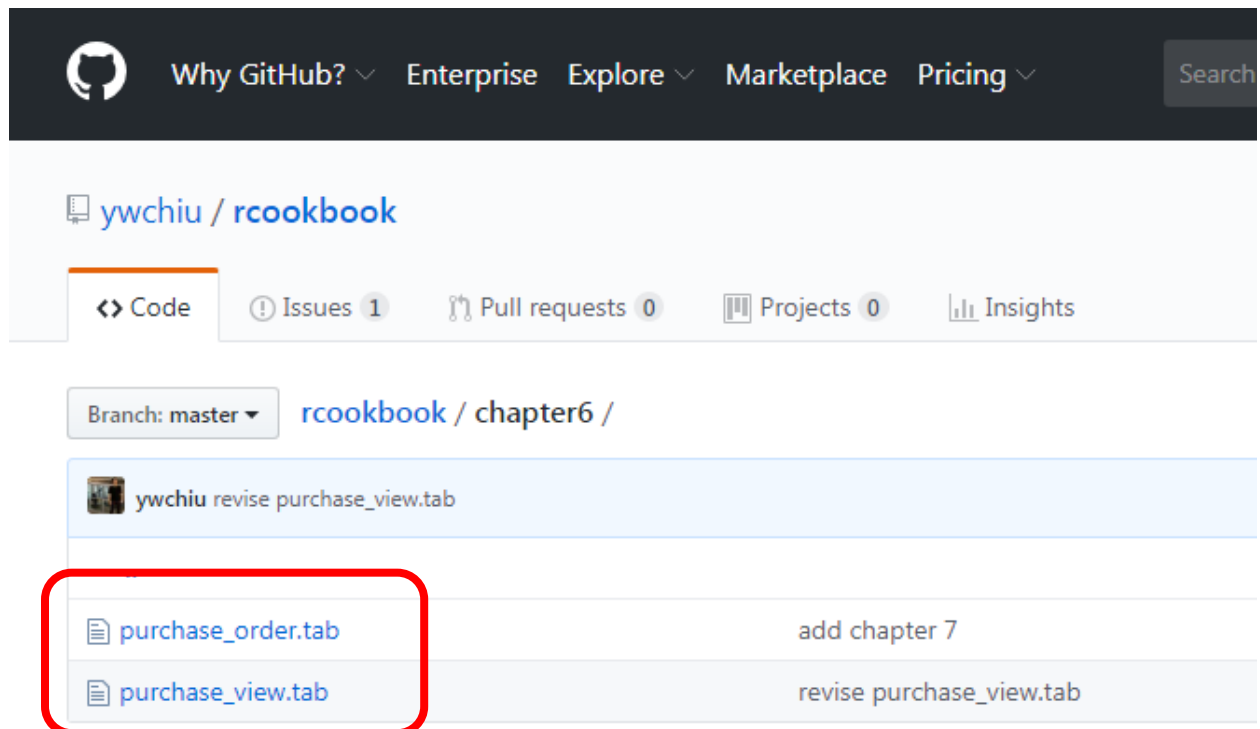


Pendahuluan

Data frame memiliki struktur yang fleksibel dan ekspresif untuk jenis data tabular (tabel). Namun demikian, data frame hanya efektif untuk data yang ukurannya kecil. Data frame tidak ideal digunakan untuk pemrosesan data yang ukurannya besar. Untuk mengatasi hal tersebut maka digunakan *data.table* untuk pemrosesan data. Dalam materi ke-3 ini, kita akan belajar menggunakan *data.table* untuk manipulasi, join, dan pemrosesan data.

1. Meningkatkan performa *data.frame* dengan *data.table*

Download data “purchase_order” dan “purchase_view” pada link: <https://github.com/ywchiu/rcookbook/tree/master/chapter6>



- a. Install dan load packages *data.table*

```
install.packages("data.table")  
library(data.table)
```

- b. Baca data menggunakan cara biasa (menggunakan fungsi *read.table* bawaan R)

```
melihat = read.table("purchase_view.tab", header = TRUE, sep = "\t")
pesan = read.table("purchase_order.tab", header = TRUE, sep = "\t")
```

- c. Baca dengan menggunakan fungsi *fread* dari packages *data.table*

```
melihat2 = fread("purchase_view.tab", header = TRUE, sep = "\t")
pesan2 = fread("purchase_order.tab", header = TRUE, sep = "\t")
```

- d. Bandingkan kecepatan bacanya

```
system.time(melihat <- read.table("purchase_view.tab",
                                header = TRUE, sep = "\t"))
system.time(melihat2 <- fread("purchase_view.tab",
                              header = TRUE, sep = "\t"))
```

- e. Bandingkan tipe datanya

```
str(melihat)
str(melihat2)
```

- f. Merubah Class *data.frame* ke *data.table*, misal merubah data *melihat* (*data.frame*) kedalam format *data.table* kemudian menyimpannya dengan nama *melihat_transform*

```
melihat_transform = as.data.table(melihat)
```

2. Bekerja dengan Data Table

Berdasarkan data yang telah dibuat, yaitu data type; data frame dan data table, kita akan membandingkan penggunaan kode dalam mengolah data.

- a. Melihat data ter-atas

```
head(melihat[1:3])
head(melihat2[1:3])
```

- b. Select data berdasar baris dan kolom

```
melihat[1:3, "User"]
melihat2[1:3, User]
```

- c. Select data berdasar baris dan multi-kolom

```
melihat[1:3, c("User", "Product")]
melihat2[1:3, .(User, Product)]
```

- d. Melakukan filter data, misal memfilter *data pesan* yang memiliki nilai diatas 80000 pada kolom *Price*

```
pesan[pesan$Price>80000,"Price"]  
pesan2[Price>80000,Price]
```

- e. Membuat data baru (tanpa NA pada kolom *Price*) dari *data pesan*

Untuk membuat dan menghapus kolom baru pada type data table, digunakan “ := ”

- a. Membuat kolom baru, misal dengan nama *Rt2_Price* yang merupakan pembagian nilai kolom *Price* dengan *Quantity*

```
pesan2_tnp_NA[,Rt2_Price:= Price/Quantity]
```

- b. Menghapus kolom, misal menghapus kolom *Rt2_Price*

```
pesan2_tnp_NA[,Rt2_Price:=NULL]
```

Untuk mengecek apakah suatu data sama atau tidak, maka digunakan fungsi *identical*, misal kita bandingkan data *pesan* dan *pesan2*

```
identical(pesan,pesan2)
```

3. Mencoba Olah Data Table

Kita akan mencoba penggunaan data.table dalam melakukan analisa data.

- a. Melihat rata-rata harga (Price) pada data *pesan2*

```
pesan2[,mean(Price)]
```

- b. Melihat rata2 harga (Price) pada data *pesan2* tanpa nilai NA

```
pesan2[,mean(na.omit(Price))]
```

- c. data rata-rata pemasukan dari tiap user

```
rata2pesan_user = pesan2[,mean(na.omit(Price)), User]
```

atau cara lain, agar ada nama kolom

```
rata2pesan_user = pesan2[,.(mean_belanja=mean(na.omit(Price))),User]
```

- d. data rata-rata pemasukan tiap hari

```
rata2pesan_hari = pesan2[,.(mean_pemasukan=mean(na.omit(Price))),  
                           by=as.Date(Time)]
```

- e. data jumlah pemasukan dan jumlah pemesanan tiap hari

```
jml_belanja_pemesan = pesan2[,.(jml_pemasukan = sum(na.omit(Price)),  
                                jml_pemesan = length(unique(User))),  
                                by=as.Date(Time)]
```

- f. data rata-rata pemasukan per-user tiap hari

```
rata2_belanja_hari = pesan2[,.(rata2_pemasukan_peruser=sum(na.omit(Price))/uniqueN(User)),  
                              by=as.Date(Time)]
```

- g. data total pemasukan berdasar tanggal dan produk

```
rata2_pemasukan_produk = pesan2[,.(total_pemasukan=na.omit(sum(Price*Quantity))),  
                                   by=(as.Date(Time), Product)]
```

4. Menggabungkan Data

Penggabungan data pada data table masih menggunakan fungsi yang sama dengan data frame, hanya saja ada sedikit berbeda pada penulisan kodenya

- a. data frekuensi produk dipesan

```
Frek_Produk_dibeli = pesan2[,.(Frek_dibeli = length(Action)),by=Product]
```

- b. data frekuensi produk dilihat

```
Frek_Produk_dilihat = melihat2[,.(Frek_dilihat = length(Action)),by=Product]
```

- c. Gabungan produk dipesan dan dilihat

```
data_gabung = merge(Frek_Produk_dibeli,Frek_Produk_dilihat,by="Product")
```

- d. abungan produk dipesan dan dilihat (ada NA)

```
data_gabung_full = merge(Frek_Produk_dibeli,Frek_Produk_dilihat,  
                        by="Product",all = TRUE)
```