

Printing tibbles works differently than dataframes

```
tibble(  
  a = lubridate::now() + runif(1e3) * 86400,  
  b = lubridate::today() + runif(1e3) * 30,  
  c = 1:1e3,  
  d = runif(1e3),  
  e = sample(letters, 1e3, replace = TRUE)  
)
```

```
#> # A tibble: 1,000 x 5  
#>   a                b                c          d e  
#>   <dtm>           <date>         <int> <dbl> <chr>  
#> 1 2020-01-15 20:43:23 2020-01-22      1 0.368 n  
#> 2 2020-01-16 14:48:32 2020-01-27      2 0.612 l  
#> 3 2020-01-16 09:12:12 2020-02-06      3 0.415 p  
#> 4 2020-01-15 22:33:29 2020-02-05      4 0.212 m  
#> 5 2020-01-15 18:57:45 2020-02-02      5 0.733 i  
#> 6 2020-01-16 05:58:42 2020-01-29      6 0.460 n  
#> # ... with 994 more rows
```

Subsetting tibbles works differently than dataframes

```
df <- tibble(  
  x = runif(5),  
  y = rnorm(5)  
)  
  
# Extract by name  
df$x  
#> [1] 0.7330 0.2344 0.6604 0.0329 0.4605  
df[["x"]]  
#> [1] 0.7330 0.2344 0.6604 0.0329 0.4605  
  
# Extract by position  
df[[1]]  
#> [1] 0.7330 0.2344 0.6604 0.0329 0.4605
```

Compared to a `data.frame`, tibbles are more strict: they never do partial matching, and they will generate a warning if the column you are trying to access does not exist.