Week 2 Quiz

5/5 questions correct

Quiz passed!

Continue Course (/learn/data-cleaning/lecture/aqd2Y/subsetting-and-sorting)

Back to Week 2 (/learn/data-cleaning/home/week/2)

1.

Register an application with the Github API here https://github.com /settings/applications (https://github.com/settings/applications). Access the API to get information on your instructors repositories (hint: this is the url you want "https://api.github.com/users/jtleek /repos"). Use this data to find the time that the datasharing repo was created. What time was it created?

This tutorial may be useful (https://github.com/hadley/httr/blob /master/demo/oauth2-github.r (https://github.com/hadley /httr/blob/master/demo/oauth2-github.r)). You may also need to run the code in the base R package and not R studio.

- 2014-03-05T16:11:46Z
- 2013-11-07T13:25:07Z

Well done!

- 2012-06-20T18:39:06Z
- 2013-08-28T18:18:50Z

2.

The sqldf package allows for execution of SQL commands on R data frames. We will use the sqldf package to practice the queries we might send with the dbSendQuery command in RMySQL.

Download the American Community Survey data and load it into an R object called

```
acs
```

https://d396qusza40orc.cloudfront.net/getdata%2Fdata%2Fss06pid.csv (https://d396qusza40orc.cloudfront.net/getdata%2Fdata%2Fss06pid.csv)

Which of the following commands will select only the data for the probability weights pwgtp1 with ages less than 50?

- sqldf("select * from acs where AGEP < 50")
- sqldf("select * from acs")
- sqldf("select pwgtp1 from acs where AGEP < 50")

Well done!

- sqldf("select pwgtp1 from acs")

3.

Using the same data frame you created in the previous problem, what is the equivalent function to unique(acs$AGEP)

- sqldf("select AGEP where unique from acs")
- sqldf("select distinct AGEP from acs")

Well done!

- sqldf("select unique * from acs")
- sqldf("select distinct pwgtp1 from acs")

4.

How many characters are in the 10th, 20th, 30th and 100th lines of HTML from this page:

http://biostat.jhsph.edu/~jleek/contact.html

(Hint: the nchar() function in R may be helpful)

- 45 31 2 25
- 43 99 8 6
- 45 31 7 25

Well done!

- 45 0 2 2
- 43 99 7 25
- 45 92 7 2
- 45 31 7 31

5.

Read this data set into R and report the sum of the numbers in the fourth of the nine columns.

https://d396qusza40orc.cloudfront.net/getdata%2Fwksst8110.for (https://d396qusza40orc.cloudfront.net/getdata%2Fwksst8110.for)

Original source of the data: http://www.cpc.ncep.noaa.gov/data/indices/wksst8110.for (http://www.cpc.ncep.noaa.gov/data/indices/wksst8110.for)

(Hint this is a fixed width file format)

- 222243.1
- 32426.7

Well done!

- 35824.9
- 28893.3
- 101.83
- 36.5