

Lab Week 4

Sasha Uspenskaya

2024-02-11

```
library(tidyverse)

## Warning: package 'tidyr' was built under R version 4.3.2

## — Attaching core tidyverse packages — tidyverse
## 2.0.0 —
## ✓ dplyr      1.1.4      ✓ readr      2.1.5
## ✓ forcats    1.0.0      ✓ stringr    1.5.1
## ✓ ggplot2     3.4.4      ✓ tibble     3.2.1
## ✓ lubridate  1.9.3      ✓ tidyr      1.3.1
## ✓ purrr      1.0.2
## — Conflicts —
tidyverse_conflicts() —
## ✗ dplyr::filter() masks stats::filter()
## ✗ dplyr::lag()     masks stats::lag()
## ⓘ Use the conflicted package (<http://conflicted.r-lib.org/>) to force
all conflicts to become errors

library(dplyr)

x <- list(2, 4, 5, 9, 1)
y <- list(8, 7, 2, 8, 3)
z <- list(1, 8, 5, 4, 2)
```

1) 1) For the lists given above, show and use R code (a map function) to iteratively find the square of the sums across the vectors.

```
pmap_dbl(list(x,y,z), function(first, second, third) (first + second +
third)^2)

## [1] 121 361 144 441 36

tribble( ~Student, ~Gender, ~Salary,
"John", "Male", 65000,
"Alice", "Female", 73000,
"Juan", "Male", 66000,
"Beth", "Female", 71500,
"Denise", "Female", 82000
) -> table
table
```

```
## # A tibble: 5 × 3
##   Student Gender Salary
##   <chr>   <chr>   <dbl>
## 1 John    Male     65000
## 2 Alice   Female   73000
## 3 Juan    Male     66000
## 4 Beth    Female   71500
## 5 Denise  Female   82000
```

2) Using the data table above, use and show R code that will output a statement that is descriptive for all rows of the data table.

```
table %>%
  pmap_chr(~ str_glue("{..1} who is a {..2}, has a salary that is {..3}
dollars per year."))

## [1] "John who is a Male, has a salary that is 65000 dollars per year."
## [2] "Alice who is a Female, has a salary that is 73000 dollars per year."
## [3] "Juan who is a Male, has a salary that is 66000 dollars per year."
## [4] "Beth who is a Female, has a salary that is 71500 dollars per year."
## [5] "Denise who is a Female, has a salary that is 82000 dollars per year."
```

3) Write a nested loop that will produce a 5 by 5 matrix that whose matrix elements are sums of the corresponding columns and rows. Your code should produce the matrix below:

```
z <- matrix( nrow = 5, ncol = 5)
for (m in 1:5) {
  for (n in 1:5) {
    z[m, n] <- 1*(m + n)
  }
}
print(z)

##      [,1] [,2] [,3] [,4] [,5]
## [1,]    2    3    4    5    6
## [2,]    3    4    5    6    7
## [3,]    4    5    6    7    8
## [4,]    5    6    7    8    9
## [5,]    6    7    8    9   10
```

4) Use for loop coding to produce the number sequence shown below: Note that the numbers 5 and 10 are missing

```
x <- 1:20
for (val in x) {
  if (val == 5){
```

```
    next
  }
  if (val==10){
    next
  }
  print(val)
}
```

```
## [1] 1
## [1] 2
## [1] 3
## [1] 4
## [1] 6
## [1] 7
## [1] 8
## [1] 9
## [1] 11
## [1] 12
## [1] 13
## [1] 14
## [1] 15
## [1] 16
## [1] 17
## [1] 18
## [1] 19
## [1] 20
```