Visualisatie League of Legends Data

Maurice Ponte, Tjeerd van Gelder, Chris-Jan Borgers, Albert Berends

Data & Required packages.

In deze blok defineren we de benodigde connectie met de database. Deze is opgezet in Postgresql en word aangevuld met data door de, in java geschreven, parser. Hier kunnen ook de parametes worden aangepast om de database te bereiken. (Port etc)

```
# install.packages("RPostgreSQL")
require("RPostgreSQL")
```

```
## Loading required package: RPostgreSQL
```

```
## Warning: package 'RPostgreSQL' was built under R version 3.6.3
```

```
## Loading required package: DBI
```

```
## Warning: package 'DBI' was built under R version 3.6.3
```

```
library(tibble)
# create a connection

# loads the PostgreSQL driver
drv <- dbDriver("PostgreSQL")
# creates a connection to the postgres database
# note that "con" will be used later in each connection to the database
connectDB <- function(user,password){
  host <- "localhost"
  dbName <- "lolparserdata"    #name of database
  port <- 5432                 #port of database server

  driver <- dbDriver("PostgreSQL")
  conn <- dbConnect(driver, dbname = dbName, host = host, port = port, user = user,password =  password)
}

username <- "postgres"
password <- "!RappaR1964"
```

Inleiding

In dit bestand gaan we de vragen beantwoord die in de plan van aanpak worden beschreven. Welke vragen willen we beantwoorden met deze database?

1. Which champion is banned the most in tournament/ brackets(ranks)?
2. Which champion has the highest win rate in tournament/ brackets(ranks)?
3. Does getting “First Blood” increase the odds of winning a match?
4. Is a high champion mastery (onetrick) relate to a high rank?
5. How long does an average game last per bracket?
6. What is the average vision score per region?
7. How long does an average game last per bracket?
8. What is the most used summoner spell combo?
9. What is the least used summoner spell combo?
10. What are the top 5 most bought items.
11. What are the top 5 least bought items.

#Vragen & antwoorden

#Vraag 1: Which champion Which champion is banned the most in tournament/ brackets(ranks)?

```
# Get the data
con <- connectDB(username,password)
qResult <- dbGetQuery(con, "SELECT champion.name as Name,
                           count(bans.bannedchampion) as Count,
                           count(bans.bannedchampion) /
                           (SELECT count(bannedchampion)/1000 FROM bans) as banrate

                           FROM champion, bans where champion.championID = bans.bannedchampion
                           GROUP BY champion.name
                           ORDER BY Count DESC
                           Fetch first 10 rows only")
as_tibble(qResult)
```

```
## # A tibble: 10 x 3
##    name      count banrate
##    <chr>     <dbl>   <dbl>
##  1 Yasuo      4374      48
##  2 Darius     3927      43
##  3 Master Yi  3329      36
##  4 Aphelios   3198      35
##  5 Zed        3119      34
##  6 Morgana    3098      34
##  7 Sett       3043      33
##  8 Kassadin   2782      30
##  9 Nautilus   2744      30
## 10 Diana      2317      25
```

```
dbDisconnect(con)
```

```
## [1] TRUE
```

#Conclusie vraag 1

Zie bovenstaande tabel

#Vraag 2: Which champion has the highest win rate? (Of the 10 most chosen games)

```
# Get the data
con <- connectDB(username,password)
qResult <- dbGetQuery(con, "SELECT C.name, COUNT(C.CHAMPIONID) AS WinCount,(SELECT COUNT(CHAMPIONID) FR
dbDisconnect(con)
```

```
## [1] TRUE
```

```
percentageYasuo <- qResult[1,2] / qResult[1,3] * 100
percentageTresh <- qResult[2,2] / qResult[2,3] * 100
percentageEzreal <- qResult[3,2] / qResult[3,3] * 100
percentageMissFortune <- qResult[4,2] / qResult[4,3] * 100
percentageLeeSin <- qResult[5,2] / qResult[5,3] * 100
percentageEkko <- qResult[6,2] / qResult[6,3] * 100
percentageJax <- qResult[7,2] / qResult[7,3] * 100
percentageKatarina <- qResult[8,2] / qResult[8,3] * 100
percentageKaiSa <- qResult[9,2] / qResult[9,3] * 100
percentageVladimir <- qResult[10,2] / qResult[10,3] * 100

winPercentage <- c(percentageYasuo,percentageTresh,percentageEzreal,percentageMissFortune,
                   percentageLeeSin,percentageEkko,percentageJax,percentageKatarina,
                   percentageKaiSa,percentageVladimir)

qResult['winPercentage'] <- winPercentage

as_tibble(qResult)
```

```
## # A tibble: 10 x 4
##    name         wincount gamesplayed winPercentage
##    <chr>           <dbl>       <dbl>         <dbl>
##  1 Yasuo              93         212          43.9
##  2 Ezreal             75         151          49.7
##  3 Jax                73         157          46.5
##  4 Master Yi          69         133          51.9
##  5 Thresh             67         163          41.1
##  6 Lucian             65         121          53.7
##  7 Ekko               65         149          43.6
##  8 Miss Fortune       64         161          39.8
##  9 Katarina           61         131          46.6
## 10 Lee Sin            59         143          41.3
```

Conclusie vraag 2

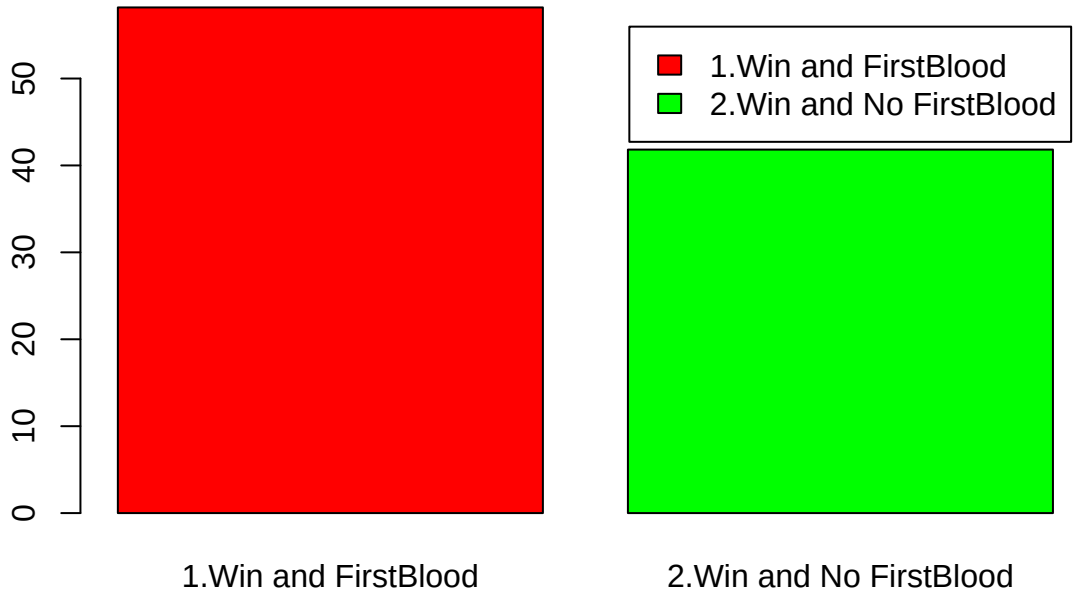
We zien dat er minimale verschillen zijn. Echter zijn we tot de conclusie gekomen dat deze vraag niet geheel het hele beeld laat zien. De gamesplayed zijn bijvoorbeeld minimaal en speler afhankelijk. Om deze vraag wel goed te kunnen beantwoorden zouden we meer data moeten binnenhalen.

#Vraag 3: Does getting "First Blood" increase the odds of winning a match?

```
# Get the data
con <- connectDB(username,password)
qResult <- dbGetQuery(con,"SELECT COUNT(T.matchteamid) as aantal, T.WIN, T.firstbloodteam FROM teamdata
dbDisconnect(con)
```

```
## [1] TRUE
```

```
qResultTotal <-  qResult[2,1] + qResult[3,1]

#Percentages berekenen
pWT <- (qResult[3,1] / qResultTotal) * 100 # Win and FirstBlood
pWF <- (qResult[2,1] / qResultTotal) * 100 # Win and no FirstBlood


#Plot data
slices <- c(pWT,pWF)
lbls <- c("1.Win and FirstBlood","2.Win and No FirstBlood")

barplot(slices,names.arg = lbls, legend = lbls, col = c("red","green"))
```




```
printP <- function(){
cat("Percentage van wins met FirstBlood: ")
cat(pWT)
cat(" % \n")
cat("Percentage van wins met zonder FirstBlood: ")
cat(pWF)
cat(" % \n")
cat("Verschil: ")
cat(pWT-pWF)
cat(" % \n")
}
printP()
```

```
## Percentage van wins met FirstBlood: 58.17856 %
## Percentage van wins met zonder FirstBlood: 41.82144 %
## Verschil: 16.35712 %
```

#Conclusie vraag 3

We zien in de geplotte data, een minimaal verschil in de percentages van Wins met en zonder FirstBlood het verschil is dan ook 16%. Dit betekend niet dat er letterlijk 16% meer kans is op winst bij het behalen van firstBlood. Het zegt enkel dat bij de gewonnen matches er 16% vaker is gewonnen met FirstBlood dan zonder.

#Vraag 4 : Is a high champion mastery (onetrick) relate to a high rank?

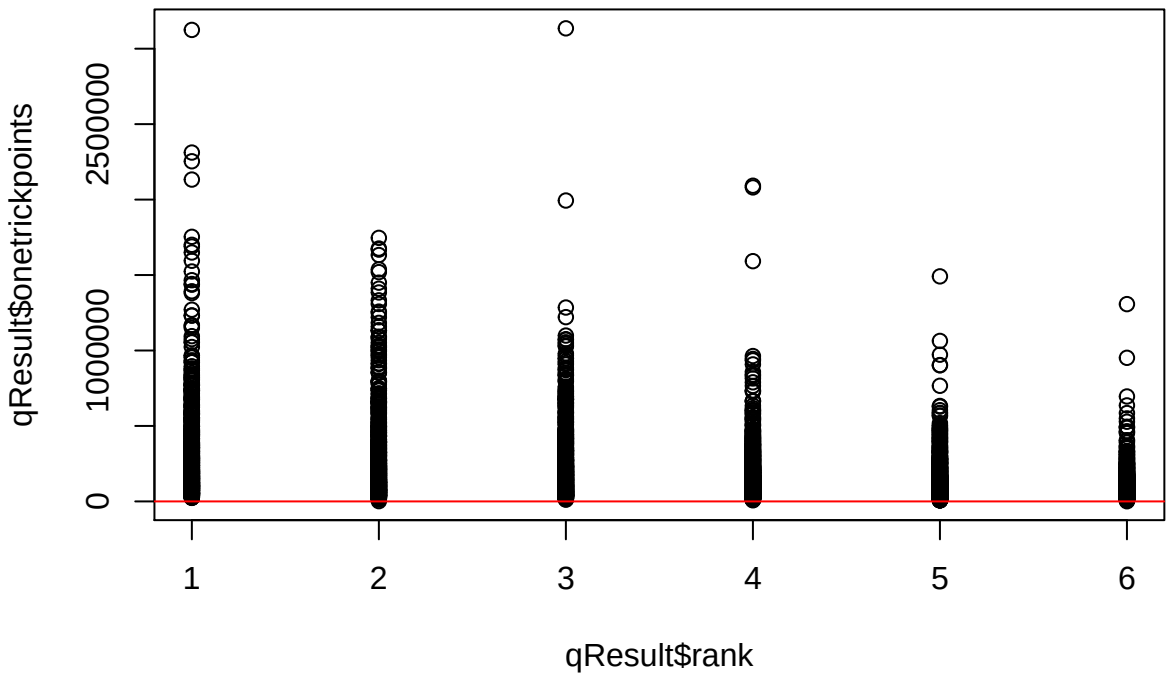
```
# Get the data
con <- connectDB(username,password)
qResult <- dbGetQuery(con, 'SELECT CASE S.tier
                           WHEN cast(\'"DIAMOND"\' as varchar) THEN 1
                           WHEN cast(\'"PLATINUM"\' as varchar) THEN 2
                           WHEN cast(\'"GOLD"\' as varchar) THEN 3
                           WHEN cast(\'"SILVER"\' as varchar) THEN 4
                           WHEN cast(\'"BRONZE"\' as varchar) THEN 5
                           WHEN cast(\'"IRON"\' as varchar) THEN 6
                           ELSE 0 END AS RANK,
                                 (SELECT CHAMPIONPOINTS
                                 FROM CHAMPIONMASTERY
                                 WHERE S.SUMMONERID = CHAMPIONMASTERY.ACCOUNTID
                                 FETCH FIRST ROW ONLY) AS ONETRICKPOINTS
                           FROM SUMMONER S
                           ')

dbDisconnect(con)
```

```
## [1] TRUE
```

```
plot(qResult$rank,qResult$onetrickpoints)
lrm <- lm(data=qResult)
summary(lrm)
```

```
##
## Call:
## lm(data = qResult)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.8709 -1.5013  0.1872  1.2995  5.7426
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)     3.920e+00  3.096e-02  126.61   <2e-16 ***
## onetrickpoints -2.125e-06  1.028e-07  -20.69   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.638 on 4918 degrees of freedom
## Multiple R-squared:  0.08004,    Adjusted R-squared:  0.07985
## F-statistic: 427.9 on 1 and 4918 DF,  p-value: < 2.2e-16
```

```
abline(lrm,col='red')
```



#Conclusie vraag 4

In de bovenstaande tabel is rank 1 het hoogste en rank 6 de laagste. De fit van dit model, die word aangegevens door de rode lijn, is recht. Dit betekend dat er niet een linear verband is tussen de rank en de hoeveelheid onetrickpoints. We zien wel in de grafiek dat de spelers met het hoogste aantal onetrick points, per rank ook omhoog gaan. Wat is te verklaren omdat, wanneer een speler langer speelt, automatisch in een hogere rank komt en dus ook meer tijd heeft gehad om deze points te verzamelen. In de laagste rank zitten spelers die wellicht enkel speeltijd met 1 champion hebben gehad en daardoor veel onetrickpoints hebben.

#Vraag 5: How long does an average game last per bracket?

```
# Get the data
con <- connectDB(username,password)
qResult <- dbGetQuery(con, "SELECT S.Tier AS TIER,
                                AVG(T.matchduration) AS AverageDurationMins
                                FROM Summoner S, TEAMDATA T, MATCHHISTORY M
                                WHERE M.accountid = S.accountid AND M.matchid = T.matchID
                                GROUP BY S.Tier
                                ")

#Sort data by tier highest to lowest
tiers <- c(qResult[4,1],qResult[3,1],qResult[1,1],qResult[2,1],qResult[5,1])
tiersData <- c(qResult[4,2],qResult[3,2],qResult[1,2],qResult[2,2],qResult[5,2])
averageGame <- data.frame(tiers,tiersData)

dbDisconnect(con)
```

```
## [1] TRUE
```

```
as_tibble(averageGame)
```

```
## # A tibble: 5 x 2
##   tiers          tiersData
##   <fct>              <dbl>
## 1 "\"DIAMOND\""       26.8
## 2 "\"PLATINUM\""      27.7
## 3 "\"GOLD\""          28.5
## 4 "\"SILVER\""        29.2
## 5 "\"BRONZE\""        29.4
```

#Conclusie vraag 5 Zie bovenstaande tabel. We zien hier dat hoe hoger het niveau dat gemiddeld de duratie van een potje daald.

#Vraag 6: What is the average vision score per region?

```
# Get the data
con <- connectDB(username,password)
qResult <- dbGetQuery(con, 'SELECT CASE S.tier
                           WHEN cast(\'"DIAMOND"\' as varchar) THEN 1
                           WHEN cast(\'"PLATINUM"\' as varchar) THEN 2
                           WHEN cast(\'"GOLD"\' as varchar) THEN 3
                           WHEN cast(\'"SILVER"\' as varchar) THEN 4
                           WHEN cast(\'"BRONZE"\' as varchar) THEN 5
                           WHEN cast(\'"IRON"\' as varchar) THEN 6
                           ELSE 0 END AS RANK,
                                 (SELECT SUM(playerkills)+SUM(playerassists)
                                 FROM matchhistory
                                 WHERE S.accountid = matchhistory.ACCOUNTID
                                 FETCH FIRST ROW ONLY) AS KILLS
                           FROM SUMMONER S
                           ')
dbDisconnect(con)
```

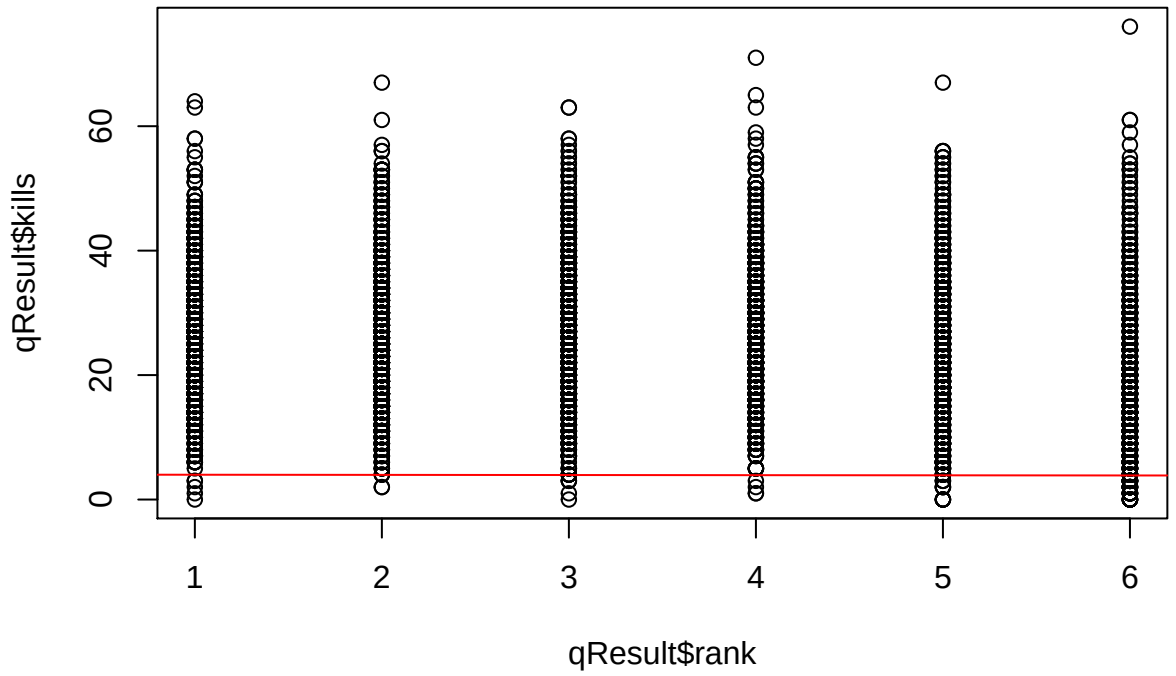
```
## [1] TRUE
```

```
plot(qResult$rank,qResult$kills)
lrm <- lm(data=qResult)
summary(lrm)
```

```
##
## Call:
## lm(data = qResult)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.0055 -1.4646 -0.0364  1.4453  3.7074
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  4.005458   0.058083   68.96   <2e-16 ***
## kills       -0.022537   0.002089  -10.79   <2e-16 ***
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.673 on 4763 degrees of freedom
##   (155 observations deleted due to missingness)
## Multiple R-squared:  0.02386,    Adjusted R-squared:  0.02365
## F-statistic: 116.4 on 1 and 4763 DF,  p-value: < 2.2e-16
```

```
abline(lrm,col='red')
```



#Conclusie vraag 6

Ook hier zien we geen direct verband.

#Vraag 7: How long does an average game last per bracket?

```
# Get the data
con <- connectDB(username,password)
qResult <- dbGetQuery(con, 'SELECT S.TIER AS TIER, AVG(T.matchduration) AS AVGMatchDurationMins FROM MA
as_tibble(qResult)
```

```
## # A tibble: 5 x 2
##   tier           avgmatchdurationmins
##   <chr>                         <dbl>
## 1 "\"BRONZE\""                   29.4
## 2 "\"DIAMOND\""                  26.8
```

```
## 3 "\"GOLD\""                    28.5
## 4 "\"PLATINUM\""                27.7
## 5 "\"SILVER\""                  29.2
```

```
#Sort data by tier highest to lowest
tiers <- c(qResult[2,1],qResult[4,1],qResult[5,1],qResult[3,1],qResult[1,1])
tiersData <- c(qResult[2,2],qResult[4,2],qResult[5,2],qResult[3,2],qResult[1,2])
matchduration <- data.frame(tiers,tiersData)

as_tibble(matchduration)
```

```
## # A tibble: 5 x 2
##   tiers          tiersData
##   <fct>              <dbl>
## 1 "\"DIAMOND\""       26.8
## 2 "\"PLATINUM\""      27.7
## 3 "\"SILVER\""        29.2
## 4 "\"GOLD\""          28.5
## 5 "\"BRONZE\""        29.4
```

#Conclusie vraag 7

We zien hier dat in de hogere tiers de potjes gemiddeld net iets minder lang duren dan bij de lagere ranks.

#Vraag 8: What is the most used summoner spell combo?

```
# Get the data
con <- connectDB(username,password)
qResult <- dbGetQuery(con, 'SELECT CASE
                           WHEN M.spell1 < M.spell2 THEN concat((SELECT NAME FROM SPELL WHERE SPELLID = M
                           ELSE concat((SELECT NAME FROM SPELL WHERE SPELLID = M.spell2),
                           END as spellcombo,

                           COUNT(CASE
                           WHEN M.spell1 < M.spell2 THEN concat(M.spell1,M.spell2)
                           ELSE CONCAT(m.spell2,m.spell1)
                           END) AS AANTAL

                           FROM MATCHHISTORY M

                           GROUP BY spellcombo
                           ORDER BY aantal desc
                           FETCH FIRST 5 Rows only
                           ')
as_tibble(qResult)
```

```
## # A tibble: 5 x 2
##   spellcombo    aantal
##   <chr>          <dbl>
## 1 FlashIgnite     2949
## 2 FlashTeleport   1964
## 3 FlashSmite      1703
## 4 FlashHeal       1582
## 5 ExhaustFlash     272
```

#Conclusie vraag 8

Zie bovenstaande tabel

#Vraag 9:What is the least used summoner spell combo?

```
# Get the data
con <- connectDB(username,password)
qResult <- dbGetQuery(con, 'SELECT CASE
                           WHEN M.spell1 < M.spell2 THEN concat((SELECT NAME FROM SPELL WHERE SPELLID = M
                           ELSE concat((SELECT NAME FROM SPELL WHERE SPELLID = M.spell2),
                           END as spellcombo,

                           COUNT(CASE
                           WHEN M.spell1 < M.spell2 THEN concat(M.spell1,M.spell2)
                           ELSE CONCAT(m.spell2,m.spell1)
                           END) AS aantal

                           FROM MATCHHISTORY M

                           GROUP BY spellcombo
                           ORDER BY aantal ASC
                           FETCH FIRST 5 Rows only
                           ')
as_tibble(qResult)
```

```
## # A tibble: 5 x 2
##   spellcombo    aantal
##   <chr>          <dbl>
## 1 SmiteBarrier       1
## 2 CleanseSmite       1
## 3 ExhaustSmite       1
## 4 GhostBarrier       1
## 5 SmiteTeleport      1
```

#Conclusie vraag 9

Zie bovenstaande tabel

#Vraag 10 & 11: What are the top 10 most and least bought items in game?

```
con <- connectDB(username, password)
#Top 10 most/least bought items.

mostBought <- dbGetQuery(con, "SELECT I.ITEMID as ID, I.name as name ,
                           (SELECT COUNT(ITEM0) FROM MATCHHISTORY WHERE I.ITEMID = ITEM0 FETCH FIRST ROW
                           (SELECT COUNT(ITEM1) FROM MATCHHISTORY WHERE I.ITEMID = ITEM1 FETCH FIRST ROW
                           (SELECT COUNT(ITEM2) FROM MATCHHISTORY WHERE I.ITEMID = ITEM2 FETCH FIRST ROW
                           (SELECT COUNT(ITEM3) FROM MATCHHISTORY WHERE I.ITEMID = ITEM3 FETCH FIRST ROW
                           (SELECT COUNT(ITEM4) FROM MATCHHISTORY WHERE I.ITEMID = ITEM4 FETCH FIRST ROW
                           (SELECT COUNT(ITEM5) FROM MATCHHISTORY WHERE I.ITEMID = ITEM5 FETCH FIRST ROW
                           FROM ITEM I ORDER BY count DESC FETCH FIRST 10 ROWS ONLY")


leastBought <- dbGetQuery(con, "SELECT I.ITEMID as ID, I.name as name ,
```

```
                          (SELECT COUNT(ITEM0) FROM MATCHHISTORY WHERE I.ITEMID = ITEM0 FETCH FIRST ROW
                          (SELECT COUNT(ITEM1) FROM MATCHHISTORY WHERE I.ITEMID = ITEM1 FETCH FIRST ROW
                          (SELECT COUNT(ITEM2) FROM MATCHHISTORY WHERE I.ITEMID = ITEM2 FETCH FIRST ROW
                          (SELECT COUNT(ITEM3) FROM MATCHHISTORY WHERE I.ITEMID = ITEM3 FETCH FIRST ROW
                          (SELECT COUNT(ITEM4) FROM MATCHHISTORY WHERE I.ITEMID = ITEM4 FETCH FIRST ROW
                          (SELECT COUNT(ITEM5) FROM MATCHHISTORY WHERE I.ITEMID = ITEM5 FETCH FIRST ROW
                          FROM ITEM I
                          WHERE ((SELECT COUNT(ITEM0) FROM MATCHHISTORY WHERE I.ITEMID = ITEM0 FETCH FI
                          (SELECT COUNT(ITEM1) FROM MATCHHISTORY WHERE I.ITEMID = ITEM1 FETCH FIRST ROW
                          (SELECT COUNT(ITEM2) FROM MATCHHISTORY WHERE I.ITEMID = ITEM2 FETCH FIRST ROW
                          (SELECT COUNT(ITEM3) FROM MATCHHISTORY WHERE I.ITEMID = ITEM3 FETCH FIRST ROW
                          (SELECT COUNT(ITEM4) FROM MATCHHISTORY WHERE I.ITEMID = ITEM4 FETCH FIRST ROW
                          (SELECT COUNT(ITEM5) FROM MATCHHISTORY WHERE I.ITEMID = ITEM5 FETCH FIRST ROW
                          ORDER BY count ASC FETCH FIRST 10 ROWS ONLY")
dbDisconnect(con)
```

```
## [1] TRUE
```

```
as_tibble(mostBought)
```

```
## # A tibble: 10 x 3
##       id name                count
##    <int> <chr>               <dbl>
##  1  3020 Sorcerer's Shoes     1776
##  2  3006 Berserker's Greaves  1636
##  3  3047 Ninja Tabi           1575
##  4  3111 Mercury's Treads     1384
##  5  1055 Doran's Blade        1219
##  6  3117 Boots of Mobility    1035
##  7  2031 Refillable Potion    1003
##  8  3031 Infinity Edge         986
##  9  2055 Control Ward          933
## 10  3157 Zhonya's Hourglass    860
```

```
as_tibble(leastBought)
```

```
## # A tibble: 10 x 3
##       id name                       count
##    <int> <chr>                      <dbl>
##  1  2138 Elixir of Iron                 1
##  2  2140 Elixir of Wrath                1
##  3  3600 Black Spear                    1
##  4  3197 Hex Core mk-2                  1
##  5  3383 Circlet of the Iron Solari     1
##  6  2139 Elixir of Sorcery              1
##  7  3374 Rabadon's Deathcrown           4
##  8  3380 Obsidian Cleaver               4
##  9  3388 Youmuu's Wraithblade           4
## 10  3400 'Your Cut'                     4
```

Conclusie vraag 10 & 11

Zie bovenstaande tabel.