

Introduction to R

300958 Social Web Analysis

Week 3 Lab Solutions

- What data structure would you choose to store your past results at UWS? Write code to store your results in the variable `past.results`.

Our past results are tabular, so we would use a data frame.

```
> past.results = data.frame(
+   UnitNumber = c(300700, 300958, 200045),
+   UnitName = c("Statistical Decision Making", "Social Web Analytics",
+               "Quantitative Project"),
+   Mark = c(73, 81, 48)
+ )
> print(past.results)
```

Assignment Project Exam Help

```
UnitNumber
1      300700 Statistica
2      300958      Soc
3      200045      Quantitative Project 48
```

<https://eduassistpro.github.io/>

Add WeChat edu_assist_pro

- Write a for loop to convert a number to a string of plus signs. For example, 5 will be converted to "+++++".

```
> n = 10
> for (a in 1:n) {
+   cat("+")
+ }
```

```
++++++
```

- Write an if statement to test if an item in `past.results` is a pass or fail grade. Then use a for loop to test all of the grades in `past.results`.

```
> n = nrow(past.results)
> for (a in 1:n) {
+   if (past.results$Mark[a] >= 50) {
+     cat(past.results$UnitNumber[a], ":", as.character(past.results$UnitName[a]),
+         ":", "PASS", "\n")
+   }
```

```
+ } else {
+   cat(past.results$UnitNumber[a], ":", as.character(past.results$UnitName[a]),
+       ":", "FAIL", "\n")
+ }
+ }
```

```
300700 : Statistical Decision Making : PASS
300958 : Social Web Analytics : PASS
200045 : Quantitative Project : FAIL
```

- Write a function `check.pass` that takes the data structure `past.results` and returns a vector showing if each grade was a pass or a fail.

```
> check.pass = function(results) {
+   n = nrow(results)
+   passStatus = rep(0,n)
+   for (a in 1:n) {
+     if (results$Mark[a] >= 50) {
+       passStatus[a] = T
+     } else {
+       passStatus[a] = F
+     }
+   }
+   return(passStatus)
+ }
> check.pass(past.results)
```

Assignment Project Exam Help

<https://eduassistpro.github.io/>

Add WeChat edu_assist_pro

```
[1] 1 1 0
```

1 Challenge 1: Finding Force of Will

Load the JSON data into a list and examine the type and categories.

```
> library("jsonlite")
```

```
Warning: package 'jsonlite' was built under R version 3.2.5
```

```
Loading required package: methods
```

```
> mtg = fromJSON("AllSets.json")
> class(mtg)
```

```
[1] "list"
```

```
> names(mtg)
```

```
[1] "LEA"      "LEB"      "ARN"      "2ED"      "CED"      "CEI"
[7] "pDRC"     "ATQ"      "3ED"      "LEG"      "DRK"      "pMEI"
[13] "FEM"      "pLGM"     "4ED"      "ICE"      "CHR"      "HML"
[19] "ALL"      "RQS"      "pARL"     "pCEL"     "MIR"      "MGB"
[25] "ITP"      "VIS"      "5ED"      "pPOD"     "POR"      "VAN"
[31] "WTH"      "pPRE"     "TMP"      "STH"      "PO2"      "pJGP"
[37] "EXO"      "UGL"      "pALP"     "USG"      "ATH"      "ULG"
[43] "6ED"      "PTK"      "UDS"      "S99"      "pGRU"     "pWOR"
[49] "pWOS"     "MMQ"      "BRB"      "pSUS"     "pFNM"     "pELP"
[55] "NMS"      "S00"      "PCY"      "BTD"      "INV"      "PLS"
[61] "7ED"      "pMPR"     "APC"      "ODY"      "DKM"      "TOR"
[67] "JUD"      "ONS"      "LGN"      "SCG"      "pREL"     "8ED"
[73] "MRD"      "DST"      "5DN"      "CHK"      "UNH"      "BOK"
[79] "SOK"      "9ED"      "RAV"      "p2HC"     "pGTW"     "CPT"
[85] "pCMP"     "DIS"      "CSI"     "SST"      "LIP"      "TSB"
[91] "pHHO"     "PLC"      "pPRO"     "pGPX"     "FUT"      "10E"
[97] "pMGD"     "MED"
[103] "p15A"     "SHM"
[109] "pWPN"     "ALA"      "DD2"      "CON"      "DDC"      "ARB"
[115] "M10"      "V09"      "HOP"      "ME3"      "ZEN"
[121] "H09"      "WWK"      "DDF"      "ROE"      "DFA"
[127] "M11"      "V10"      "DDF"      "SOM"      "PD2"      "ME4"
[133] "MBS"      "DDG"      "NPH"      "CMD"      "M12"      "V11"
[139] "DDH"      "ISD"      "PD3"      "DKA"      "DDI"      "AVR"
[145] "PC2"      "M13"      "V12"      "DDJ"      "RTR"      "CM1"
[151] "GTC"      "DDK"      "pWCQ"     "DGM"      "MMA"      "M14"
[157] "V13"      "DDL"      "THS"      "C13"      "BNG"      "DDM"
[163] "JOU"      "MD1"      "CNS"      "VMA"      "M15"      "CPK"
[169] "V14"      "DDN"      "KTK"      "C14"      "DD3_DVD"  "DD3_EVG"
[175] "DD3_GVL"  "DD3_JVC"  "FRF_UGIN" "FRF"      "DDO"      "DTK"
[181] "TPR"      "MM2"      "ORI"      "V15"      "DDP"      "BFZ"
[187] "EXP"      "C15"      "OGW"      "DDQ"      "W16"      "SOI"
[193] "EMA"
```

Assignment Project Exam Help

<https://eduassistpro.github.io/>

Add WeChat [edu_assist_pro](#)

We see that the categories are the three letter set names, we want to examine `EMA`.

```
> class(mtg$EMA)
```

```
[1] "list"
```

```
> names(mtg$EMA)
```

```
[1] "name"      "code"      "releaseDate" "border"
[5] "type"      "booster"   "translations" "cards"
```

The sub-categories are the set details and cards. We want to examine the cards.

```
> class(mtg$EMA$cards)
```

```
[1] "data.frame"
```

```
> names(mtg$EMA$cards)
```

```
[1] "artist"      "cmc"      "colorIdentity" "colors"
[5] "id"          "imageName" "layout"      "manaCost"
[9] "multiverseid" "name"     "number"     "power"
[13] "rarity"      "subtypes" "text"       "toughness"
[17] "type"        "t"
[21] "loyalty"
```

Assignment Project Exam Help

<https://eduassistpro.github.io/>

The cards are stored in a data frame (table). So `mtgEMAcards` contains the name "Force of Will".

Add WeChat edu_assist_pro

```
> pos = which(mtg$EMA$cards$name == "Force of Will")
```

Then find the artist at that position.

```
> mtg$EMA$cards$artist[pos]
```

```
[1] "Terese Nielsen"
```

Or simply print out all of the card details for that row.

```
> mtg$EMA$cards[pos, ]
```

```
      artist cmc colorIdentity colors
49 Terese Nielsen    5             U   Blue
      id      imageName layout manaCost
```

```

49 6c385f9c1ff70244a8bc62b9b8131d262f051b33 force of will normal {3} {U} {U}
    multiverseid      name number power      rarity subtypes
49      413591 Force of Will      49 <NA> Mythic Rare      NULL

text
49 You may pay 1 life and exile a blue card from your hand rather than pay Force of Will's mana
cost.\nCounter target spell.

    toughness      type      types                                flavor supertypes
49      <NA> Instant Instant "I alone determine my destiny."      NULL

    loyalty
49      NA

```

2 Challenge 2: The Game of Life

- Use your knowledge of R from the previous part of this lab to write the missing code, then enjoy the game of life.

```

> count.neighbours <- function(grid, pos) {
+ ## Count the number of living neighbours of a point.
+
+   counter = 0
+   x = pos$x
+   y = pos$y
+
+   ## check the 3x3 grid centred at pos
+   for (x in pos$x + c(-1,0,1)) {
+     for (y in pos$y + c(-1,0,1)) {
+       ## wrap the coordinates, in case the point is at the grid boundary
+       wrap.pos = wrap.coordinates(grid, list(x = x, y = y))
+       ## if there is life, add 1
+       if (inspect(grid, wrap.pos) == 1) {
+         counter = counter + 1
+       }
+     }
+   }
+   wrap.pos = wrap.coordinates(grid, pos)
+   ## remove state of centre point
+   counter = counter - inspect(grid, wrap.pos)
+   return(counter)
+ }

```

Assignment Project Exam Help

<https://eduassistpro.github.io/>

Add WeChat edu_assist_pro