

Untitled

Feiyang Sun's Quiz Hist



q.utoronto.ca

Here is a bit of R code for a dataframe that contains gene identifiers and names, and gene expression values for a control experiment and a stimulated state. Several questions will refer to “myIDs”.

```
myIDs <- data.frame(uID = c("P38903", "P31383", "P47177", NA, "Q07747",
                             "008641", "P47129", "P52910", "P00330", "P81450"),
                        name = c("2A5D", "2AAA", "2NDP", "3HAO", "AAD4",
                               "AB140", "ACF4", "ACS2", "ADH1", "ATP18"),
                        refID = c("NP_014657", "NP_009386",
                                 "NP_012683.1", "NP_012559",
                                 "NP_010038", "XP_014882",
                                 "NP_012616", "NP_013254",
                                 NA, NA),
                        ctrl = c(19.489, 7.776, 9.221, 21.686, 9.585,
                               18.151, 3.251, 12.786, 28.885, 11.026),
                        stim = c(6.841, 2.164, 61.569, 27.792, 96.769,
                               0.001, 18.772, 2069.059, 1989.718, 168.802),
                        stringsAsFactors = FALSE)
rownames(myIDs) <- myIDs$name
```

Unanswered

Question 2

0 / 1.5 pts

Write an R expression that returns all gene names from myIDs that begin with a number.

```
myIDs$name[grep(myIDs$name, pattern = "^[0-9]")]
```

$^{[0-9]}$ is not the same thing as $[^0-9]$. Inside the square brackets it means "Not". Outside the square brackets it means "starts with".

$[^0-9]$ this regular expression will match every name as it doesn't specify position. It just needs to have at least one character that is not

Untitled

Feiyang Sun's Quiz Hist



q.utoronto.ca



Incorrect

Question 3

0 / 1.5 pts

Write an R expression that returns all gene names which have expression greater than 20 in at least one condition.

```
myIDs$name[myIDs$ctrl > 20 | myIDs$stim > 20]
```

```
myIDs$name [which (myIDs$ctrl>20|myIDs$stim>20) ]
```

Incorrect

Question 4

0 / 1.5 pts

Incorrect

Question 4

0 / 1.5 pts

Write an R expression that adds a column of expression changes as fold changes (i.e. log-ratio) values to the myIDs data frame.

```
myIDs$fold_change = NA
```

myIDs\$fold_change = log(myIDs\$stim/myIDs\$ctrl, base=2)

Incorrect

Question 5

0 / 1.5 pts

Incorrect

Question 5

0 / 1.5 pts

Write an R expression that returns all gene names from myIDs for which both UniProt ID and RefSeq ID are available.

```
myIDs$name[]
```

```
myIDs$name[which(!is.na(myIDs$uID) & !is.na(myIDs$refID))]
```

Question 6

1 / 1 pts

You type install.packages("stringr") into the RStudio

Question 6

1 / 1 pts

You type `install.packages("stringr")` into the RStudio console, since you want to use the `str_pad()` function. But when you then type `str_pad("DIVMTQSPSS", 15)`, you get the error: `Error in str_pad("DIVMTQSPSS", 15) : could not find function "str_pad". What happened, how do you fix this?`

```
library("stringr")
```

Unanswered

Unanswered

Question 7

0 / 1 pts

Given:

seq <-

"mssyfvnsfcgrypngpdylhnygdhssvseqfrdsasmhsgrygy

What is one way this string representation of a sequence can be split into a vector of single characters?

Unlist(strsplit(seq, split = ""))

Untitled

Feiyang Sun's Quiz Hist



q.utoronto.ca

You designed the R object tf to store many transcription factors and you initialize it like so:

```
tf <- data.frame( name = "Irx2",
                  species = "Maylandia zebra",
                  taxId = "106582",
                  seqLen = 3884,
                  stringsAsFactors = FALSE)
tf <- rbind(tf, data.frame( name = "Sox2",
                            species = "Gallus gallus",
                            taxId = "208526",
                            seqLen = 1182,
                            stringsAsFactors = FALSE))
tf <- rbind(tf, data.frame( name = "SOX2",
                            species = "Homo Sapiens",
                            taxId = "9606",
                            seqLen = 2512,
                            stringsAsFactors = FALSE))
tf <- rbind(tf, data.frame( name = "Hoxa5",
                            species = "Mus musculus",
                            taxId = "10090",
                            seqLen = 1877,
                            stringsAsFactors = FALSE))
tf <- rbind(tf, data.frame( name = "HOXA1",
                            species = "Homo Sapiens",
                            taxId = "9606",
                            seqLen = 2543,
                            stringsAsFactors = FALSE))
```

Unanswered

Question 8

0 / 1 pts

Write a command to return all names of proteins
shorter than 2,000 residues.

`tf$name[which(tf$seqLen < 2000)]`

Unanswered

Question 9

0 / 1 pts

Write a command to return the number of proteins

Untitled

Feiyang Sun's Quiz Hist



q.utoronto.ca

Unanswered

Question 9

0 / 1 pts

Write a command to return the number of proteins longer than "Hoxa5".

```
Length(which(tf$seqLen > tf$seqLen[which(tf$name=="Hoxa5") ]))
```

Unanswered

Question 10

0 / 1 pts

Write a command that retrieves all distinct species in

Unanswered

Question 10

0 / 1 pts

Write a command that retrieves all distinct species in tf.

Unique(tf\$species)

Unanswered

Question 11

0 / 2 pts

Given:



2026 Winter

Home

Syllabus

Announcements

Assignments

Quizzes

Discussions

Grades

Modules

Library Resources

?

Question 11

0 / 2 pts

Given:

 $x <- 6:1 \quad 6 \ 5 \ 4 \ 3 \ 2 \ 1$

For each of the below check true for the R commands that give the result 3.5.

 `mean(x)` 3.5 `mean(x[])` `mean(x[c(-1,-6)])` 5 4 3 2, 3.5 `mean(x[c(2, 2, 5, 5)])` `sum(x) / length(x)` `mean(x[1:6])` `mean(6:1)``mean(x[c(TRUE, FALSE, FALSE, FALSE, FALSE, TRUE)])`

6 1, 3.5

 `mean(x[x==3]) + 1/2` 3+0.5 `mean(x[x>2]) - 1`

Untitled

Feiyang Sun's Quiz Hist



...



q.utoronto.ca



Vinter

e

us

uncements

nments

res

ssions

es

les

y Resources

Unanswered

Question 12

0 / 1 pts

What does this piece of code set mySeq to?

```
mySeq <- "ABC_12"
```

```
mySeq <- gsub("[^a-zA-Z]", "", mySeq)
```

ABC

ABC_12

_12

12

ABC_1

ous
nouncements
nments
res
ssions
es
iles
y Resources

Unanswered

Question 13

Not yet graded / 2 pts

Describe an abstraction to represent protein – protein interactions

Your Answer:

by column

Given the R code:

```
x <- matrix(c(0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89),  
byrow=FALSE, ncol = 3, 4)
```

... what is the output of:

default: by column

0	3	21
1	5	34
1	8	55
2	13	89

Unanswered

Question 14

0 / 0.2 pts

ous
nouncements
nments
res
ssions
es
iles
y Resources

Unanswered

Question 14

0 / 0.2 pts

1

x[as.integer(TRUE), TRUE]

0 3 21

1 5 34

0 1 1

2 3 5

0

Unanswered

Question 15

0 / 0.2 pts

x[nrow(x), ncol(x)]

ous
nouncements
nments
res
ssions
es
iles
y Resources

Unanswered

Question 15

0 / 0.2 pts

`x[nrow(x), ncol(x)]`



Error in `x[nrow(x), ncol(x)]` : subscript out of bounds



Unanswered

Question 16

0 / 0.2 pts

`x[1, 2]`

ous
nouncements
nments
res
ssions
es
iles
y Resources

Unanswered

Question 16

0 / 0.2 pts

$x[1, 2]$

3

1

2

8

21

Unanswered

Question 17

0 / 0.2 pts

ous
nouncements
nments
res
ssions
es
iles
ry Resources

Unanswered

Question 17

0 / 0.2 pts

 $x[, 2]$ 21 34 55 89 3 5 8 13 1 3 13 55 2 3 5 1 5 34

Unanswered

Question 18

0 / 0.2 pts

cements
ents
ons
s
Resources

Unanswered

Question 18

0 / 0.2 pts

`x[ncol(x), nrow(x)]`



Error in `x[ncol(x), nrow(x)]` : subscript out of bounds

89

13

Unanswered

Question 19

Not yet graded / 1 pts

Given the R code:

```
x <- matrix(c(0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89), ncol  
          = 4)
```

Untitled

Feiyang Sun's Quiz Hist



...

← → ⟳ 🔍

q.utoronto.ca

🔊 📤 📝 ...

Unanswered

Question 19

Not yet graded / 1 pts

Given the R code:

```
x <- matrix(c(0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89), ncol = 3, 4)
```

... by excluding the parameter 'byrow' would the answer to the previous questions be different? If yes, how? If not, why not? How is it better to write the function, with or without the byrow parameter and why?

Your Answer:

Automatically by col

even though it is the default parameter it is always better form to specify the variables, especially variables that will change the value so drastically. Although technically speaking it is not required.

us
ncements
ments
es
ssions
s
es
y Resources

Unanswered

Question 20

0 / 1 pts

Name the different types of identifiers that can be associated with a protein?

uniprot **Q00987**

refseq **NP_001138811, NM_001145339, XP_006719462**

ensembl **ENST0000258149**

HGNC

Hugo Gene ID **HGNC:6973**

MGI **Finally recently retired**

Gene bank AAA60568

Same

Unanswered

Question 21

Not yet graded / 1 pts

refseq ensembl HGNC Hugo Gene ID MGI

Unanswered

Question 21

Not yet graded / 1 pts

We often encounter the idiom `x <- unlist(strsplit(mySeq, ""))`. Why does `strsplit()` need `unlist`? Couldn't it just return a vector?

Your Answer:

Because `strsplit()` is designed to handle a character vector and each string may split into a different-length result, it must return a list, and `unlist()` is only needed when you know there is a single string and want a flat vector.