1. Les bases à connaître en R
   1. .Rproj
   2. R de base : codage efficace  
      Voir Writing Efficient R Code
   3. Quand créer une fonction
2. Packages à connaître
   1. stringr
   2. lubridate
   3. data.table / dplyr
3. Data.table
   1. Base
   2. .N, .I, .SD, .BY…
   3. Multi Thread
4. R Markdown
5. Créer des packages
   1. usethis
   2. Base à connaître
   3. Défis CRAN

# Notes Datacamp.com

**Writing Efficient R Code**

Benchmarking

How good is your machine

Memory allocation

**First rule of R club : never, ever grow a vector**

Importance of vectorizing your code

**Second rule of R club : use a vectorized solution wherever possible**

Data frames and matrices

Select columns vs rows : image intéressante expliquant pourquoi c’est plus lent de prendre les lignes que les colonnes. C’est ce qu’on a appris dans le cours OSA Ordinateur Structure et Application. C’est plus lent parce qu’on doit trouver dans la mémoire où chaque élément de chaque colonne comment dans la mémoire

Les matrices sont plus performantes parce que toutes les colonnes ont le même type de données. Donc rechercher une ligne fonctionne bien parce que le *saut/bon* que l’ordinateur fait est constant

**Third rule of R club : Use a matrix whenever appropriate**

What is code profiling

Profvis: Larger example

Monopoly recap

CPUs – why do we have more than one

What sort of problems benefit from parallel computing?

The parallel package – parApply

The parallel package – parSapply