

# Tarea14

Yimmy Eman

2022-07-11

## Pregunta 1

¿Tibble o no tibble? ¿Cómo sabes si un objeto es una tibble o no? Pista: imprime los objetos mtcars por un lado y nycflights13::flights que son respectivamente un data frame y una tibble.

```
class(mtcars)
```

```
## [1] "data.frame"
```

```
class(flights)
```

```
## [1] "tbl_df"      "tbl"        "data.frame"
```

## Pregunta 2

Compara y contrasta las siguientes operaciones en el data frame y su equivalente en tibble. - ¿En qué se parecen? - ¿En qué difieren? - ¿Por qué a veces el data frame por defecto nos puede causar mucha frustración?

```
df <- data.frame(abc = 1, xyz = "a")
df$x
```

```
## [1] "a"
```

```
df[, "xyz"]
```

```
## [1] "a"
```

```
df[, c("abc", "xyz")]
```

```
##   abc xyz
## 1   1   a
```

R: Usar \$ en un df muestra los nombres de las columnas. Veo que df\$x devuelve lo mismo que df\$xyz, pero puede tender a confundir si se mezcla con otra variable.

## Pregunta 3

Si tenemos el nombre de una variable almacenada en un objeto tipo string (por ejemplo `myvar <- "mpg"`), ¿cómo podemos extraer la variable referenciada de una tibble? ¿Y en un data frame?

`df[[var]]`

## Pregunta 4

Toma la siguiente tibble formada por variables con nombres no sintácticos.

```
df <- tibble(  
  `1` = 1:12,  
  `2` = `1` * 2 + `1`*runif(length(`1`)))
```

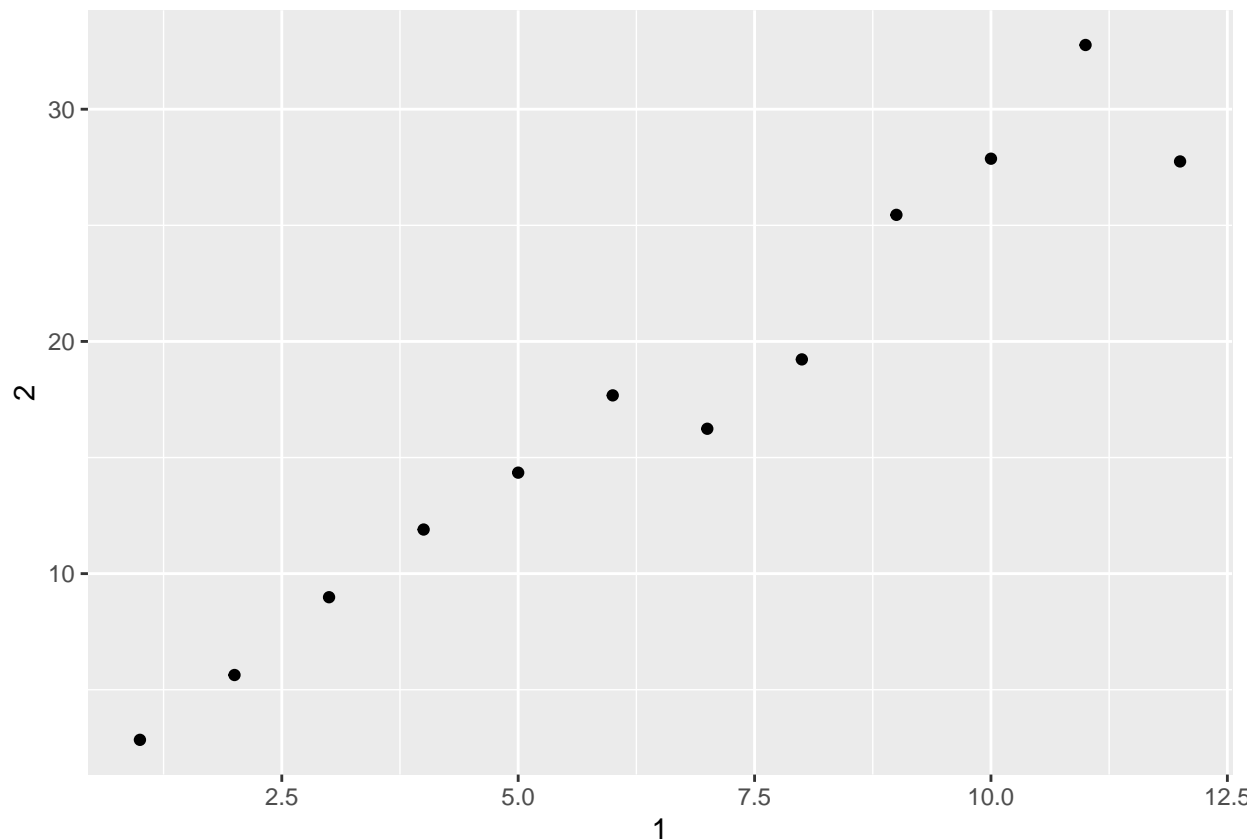
- Extrae el valor de la variable 1

```
df$`1`
```

```
## [1] 1 2 3 4 5 6 7 8 9 10 11 12
```

- Haz un scatterplot de la variable 1 contra la variable 2

```
df %>%  
  ggplot(aes(`1`, `2`)) +  
  geom_point()
```



- Crea una nueva columna llamada 3 que sea el cociente de 2 entre 1.

```
df <- df %>%
  mutate(`3` = `2` / `1`)
```

- Renombra las columnas para que se llamen x, y, z respectivamente.

```
df <- df %>%
  rename(
    x = `1`,
    y = `2`,
    z = `3`
  )
```

- ¿Qué nombre crees que es mejor?

## Pregunta 5

Investiga acerca de la función `tibble:enframe()` y `tibble:deframe()`. ¿Qué hace y para qué puede servirte?

`enframe()` converts named atomic vectors or lists to one- or two-column data frames. For a list, the result will be a nested tibble with a column of type list. For unnamed vectors, the natural sequence is used as name column.

`deframe()` converts two-column data frames to a named vector or list, using the first column as name and the second column as value. If the input has only one column, an unnamed vector is returned.

## Pregunta 6

¿Cómo podemos controlar cuántos nombres de columna adicionales se imprimen en el footer de una tibble?

Con la función `print` de tibble.