Import Chess Data

Julia Ferris

2023-09-27

Introduction

In this document, the text file containing information about players in a chess tournament is imported and then formatted a specific way. First, the text file is imported. After it is imported, it is formatted into a new data frame that contains all relevant columns. Once the new data frame is completed, it is then output as a csv file. Lastly, the data frame is displayed as a table.

Import Text File and Format Data

In this section, the file is imported and then formatted into a new table

```
test1 <- read.table(file = 'https://raw.githubusercontent.com/juliaDataScience-22/cuny-fall-23/manage-a

test1 <- test1[,-1]
test1 <- test1[,-1]
test1 <- test1[,-1]

test1 <- test1[-149,]
test1 <- test1[-156,]
test1 <- test1[-163,]
test1 <- test1[-164,]
test1 <- test1[-167,]
test1 <- test1[-170,]
test1 <- test1[-173,]
test1 <- test1[-176,]
test1 <- test1[-179,]
test1 <- test1[-182,]
test1 <- test1[-185,]
test1 <- test1[-190,]
test1 <- test1[-191,]
test1 <- test1[-194,]
test1 <- test1[-136,]

library(tidyr)
test1 <- unite(test1, col = 'V9', c(V9, V10), sep="")

test2 <- data.frame(Name = c(1:64), State = c(1:64), Total.Points = c(1:64), Pre.Rating = c(1:64), Avg.

test2$Pre.Rating <- test1$V9[c(seq(6,197,3))]

library(stringr)
```

```
test2[c('Pre.Rating', 'To.Delete')] <- str_split_fixed(test2$Pre.Rating, 'P', 2)

test2 <- test2[,-6]

test1 <- unite(test1, col = 'Names2', c(V6, V7, V8, V9), sep=" ")

library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
test1 <- test1 %>% mutate(Names2 = str_replace_all(Names2, '\\|', ""))



test2$Name <- test1$Names2[c(seq(5, 194, 3))]

test2$State <- test1$V4[c(seq(6, 195, 3))]

indices <- which(test1 == "|6.0" | test1 == "|5.5" | test1=="|5.0" | test1 == "|4.5" | test1 == "|4.0"
indices <- indices[order(indices[,1],decreasing=FALSE),]

num <- 1
for (i in 1:64)
{
  loc <- c(indices[[i]], indices[[i+64]])
  test2$Total.Points[num] <- test1[loc[1],loc[2]]
  num <- num + 1
}

test2[c('Total.Points', 'To Delete')] <- str_split_fixed(test2$Total.Points, "\\|", 2)
test2$Total.Points <- test2$`To Delete`
test2 <- test2[,-6]

count <- 1
for (i in seq(5, 194, 3))
{
  test4 <- test1[i,9:43]
  test4 <- test4 |> mutate_all(na_if,"")
  indices <- which(!is.na(test4), arr.ind=TRUE)
  indices <- indices[-1,]
  indices <- indices[-1,]
  avgRatingOpp <- 0
  players <- 0
  for (num in 1:nrow(indices))
  {
      loc <- indices[[num + nrow(indices)]]
      player1 <- str_split_fixed(test4[loc], "\\|", 2)[1]
      if(player1 != "")
      {
        value1 <- test2$Pre.Rating[as.numeric(player1)]
        avgRatingOpp <- (as.numeric(value1) + avgRatingOpp)
        players <- players + 1
      }
  }
  test2$Avg.Pre.Rating.Opponents[count] <- round(avgRatingOpp / players)
  count <- count + 1
}

write.csv(test2, "chessData.csv", row.names = FALSE)
```

New Data Frame

In this section, a table is created to display the results. To see the table as a csv file, locate the file named chessData.csv.

```
library(gt)
gt(test2) |>
  tab_header(
    title = "Table 1",
    subtitle = "Chess Tournament - Player Information"
    )
```

Table 1

Chess Tournament - Player Information

Name	State	Total.Points	Pre.Rating	Avg.Pre.Rating.Opponents
GARY HUA	ON	6.0	1794	1605
DAKSHESH DARURI	MI	6.0	1553	1469
ADITYA BAJAJ	MI	6.0	1384	1564
PATRICK H SCHILLING	MI	5.5	1716	1574
HANSHI ZUO	MI	5.5	1655	1501
HANSEN SONG	OH	5.0	1686	1519
GARY DEE SWATHELL	MI	5.0	1649	1372
EZEKIEL HOUGHTON	MI	5.0	1641	1468
STEFANO LEE	ON	5.0	1411	1523
ANVIT RAO	MI	5.0	1365	1554
CAMERON WILLIAM MC LEMAN	MI	4.5	1712	1468
KENNETH J TACK	MI	4.5	1663	1506
TORRANCE HENRY JR	MI	4.5	1666	1498
BRADLEY SHAW	MI	4.5	1610	1515
ZACHARY JAMES HOUGHTON	MI	4.5	1220	1484
MIKE NIKITIN	MI	4.0	1604	1386
RONALD GRZEGORCZYK	MI	4.0	1629	1499
DAVID SUNDEEN	MI	4.0	1600	1480

DIPANKAR ROY	MI	4.0	1564	1426
JASON ZHENG	MI	4.0	1595	1411
DINH DANG BUI	ON	4.0	1563	1470
EUGENE L MCCLURE	MI	4.0	1555	1300
ALAN BUI	ON	4.0	1363	1214
MICHAEL R ALDRICH	MI	4.0	1229	1357
LOREN SCHWIEBERT	MI	3.5	1745	1363
MAX ZHU	ON	3.5	1579	1507
GAURAV GIDWANI	MI	3.5	1552	1222
SOFIA ADINA STANESCU-BELLU	MI	3.5	1507	1571
CHIEDOZIE OKORIE	MI	3.5	1602	1314
GEORGE AVERY JONES	ON	3.5	1522	1144
RISHI SHETTY	MI	3.5	1494	1260
JOSHUA PHILIP MATHEWS	ON	3.5	1441	1379
JADE GE	MI	3.5	1449	1277
MICHAEL JEFFERY THOMAS	MI	3.5	1399	1375
JOSHUA DAVID LEE	MI	3.5	1438	1150
SIDDHARTH JHA	MI	3.5	1355	1388
AMIYATOSH PWNANANDAM	MI	3.5	980	1385
BRIAN LIU	MI	3.0	1423	1539
JOEL R HENDON	MI	3.0	1436	1430
FOREST ZHANG	MI	3.0	1348	1391
KYLE WILLIAM MURPHY	MI	3.0	1403	1248
JARED GE	MI	3.0	1332	1150
ROBERT GLEN VASEY	MI	3.0	1283	1107
JUSTIN D SCHILLING	MI	3.0	1199	1327
DEREK YAN	MI	3.0	1242	1152
JACOB ALEXANDER LAVALLEY	MI	3.0	377	1358
ERIC WRIGHT	MI	2.5	1362	1392
DANIEL KHAIN	MI	2.5	1382	1356
MICHAEL J MARTIN	MI	2.5	1291	1286
SHIVAM JHA	MI	2.5	1056	1296
TEJAS AYYAGARI	MI	2.5	1011	1356
ETHAN GUO	MI	2.5	935	1495
JOSE C YBARRA	MI	2.0	1393	1345
LARRY HODGE	MI	2.0	1270	1206
ALEX KONG	MI	2.0	1186	1406
MARISA RICCI	MI	2.0	1153	1414
MICHAEL LU	MI	2.0	1092	1363
VIRAJ MOHILE	MI	2.0	917	1391
SEAN M MC CORMICK	MI	2.0	853	1319
JULIA SHEN	MI	1.5	967	1330
JEZZEL FARKAS	ON	1.5	955	1327
ASHWIN BALAJI	MI	1.0	1530	1186
THOMAS JOSEPH HOSMER	MI	1.0	1175	1350
BEN LI	MI	1.0	1163	1263

Sources (websites)

1. https://tidyr.tidyverse.org/reference/unite.html
2. https://www.rdocumentation.org/packages/base/versions/3.6.2/topics/seq
3. https://www.rdocumentation.org/packages/stringr/versions/0.6.2/topics/str_split_fixed
4. https://www.rdocumentation.org/packages/stringr/versions/0.6.2/topics/str_replace_all

5. https://www.geeksforgeeks.org/which-function-in-r/
6. https://www.tutorialspoint.com/how-to-sort-a-matrix-based-on-one-column-in-r
7. https://stackoverflow.com/questions/1169456/the-difference-between-bracket-and-double-bracket-for-accessing-the-el
8. https://stackoverflow.com/questions/56829690/how-to-use-mutate-all-with-a-function-having-multiple-arguments