# 多表联查，将players表和match\_details表、Heros表连载一起，用Group BY语句进行排序，在最后Having来限定分组需要满足的条件

# 第一个查询

SELECT

md.Player\_ID,

p.Player\_Name,

md.Hero\_ID,

h.Hero\_Name,

AVG(md.Kills) AS Avg\_Kills,

AVG(md.Deaths) AS Avg\_Deaths,

AVG(md.Assists) AS Avg\_Assists

FROM

Match\_Details md

JOIN

Players p ON md.Player\_ID = p.Player\_ID

JOIN

Heroes h ON md.Hero\_ID = h.Hero\_ID

GROUP BY

md.Player\_ID, md.Hero\_ID

HAVING

COUNT(md.Match\_ID) >= 10

ORDER BY

Avg\_Kills DESC,

Avg\_Assists DESC,

Avg\_Deaths ASC;

# 同样进行多表联查,本项查到是对应英雄喜欢的武器，因此要英雄表（Heros），同时也要Player\_Item表，因为设计到武器名称，故此也有Item表

# 第二个查询

SELECT

h.Hero\_Name,

t.Role,

i.Item\_Name,

COUNT(pi.Item\_ID) AS Item\_Built\_Count

FROM

Player\_Items pi

JOIN

Items i ON pi.Item\_ID = i.Item\_ID

JOIN

Players p ON pi.Player\_ID = p.Player\_ID

JOIN

Match\_Details md ON pi.Match\_ID = md.Match\_ID AND pi.Player\_ID = md.Player\_ID

JOIN

Heroes h ON md.Hero\_ID = h.Hero\_ID

JOIN

Teams t ON p.Player\_ID = t.Player\_ID

GROUP BY

h.Hero\_Name, t.Role, i.Item\_Name

ORDER BY

Item\_Built\_Count DESC;

第三个查询

# 查询英雄对组合，重点因为英雄可能属于不同的队伍，因此我们需要连接Team表来进行查询

# 同时count来计算组合出现的次数

# 用order by实现排序

SELECT

LEAST(h1.Hero\_Name, h2.Hero\_Name) AS Hero1\_Name,

LEAST(h1.Hero\_ID, h2.Hero\_ID) AS Hero1\_ID,

GREATEST(h1.Hero\_Name, h2.Hero\_Name) AS Hero2\_Name,

GREATEST(h1.Hero\_ID, h2.Hero\_ID) AS Hero2\_ID,

COUNT(\*) AS Combination\_Count

FROM

Match\_Details md1

JOIN

Match\_Details md2 ON md1.Match\_ID = md2.Match\_ID AND md1.Player\_ID < md2.Player\_ID

JOIN

Teams t1 ON md1.Player\_ID = t1.Player\_ID

JOIN

Teams t2 ON md2.Player\_ID = t2.Player\_ID AND t1.Team\_ID = t2.Team\_ID

JOIN

Heroes h1 ON md1.Hero\_ID = h1.Hero\_ID

JOIN

Heroes h2 ON md2.Hero\_ID = h2.Hero\_ID

GROUP BY

LEAST(h1.Hero\_Name, h2.Hero\_Name), LEAST(h1.Hero\_ID, h2.Hero\_ID),

GREATEST(h1.Hero\_Name, h2.Hero\_Name), GREATEST(h1.Hero\_ID, h2.Hero\_ID)

ORDER BY

Combination\_Count DESC

LIMIT 10;