



## Guía 10 - Análisis de datos científicos en R

1. Usar `str_length()` y `str_sub()` para extraer el caracter del medio de un *string*. Si el *string* tiene un número par de caracteres, extraer el caracter inmediato anterior.
2. Dado los datos en *words* (contenidos en *stringr*), crear expresiones regulares para encontrar todas las palabras que:
  - a. Empiecen con “y”.
  - b. Terminen con “x”
  - c. Tengan tres letras. (No usar `str_length()`)
  - d. **POSGRADO** – Tengan más de 7 letras. Como esta lista es larga, pueden usar el argumento `match` en `str_view` para mostrar solo las palabras que correspondientes o las que no lo son.
3. En el vector *words*:
  - a. ¿Qué palabra tiene la mayor cantidad de vocales?
  - b. ¿Qué palabra tiene la mayor proporción de vocales?
4. ¿Qué hace el argumento `tzzone` en `today()`? ¿Porqué es importante?
5. Usar la función apropiada de `lubridate` para procesar las siguientes fechas:

```
d1 <- "January 1, 2010"
d2 <- "2015-Mar-07"
d3 <- "06-Jun-2017"
d4 <- c("August 19 (2015)", "July 1 (2015)")
d5 <- "12/30/14" # Dec 30, 2014
```
6. **POSGRADO** – Sobre el dataframe de `nycflights13` ¿En qué día de la semana habría que despegar si queremos minimizar la chance de demora?
7. Escribir una función que dado un cumpleaños (como un objeto tipo *date*) nos devuelva cuántos años tiene la persona.