ds105-practice

File: 7a_dataframe_practice.org

README

- Practice file for the lecture on "Subsetting and extracting data frames in R" (lecture and review file in GitHub)
- Create, execute and debug R code blocks as needed
- Emacs + ESS + Org-mode and R must be installed
- You can find the solutions in the PDF repository for the course

TODO IDENTIFY YOURSELF

- Update the `#+AUTHOR:` information in the header
- Add `(pledged)` after your name
- Put your cursor on the headline of this section, and type `S <LEFT>` until you see `DONE` instead of `TODO` next to the title.
- Perform this last step each time you complete a section.

TODO Create data frame

1. Create the data frame shown in the image (link below)
2. Name the dataframe `df`
3. Print the dataframe
4. Print the dataframe's structure information

   tinyurl.com/2hs7dpp4

```
## create dataframe df
```

Solution

```
df <- data.frame (
  "ID" = seq(from=10,to=40,by=10),
  "items" = c("book", "pen", "textbook", "pencil_case"),
  "store" = rep(c(TRUE,FALSE), times=2),
  "price" = c(2.5, 8, 10, 7))
df
str(df)
```

```
 ID       items store price
1 10        book  TRUE   2.5
2 20         pen FALSE   8.0
3 30    textbook  TRUE  10.0
```

```
4 40 pencil_case FALSE   7.0
'data.frame':     4 obs. of  4 variables:
$ ID   : num  10 20 30 40
$ items: chr  "book" "pen" "textbook" "pencil_case"
$ store: logi  TRUE FALSE TRUE FALSE
$ price: num  2.5 8 10 7
```

TODO Select subsets using index operators

Select subsets using the index operators `$` and `[]`:

1. Select rows 1 to 2 using `[]`
2. Select column 2 using `[]` and `$`
3. Select row 3 of column 2 using `[]` and `$`
4. Select rows 1 to 3 in columns 3 and 4 using `[]` and `$`
5. Select rows 2 to 3, and columns 1 and 4 using `[]` and `$`

```
## select subsets using index operators
```

Solution

GitHub image 7[subset.png]

```
df[3,2]     # select row 3 in column 2 with []
df$items[3] # select row 3 in column 2 with $
df[1:2,]    # select rows 1 to 2
df[,1]      # select column 1 with []
df$ID       # select column 1 with $
df[1:3,3:4] # select rows 1 to 3, and columns 3 to 4
df[2:3,c(1,4)] # select rows 2 to 3, and columns 1 and 4
```

```
[1] "textbook"
[1] "textbook"
 ID items store price
1 10  book  TRUE   2.5
2 20   pen FALSE   8.0
[1] 10 20 30 40
[1] 10 20 30 40
 store price
1  TRUE   2.5
2 FALSE   8.0
3  TRUE  10.0
 ID price
2 20     8
3 30    10
```

TODO Select indices and values using `which`, `names`, `%in%`

Select data frame indices using functions `which`, `colnames` and `%in%`:

1. Print the column index for the column `price` using `colnames`
2. Store the column index for `textbook` in `idx`
3. Print the entry `textbook` using `idx`

4. Check if `pencil_case` is a store item using `%in%`

For the last question, the format of the `%in%` matching function to find a string is: `string %in% vector`

```
which(colnames(df)=="price")
idx <- which(df$items=="textbook")
df$items[idx]
"pencil_case" %in% df
```

```
[1] 4
[1] "textbook"
[1] FALSE
```

TODO Select subsets using the `subset` function

Using `subset`,

1. Select product IDs above 20
2. Select all books listed
3. Select all items cheaper than $11 and in stock

```
## select subsets using the subset function
```

Solution

1. Using `subset`:

```
subset(x=df, ID > 20)    # product IDs above 20
subset(x=df, items == "book")  # items called "book"
subset(x=df, (price < 11) & (store == TRUE)) # compound condition
```

```
 ID       items store price
3 30    textbook  TRUE    10
4 40 pencil_case FALSE     7
 ID items store price
1 10  book  TRUE   2.5
 ID    items store price
1 10     book  TRUE   2.5
3 30 textbook  TRUE  10.0
```