Class Activity 3

Your name here

2024-03-20

```
# load the necessary libraries
library(tidyverse)
library(ggplot2)
library(babynames)
```

```
# some interesting data objects
x <- c(3,6,9,5,10)
x.mat <- cbind(x, 2*x)
x.df <- data.frame(x=x,double.x=x*2)
my.list <- list(myVec=x, myDf=x.df, myString=c("hi","bye"))
```

Question 1: data types

- What data type is `x`?

Answer:

```
# code
typeof(x)
```

```
[1] "double"
```

```
typeof(babynames$number)
```

```
[1] "NULL"
```

- What data type is `c(x, x/2)`?

Answer:

```
# code
typeof(c(x, babynames$year))
```

```
[1] "double"
```

- What data type is `c(x,NA)`? What data type is `c(x,"NA")`?

Answer:

```
# code
typeof(c(x, NA))
```

```
[1] "double"
```

```
typeof(c(x, "NA"))
```

```
[1] "character"
```

Question 2: Subsetting and coercion

- How can we reverse the order of entries in `x`?

Answer:

```
# code
rev(x)
```

```
[1] 10  5  9  6  3
```

```
x[length(x):1]
```

```
[1] 10  5  9  6  3
```

- What does `which(x < 5)` equal?

Answer:

```
# code
which(x<5)
```

```
[1] 1
```

- Extract the element of x that corresponds to the location in the preceding question.

Answer:

```
# code
x[which(x<5)]
```

```
[1] 3
```

- What does `sum(c(TRUE,FALSE,TRUE,FALSE))` equal?

Answer:

```
# code
sum(c(TRUE,FALSE,TRUE,FALSE))
```

```
[1] 2
```

- What does `sum(x[c(TRUE,FALSE,TRUE,FALSE)])` equal?

Answer:

```
# code
sum(x[c(TRUE,FALSE,TRUE,FALSE, TRUE)])
```

```
[1] 22
```

- What does `sum(x < 5)` equal?

Answer:

```
# code
sum(x < 5)
```

```
[1] 1
```

- What does `sum(x[x < 5])` equal?

Answer:

```
# code
sum(x[x < 5])
```

```
[1] 3
```

- Why `dim(x.mat[1:2,1])` return `NULL` while `dim(x.mat[1:2,1:2])` returns a dimension?

Answer:

```
# code
dim(x.mat[1:2,1])
```

```
NULL
```

```
dim(x.mat[1:2,1:2])
```

```
[1] 2 2
```

Question 3: Lists

- Using `my.list`, show three ways to write one command that gives the 3rd entry of variable `x` in data frame `myDf`

Answer:

```
# code
my.list[[1]][3]
```

```
[1] 9
```

```
my.list[["myVec"]][3]
```

```
[1] 9
```

```
my.list[1]$myVec[3]
```

```
[1] 9
```

```
my.list$myVec[3]
```

```
[1] 9
```

- What class of object does the command `my.list[3]` return?

Answer:

```
# code
class(my.list[3])
```

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

- What class of object does the command `my.list[[3]]` return?

Answer:

```
# code
class(my.list[[3]])
```

```
[1] "character"
```

- What class of object does the command `unlist(my.list)` return? Why are all the entries `characters`?

Answer:

```
# code
class(unlist(my.list))
```

```
[1] "character"
```