberOfPlayers)`。

思路：在 `Player` 和 `Team` 的联结表中，根据队伍 `id` 分组对相同队伍 `id` 的玩家计数。

```
select title, count(Player.id) numberOfPlayers
from Player, Team
where Player.team = Team.id
group by Team.id
order by count(Player.id) desc;
```



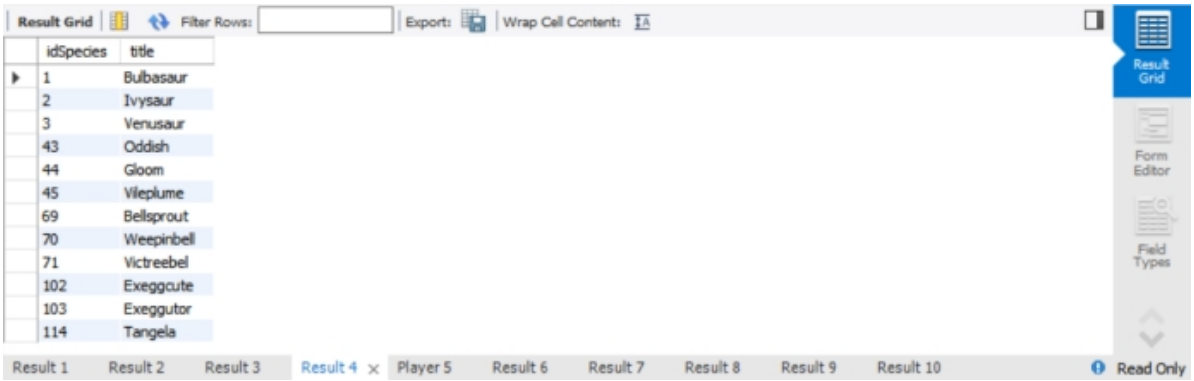
title	numberOfPlayers
Mystic	8
Valor	6
Instinct	5

任务4：

问题描述：哪些物种species具有类型 `type'grass'`？查询以如下形式返回：`(idSpecies, title)`。

思路：在 `Species` 和 `Type` 的联结表中，选出类型为 `grass` 的元组。由于 `species` 具有2个 `Type.id` 外键: `type1` 和 `type2`，故联结条件为 `type1 = Type.id OR type2 = Type.id`.

```
select Species.id idSpecies, Species.title
from Species, Type
where Type.title = 'grass' AND (type1 = Type.id OR type2 = Type.id);
```



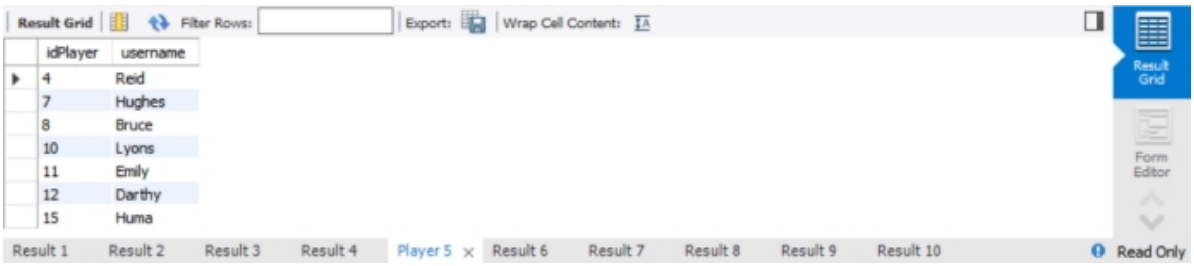
idSpecies	title
1	Bulbasaur
2	Ivysaur
3	Venusaur
43	Oddish
44	Gloom
45	Vileplume
69	Bellsprout
70	Weepinbell
71	Victreebel
102	Exeggcute
103	Exeggutor
114	Tangela

任务5：

问题描述：列出从未购买过食物 `food` 的玩家 `player`。查询以如下形式返回：`(idPlayer, username)`。

思路：在 `Player, Item, Purchase` 的联结表中选出购买过食物的玩家 `id`,然后在 `Player` 表中通过谓词 `not in` 选出未购买过食物的玩家。

```
select id idPlayer, username
from Player
where Player.id not in
(
    select Player.id
    from Player, Item, Purchase
    where Player.id = Purchase.player and Purchase.item = Item.id and Item.type
= 'F'
);
```



任务6：

问题描述：游戏中的每个玩家 `player` 具有特定的等级 `level`。以金额降序列出每一特定等级以及该等级的所有玩家在购买上花费的总金额。查询以如下形式返回：`(level, totalAmountSpentByAllPlayersAtLevel)`。

思路：在 `Player, Purchase, Item` 的联结表中按玩家等级分类，统计各等级在购买中花费金额的和。

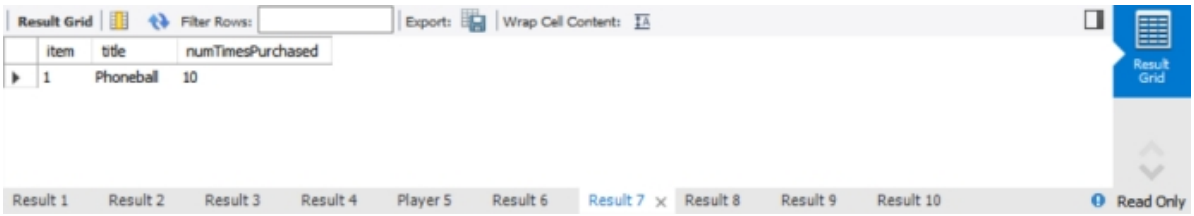
```
select Player.level, sum(Purchase.quantity * Item.price)
totalAmountSpentByAllPlayersAtLevel
from Player, Purchase, Item
where Player.id = Purchase.player and Item.id = Purchase.item
group by level
order by sum(Purchase.quantity * Item.price) desc;
```

任务7：

问题描述：什么物品 `item` 被购买次数最多？如有并列，列出所有购买次数最多的物品。查询以如下形式返回：`(item, title, numTimesPurchased)`。

思路：在 `purchase, item` 的联结表中，按 `purchase.item` 分组统计被购买的次数，用 `having` 选出购买次数大于等于 `purchase` 表中所有 `item` 购买次数的分组。

```
select purchase.item, item.title, count(purchase.item) numTimesPurchased
from purchase, item
where purchase.item = item.id
group by purchase.item
having numTimesPurchased >= all(
    select count(purchase.item)
    from purchase
    group by purchase.item
);
```

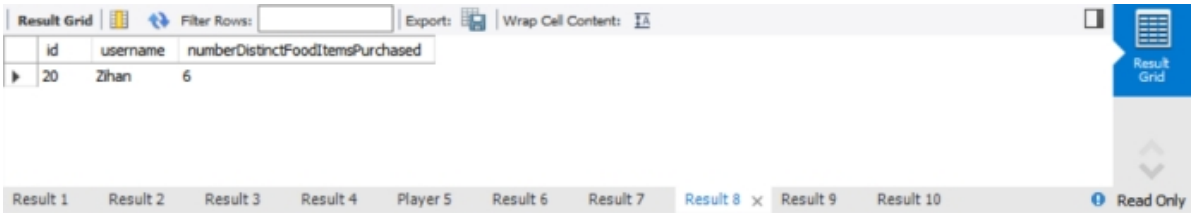


任务8：

问题描述：找到可获取的食物的数量，和购买所有种类食物至少各一次的玩家。查询以如下形式返回：`(playerID, username, numberDistinctFoodItemsPurchased)`。

思路：在`Player, Purchase, Item`的联结表中，按玩家`id`分类，统计购买食物的种类数量，用`having`选出购买食物种类数量和`food`表中记录数量（即食物种类数）相同的组。

```
select Player.id, Player.username, count(distinct Item.id)
numberDistinctFoodItemsPurchased
from Player, Purchase, Item
where Player.id = Purchase.player and Purchase.item = Item.id and Item.type =
'F'
group by Player.id
having numberDistinctFoodItemsPurchased = (select count(*) from food);
```



任务9：

问题描述：将距离最近的两个`Phonemon`之间的欧氏距离称为`X`。计算相互之间距离为`X`的`Phonemon`对的数量。查询以如下形式返回：`(numberOfPhonemonPairs,distanceX)`。

思路：在`Phonemon`的自联结表`t1, t2`中计算点对距离，精度取小数点后3位。为不重复计算点对距离，联结条件为`t1.id < t2.id`. 按点对距离`distanceX`分组，统计具有相同距离的点对数量，用`having`选出`distanceX`小于等于所有点对距离的组。

```
select count(*) numberOfPhonemonPairs, round(sqrt(power((t1.latitude -
t2.latitude), 2) + power((t1.longitude - t2.longitude), 2)) * 100, 3) distanceX
from Phonemon t1, Phonemon t2
where t1.id < t2.id
group by distanceX
having distanceX <= all(
    select round(sqrt(power((t1.latitude - t2.latitude), 2) +
power((t1.longitude - t2.longitude), 2)) * 100, 3) distanceX
    from Phonemon t1, Phonemon t2
    where t1.id < t2.id
);
```

任务10：

问题描述：一些玩家`player`热衷于某种特定类型`type`的`Phonemon`。列出捕捉了任一特定类型`type`中每一物种`species`至少各一个`Phonemon`的玩家的名称以及该类型的名称。

思路：在`player, phonemon, species, type`的联结表中按`player.id, type.id`分组，统计同一玩家抓住的同一`type`的`phonemon`的`species`物种数量，用`having`选出这一数量等于`Species`表中同一`type`的物种数量的组。

```
select t.username, t.typeTitle
from
(
    select player.username username, type.title typeTitle, type.id
    from player, phonemon, species, type
    where player.id = phonemon.player and phonemon.species = species.id and
(species.type1 = type.id or species.type2 = type.id)
    group by player.id, type.id
    having count(distinct species.id) = (
        select count(*) from Species where Species.type1 = type.id or
Species.type2 = type.id
    )
)as t
```

四、实验中遇到的困难及解决办法

本次实验中较为困难的部分在于编写出形式简单且易于理解的mysql语句。

例如任务8：找到可获取的食物的数量，和购买所有种类食物至少各一次的玩家。在实验时最初的想法是先找出购买所有种类食物各一次的玩家id, 然后再根据找出的id在Player, Purchase, Item, Food的联结表中进行统计，给出查询结果。其实现如下：

```
select playerId, username,  num numberDistinctFoodItemsPurchased
from
    (select Player.id playerID, Player.username username,
count(Purchase.quantity) num
    from Player, Purchase, Item, Food
    where Player.id = Purchase.player and Purchase.item =  Item.id and Item.id =
Food.id
    group by Player.id, Purchase.item) as t
where playerID in (
    select t.id from
    (select Player.id id, Purchase.item item
     from Player, Purchase
     where Player.id = Purchase.player
     group by Player.id, Purchase.item) as t
    group by t.id
    having count(item) = (select count(*) from Food)
)
group by playerId;
```

实际上，可以直接通过在Player, Purchase, Item的联结表中用count(distinct Item.id)得到玩家购买的不同商品的数量，然后通过numberDistinctFoodItemsPurchased = (select count(*) from food) 判断该数量是否和所有食品的种类数相同。简化后的查询语句如下：

```
select Player.id, Player.username, count(distinct Item.id)
numberDistinctFoodItemsPurchased
from Player, Purchase, Item
where Player.id = Purchase.player and Purchase.item = Item.id and Item.type =
'F'
group by Player.id
having numberDistinctFoodItemsPurchased = (select count(*) from food);
```

让语句变得简单易懂的关键在于尽可能多地利用单次查询可以得到的信息，合理高效地使用统计方法。