### Listado de preguntas teóricas

1. Para las siguientes sentencias de `base R`, liste su contraparte de `dplyr`:

\* `str()`

\* `df[,c("a","b")]`

\* `names(df)[4] <- "new\_name"` donde la posición 4 corresponde a la variable `old\_name`

\* `df[df$variable == "valor",]`

2. Al momento de filtrar en SQL, ¿cuál keyword cumple las mismas funciones que el keyword `OR` para filtrar uno o más elementos una misma columna?

\*\*el keyword que usa en SQl es WHERE "en donde" y con ese se pueden filtrar dato en una misma columna.junto con este se usa AND y OR\*\*

3. ¿Por qué en R utilizamos funciones de la familia apply (lapply,vapply) en lugar de utilizar ciclos?

\*\*El usar aplicaciones de funciones en vez de loops en R es una decision que se toma ya que R esta diseñado para funcionar con estas funciones. Y tambien acepta usar loops pero su enfoque principal es usar sapply,vaply,lapply y demas.\*\*

4. ¿Cuál es la diferencia entre utilizar `==` y `=` en R?

\*\*"==" significa que va a comprobar o dictar que los elementos sean igual, po ejemplo 3 == 3 devolveria True , mienstras que 3 == 4 devolveria FALSE. Mienstras que "=" se usa para asignar un valor a una variable. Es por preferencia per normalmente en vez de usar el "=" se usa "<-" para que no se confunda.\*\*

5. ¿Cuál es la forma correcta de cargar un archivo de texto donde el delimitador es `:`?

6. ¿Qué es un vector y en qué se diferencia en una lista en R?

7. ¿Qué pasa si quiero agregar una nueva categoría a un factor que no se encuentra en los niveles existentes?

\*\*si queremos agregar una nueva categoria a un factor el cual no esta listado , silenciosamente nos va a correr sin problmeas pero va a combertir el nuevo dato en un NA\*\*

8. Si en un dataframe, a una variable de tipo `factor` le agrego un nuevo elemento que \*no se encuentra en los niveles existentes\*, ¿cuál sería el resultado esperado y por qué?

\* El nuevo elemento

\* `NA`

9. En SQL, ¿para qué utilizamos el keyword `HAVING`?

\*\*El key word de HAVING lo utilizamos ya que WHERE no se puede utilizar para hacer agregaciones. Entonces para poder filtrar por agregaciones necesitamos HAVING\*\*

10. Si quiero obtener como resultado las filas de la tabla A que no se encuentran en la tabla B, ¿cómo debería de completar la siguiente sentencia de SQL?

\* SELECT \* FROM A \_\_\_\_\_\_\_ B ON A.KEY = B.KEY WHERE \_\_\_\_\_\_\_\_\_\_ = \_\_\_\_\_\_\_\_\_\_