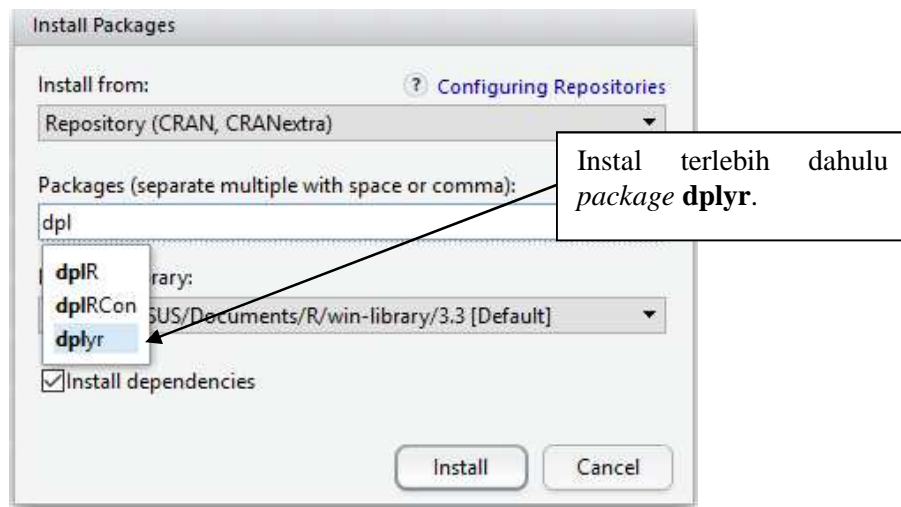


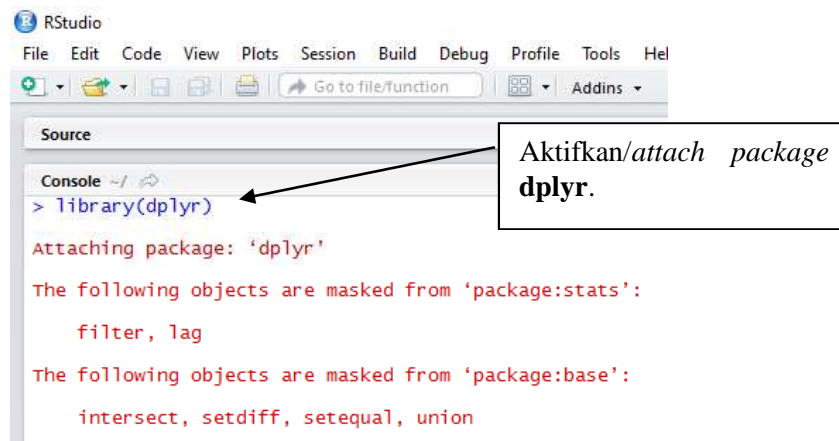
BAB 12

MEMANIPULASI DATA DENGAN FUNGSI DPLYR

Pada Bab 12 ini, disajikan langkah demi langkah mengenai memanipulasi data dengan fungsi *dplyr*(). Sebelum menggunakan fungsi *dplyr*(), terlebih dahulu telah diinstal *package dplyr* dan mengaktifkan *package dplyr* (perhatikan Gambar 12.1 dan Gambar 12.2).

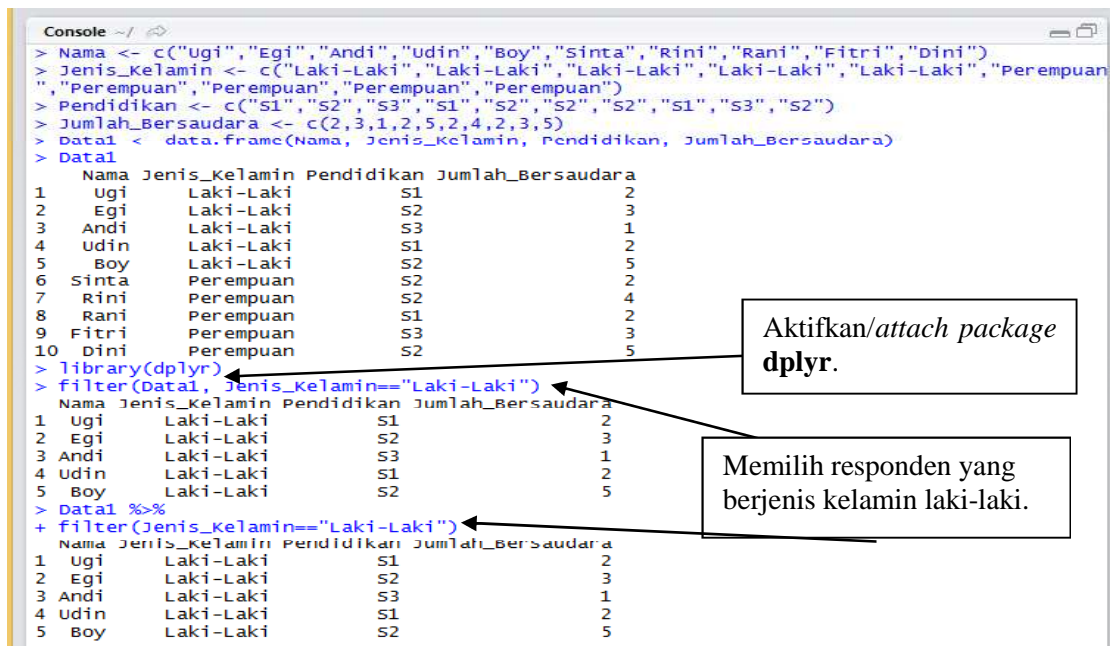


Gambar 12.1



Gambar 12.2

Praktek Ke-1: Menggunakan Fungsi *filter()*



```
> Nama <- c("Ugi", "Egi", "Andi", "Udin", "Boy", "Sinta", "Rini", "Rani", "Fitri", "Dini")
> Jenis_kelamin <- c("Laki-Laki", "Laki-Laki", "Laki-Laki", "Laki-Laki", "Laki-Laki", "Perempuan",
", "Perempuan", "Perempuan", "Perempuan")
> Pendidikan <- c("S1", "S2", "S3", "S1", "S2", "S2", "S2", "S1", "S3", "S2")
> Jumlah_Bersaudara <- c(2,3,1,2,5,2,4,2,3,5)
> Data1 <- data.frame(Nama, Jenis_kelamin, Pendidikan, Jumlah_Bersaudara)
> Data1
  Nama Jenis_kelamin Pendidikan Jumlah_Bersaudara
1  Ugi      Laki-Laki      S1             2
2  Egi      Laki-Laki      S2             3
3  Andi     Laki-Laki      S3             1
4  Udin     Laki-Laki      S1             2
5  Boy      Laki-Laki      S2             5
6  Sinta    Perempuan      S2             2
7  Rini     Perempuan      S2             4
8  Rani     Perempuan      S1             2
9  Fitri    Perempuan      S3             3
10 Dini     Perempuan      S2             5
> library(dplyr)
> filter(Data1, Jenis_kelamin=="Laki-Laki")
  Nama Jenis_kelamin Pendidikan Jumlah_Bersaudara
1  Ugi      Laki-Laki      S1             2
2  Egi      Laki-Laki      S2             3
3  Andi     Laki-Laki      S3             1
4  Udin     Laki-Laki      S1             2
5  Boy      Laki-Laki      S2             5
> Data1 %>%
+ filter(Jenis_kelamin=="Laki-Laki")
  Nama Jenis_kelamin Pendidikan Jumlah_Bersaudara
1  Ugi      Laki-Laki      S1             2
2  Egi      Laki-Laki      S2             3
3  Andi     Laki-Laki      S3             1
4  Udin     Laki-Laki      S1             2
5  Boy      Laki-Laki      S2             5
```

Aktifkan/attach package dplyr.

Memilih responden yang berjenis kelamin laki-laki.

Gambar 12.1

Berdasarkan data Gambar 12.1:

- ⇒ Terdapat 4 kolom atau variabel, yakni **Nama**, **Jenis_Kelamin**, **Pendidikan**, dan **Jumlah_Bersaudara**.
- ⇒ Jumlah responden yang diamati sebanyak 10 responden.
- ⇒ Sebagai contoh responden ke-1, bernama Ugi, dengan jenis kelamin laki-laki, pendidikan S1, jumlah bersaudara 2. Responden ke-10 bernama Dini, dengan jenis kelamin perempuan, pendidikan S2, jumlah bersaudara 5.

"Tugas:

- ⇒ **Gunakan fungsi *filter()* untuk memilih hanya responden berjenis kelamin laki-laki.**
- ⇒ **Kemudian tampilkan informasi dari variabel nama, jenis kelamin, pendidikan, dan jumlah bersaudara."**

Perlu diperhatikan bahwa sebelum menggunakan fungsi *filter()*, terlebih dahulu memasang (*attach*) package **dplyr**. Fungsi *filter()* tersedia dalam package **dplyr**.

Perintah R

```
library(dplyr)
filter(Data1, Jenis_kelamin=="Laki-Laki")
```

Alternatif Perintah R

```
Data1 %>%
filter(Jenis_kelamin=="Laki-Laki")
```

Praktek Ke-2: Menggunakan Fungsi `filter()`

```
Console -1
> Data1
  Nama Jenis_Kelamin Pendidikan Jumlah_Bersaudara
1  Ugi      Laki-Laki      S1           2
2  Egi      Laki-Laki      S2           3
3  Andi     Laki-Laki      S3           1
4  Udin     Laki-Laki      S1           2
5  Boy      Laki-Laki      S2           5
6  Sinta    Perempuan      S2           2
7  Rini     Perempuan      S2           4
8  Rani     Perempuan      S1           2
9  Fitri    Perempuan      S3           3
10 Dini     Perempuan      S2           5
> filter(Data1, Pendidikan=="S2" | Pendidikan=="S3")
  Nama Jenis_Kelamin Pendidikan Jumlah_Bersaudara
1  Egi      Laki-Laki      S2           3
2  Andi     Laki-Laki      S3           1
3  Boy      Laki-Laki      S2           5
4  Sinta    Perempuan      S2           2
5  Rini     Perempuan      S2           4
6  Fitri    Perempuan      S3           3
7  Dini     Perempuan      S2           5
> Data1 %>%
+ filter(Pendidikan=="S2" | Pendidikan=="S3")
  Nama Jenis_Kelamin Pendidikan Jumlah_Bersaudara
1  Egi      Laki-Laki      S2           3
2  Andi     Laki-Laki      S3           1
3  Boy      Laki-Laki      S2           5
4  Sinta    Perempuan      S2           2
5  Rini     Perempuan      S2           4
6  Fitri    Perempuan      S3           3
7  Dini     Perempuan      S2           5
> |
```

Memilih responden dengan pendidikan S2 atau S3.

Gambar 12.2

"Tugas:

- ⇒ Gunakan fungsi `filter()` untuk memilih responden dengan pendidikan S2 atau S3.
- ⇒ Kemudian tampilkan informasi dari variabel nama, jenis kelamin, pendidikan, dan jumlah bersaudara."

Perhatikan bahwa simbol "`|`" berarti "atau".

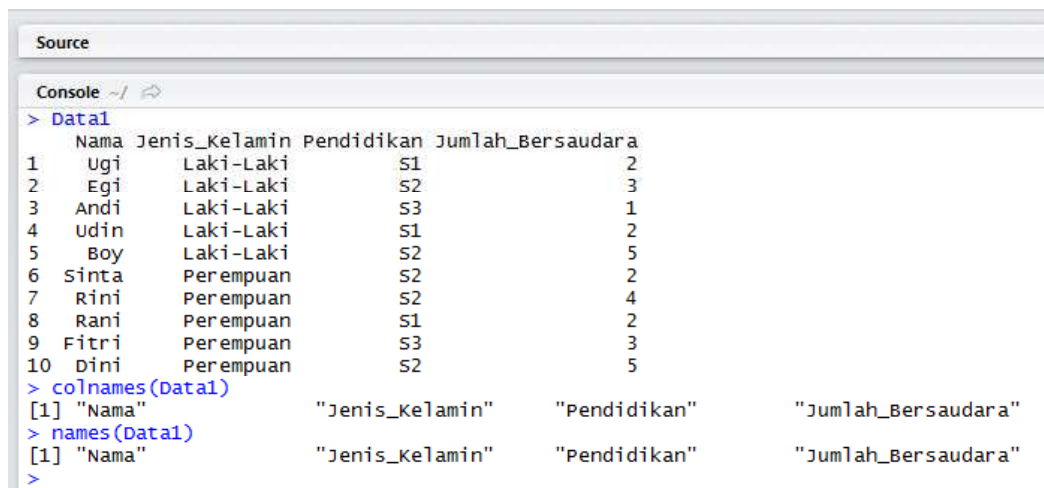
Perintah R

```
filter(Data1, Pendidikan=="S2" | Pendidikan=="S3")
```

Alternatif Perintah R

```
Data1 %>%
filter(Pendidikan=="S2" | Pendidikan=="S3")
```

Praktek Ke-3: Menggunakan Fungsi `names()` dan `colnames()`



```
Source
Console ~/ / ↵
> Data1
  Nama Jenis_Kelamin Pendidikan Jumlah_Bersaudara
1  Ugi      Laki-Laki         S1             2
2  Egi      Laki-Laki         S2             3
3  Andi     Laki-Laki         S3             1
4  Udin     Laki-Laki         S1             2
5  Boy      Laki-Laki         S2             5
6  Sinta    Perempuan          S2             2
7  Rini     Perempuan          S2             4
8  Rani     Perempuan          S1             2
9  Fitri    Perempuan          S3             3
10 Dini     Perempuan          S2             5
> colnames(Data1)
[1] "Nama"          "Jenis_Kelamin" "Pendidikan"    "Jumlah_Bersaudara"
> names(Data1)
[1] "Nama"          "Jenis_Kelamin" "Pendidikan"    "Jumlah_Bersaudara"
>
```

Gambar 12.3

"Tugas:

Gunakan fungsi `colnames()` atau `names()` untuk menampilkan nama-nama kolom atau variabel pada *data frame* `Data1`."

Berdasarkan Gambar 12.3, diketahui terdapat 4 kolom atau variabel dengan nama, **Nama**, **Jenis_Kelamin**, **Pendidikan**, dan **Jumlah_Bersaudara**. Fungsi `names()` dan `colnames()` dapat digunakan untuk menampilkan nama-nama dari variabel atau kolom pada *data frame*.

Perintah R

```
colnames(Data1)
```

Alternatif Perintah R

```
names(Data1)
```

Praktek Ke-4: Menggunakan Fungsi *length()*

```
Console ~/ |
> Data1
  Nama Jenis_Kelamin Pendidikan Jumlah_Bersaudara
1  Ugi      Laki-Laki      S1           2
2  Egi      Laki-Laki      S2           3
3  Andi     Laki-Laki      S3           1
4  Udin     Laki-Laki      S1           2
5  Boy      Laki-Laki      S2           5
6  Sinta    Perempuan      S2           2
7  Rini     Perempuan      S2           4
8  Rani     Perempuan      S1           2
9  Fitri    Perempuan      S3           3
10 Dini     Perempuan      S2           5
> length(Data1)
[1] 4
> |
```

Gambar 12.4

"Tugas:

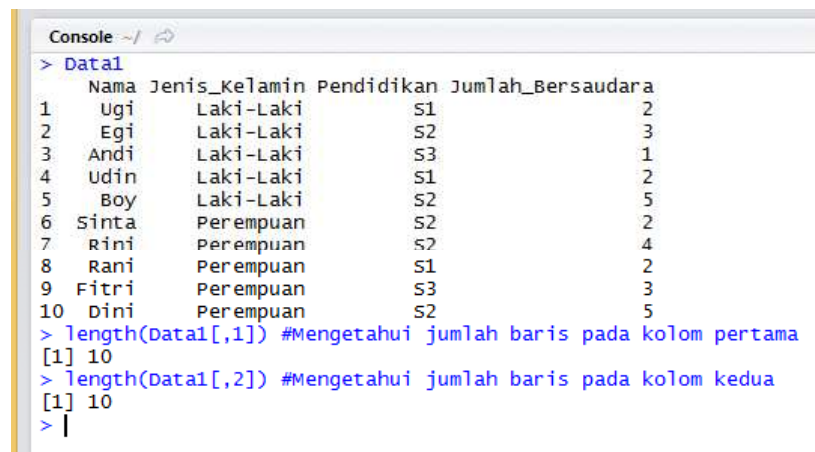
Gunakan fungsi *length()* untuk mengetahui jumlah variabel atau kolom pada *data frame* *Data1*."

Berdasarkan Gambar 12.4, diketahui jumlah kolom atau variabel sebanyak 4, yakni **Nama**, **Jenis_Kelamin**, **Pendidikan**, dan **Jumlah_Bersaudara**. Fungsi *length()* dapat digunakan untuk mengetahui jumlah kolom atau variabel pada *data frame*.

Perintah R

```
length(Data1)
```

Praktek Ke-5: Menggunakan Fungsi *length()*



```
Console ~/ / ↻
> Data1
  Nama Jenis_Kelamin Pendidikan Jumlah_Bersaudara
1  Ugi      Laki-Laki      S1           2
2  Egi      Laki-Laki      S2           3
3  Andi     Laki-Laki      S3           1
4  Udin     Laki-Laki      S1           2
5  Boy      Laki-Laki      S2           5
6  Sinta    Perempuan      S2           2
7  Rini     Perempuan      S2           4
8  Rani     Perempuan      S1           2
9  Fitri    Perempuan      S3           3
10 Dini     Perempuan      S2           5
> length(Data1[,1]) #Mengetahui jumlah baris pada kolom pertama
[1] 10
> length(Data1[,2]) #Mengetahui jumlah baris pada kolom kedua
[1] 10
> |
```

Gambar 12.5

"Tugas:

Gunakan fungsi *length()* untuk mengetahui jumlah baris dari masing-masing kolom atau variabel pada *data frame* Data1."

Berdasarkan Gambar 12.5, digunakan fungsi *length()* untuk mengetahui jumlah baris dari masing-masing kolom.

Perintah R

```
length(Data1[,1]) #Mengetahui jumlah baris pada kolom pertama
```

```
length(Data1[,2]) #Mengetahui jumlah baris pada kolom kedua
```

Praktek Ke-6: Menggunakan Fungsi `length()` dan `na.omit()`

```
Console ~/\n> Nama <- c("Ugi", "Egi", "Andi", "Udin", "Boy", "Sinta", "Rini", "Rani", "Fitri", "Dini")\n> Jenis_kelamin <- c("Laki-Laki", "Laki-Laki", "Laki-Laki", "Laki-Laki", "Laki-Laki", "Perempuan",\n", "Perempuan", "Perempuan", "Perempuan")\n> Nilai_Matematika <- c(75, NA, 45, 40, 50, 75, 70, 80, 90, 80)\n> Nilai_Kimia <- c(50, 70, 75, NA, 50, 85, NA, 90, NA, 70)\n> Nilai_Biologi <- c(NA, 50, NA, NA, 75, 80, 40, 80, NA, 90)\n> Data1 <- data.frame(Nama, Jenis_kelamin, Nilai_Matematika, Nilai_Kimia, Nilai_Biologi)\n> Data1\n  Nama Jenis_kelamin Nilai_Matematika Nilai_Kimia Nilai_Biologi\n1  Ugi      Laki-Laki           75           50           NA\n2  Egi      Laki-Laki           NA           70           50\n3  Andi     Laki-Laki           45           75           NA\n4  Udin     Laki-Laki           40           NA           NA\n5  Boy      Laki-Laki           50           50           75\n6  Sinta    Perempuan           75           85           80\n7  Rini     Perempuan           70           NA           40\n8  Rani     Perempuan           80           90           80\n9  Fitri    Perempuan           90           NA           NA\n10 Dini     Perempuan           80           70           90\n> length(Data1[,3])\n[1] 10\n> length(na.omit(Data1[,3]))\n[1] 9\n> length(Data1[,4])\n[1] 10\n> length(na.omit(Data1[,4]))\n[1] 7\n> length(Data1[,5])\n[1] 10\n> length(na.omit(Data1[,5]))\n[1] 6\n> |
```

Gambar 12.6

"Tugas:

Gunakan fungsi `length()` dan `na.omit()` untuk mengetahui jumlah baris yang tidak kosong (NA) dari masing-masing variabel atau kolom pada *data frame* Data1."

Berdasarkan Gambar 12.6:

- ⇒ Diketahui jumlah responden yang ada nilai matematika-nya sebanyak 9. NA berarti *not available*, yang berarti tidak ada.
- ⇒ Diketahui jumlah responden yang ada nilai kimia-nya sebanyak 7.
- ⇒ Diketahui jumlah responden yang ada nilai biologi-nya sebanyak 6.

Penggunaan fungsi `na.omit()` akan mengeliminasi NA.

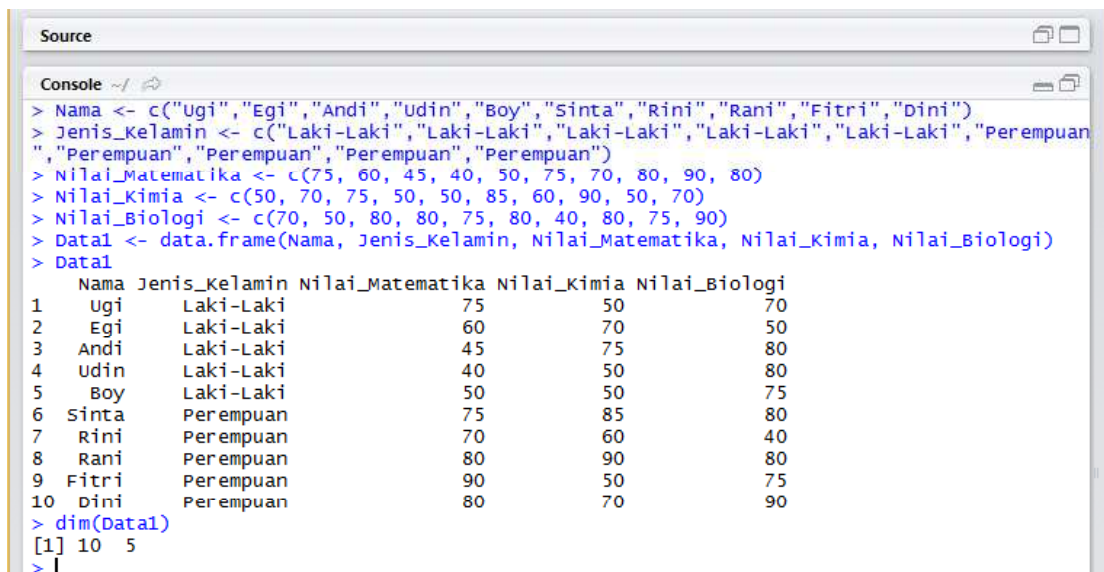
Perintah R

```
length(na.omit(Data1[,3])) #Mengetahui jumlah responden yang ada nilai matematikanya
```

```
length(na.omit(Data1[,4])) #Mengetahui jumlah responden yang ada nilai kimianya
```

```
length(na.omit(Data1[,5])) #Mengetahui jumlah responden yang ada nilai biologinya
```

Praktek Ke-7: Menggunakan Fungsi *dim()*



```
Source
Console
> Nama <- c("Ugi", "Egi", "Andi", "Udin", "Boy", "Sinta", "Rini", "Rani", "Fitri", "Dini")
> Jenis_kelamin <- c("Laki-Laki", "Laki-Laki", "Laki-Laki", "Laki-Laki", "Laki-Laki", "Perempuan", "Perempuan", "Perempuan", "Perempuan")
> Nilai_Matematika <- c(75, 60, 45, 40, 50, 75, 70, 80, 90, 80)
> Nilai_kimia <- c(50, 70, 75, 50, 50, 85, 60, 90, 50, 70)
> Nilai_Biologi <- c(70, 50, 80, 80, 75, 80, 40, 80, 75, 90)
> Data1 <- data.frame(Nama, Jenis_kelamin, Nilai_Matematika, Nilai_Kimia, Nilai_Biologi)
> Data1
  Nama Jenis_kelamin Nilai_Matematika Nilai_Kimia Nilai_Biologi
1  Ugi      Laki-Laki             75           50             70
2  Egi      Laki-Laki             60           70             50
3  Andi     Laki-Laki             45           75             80
4  Udin     Laki-Laki             40           50             80
5  Boy      Laki-Laki             50           50             75
6  Sinta    Perempuan             75           85             80
7  Rini     Perempuan             70           60             40
8  Rani     Perempuan             80           90             80
9  Fitri    Perempuan             90           50             75
10 Dini     Perempuan             80           70             90
> dim(Data1)
[1] 10 5
> |
```

Gambar 12.7

"Tugas:

Gunakan fungsi *dim()* untuk mengetahui jumlah baris dan kolom pada *data frame* *Data1*."

Berdasarkan Gambar 12.7, diketahui jumlah baris (responden) sebanyak 10 dan jumlah kolom sebanyak 5. Fungsi *dim()* dapat digunakan untuk mengetahui jumlah baris dan kolom pada *data frame*.

Perintah R

`dim(Data1)`

Praktek Ke-8: Menggunakan Fungsi `factor()` dan Fungsi `levels()`

```
Console ~/ |
> Data1
  Nama Jenis_kelamin Pendidikan Jumlah_Bersaudara
1  Ugi      Laki-Laki      S1             2
2  Egi      Laki-Laki      S2             3
3  Andi     Laki-Laki      S3             1
4  Udin     Laki-Laki      S1             2
5  Boy      Laki-Laki      S2             5
6  Sinta    Perempuan      S2             2
7  Rini     Perempuan      S2             4
8  Rani     Perempuan      S1             2
9  Fitri    Perempuan      S3             3
10 Dini     Perempuan      S2             5
> levels(factor(Data1[,2]))
[1] "Laki-Laki" "Perempuan"
> levels(factor(Data1[,3]))
[1] "S1" "S2" "S3"
> levels(factor(Data1[,4]))
[1] "1" "2" "3" "4" "5"
> |
```

Gambar 12.8

"Tugas:

Gunakan fungsi `levels()` dan `factor()` untuk mengetahui *level* dari suatu variabel."

Berdasarkan Gambar 12.8, diketahui variabel **Jenis_Kelamin** terdiri dari 2 *level*, yakni "Laki-Laki" dan "Perempuan". Diketahui variabel **Pendidikan** terdiri dari 3 *level*, yakni "S1", "S2", dan "S3". Pada variabel **Jumlah_Bersaudara** terdiri dari 5 *level*, yakni 1, 2, 3, 4, dan 5. Kombinasi dari fungsi `factor()` dan `levels()` dapat digunakan untuk menampilkan *level* dari suatu vektor.

Perintah R

```
levels(factor(Data1[,2]))
```

```
levels(factor(Data1[,3]))
```

```
levels(factor(Data1[,4]))
```

Praktek Ke-9: Menggunakan Fungsi *filter()*

```
Console ~/ |
> Data1
  Nama Jenis_kelamin Pendidikan Jumlah_Bersaudara
1  Ugi      Laki-Laki      S1           2
2  Egi      Laki-Laki      S2           3
3  Andi     Laki-Laki      S3           1
4  Udin     Laki-Laki      S1           2
5  Boy      Laki-Laki      S2           5
6  Sinta    Perempuan      S2           2
7  Rini     Perempuan      S2           4
8  Rani     Perempuan      S1           2
9  Fitri    Perempuan      S3           3
10 Dini     Perempuan      S2           5
> filter(Data1, Jenis_kelamin=="Perempuan" & Pendidikan=="S2")
  Nama Jenis_kelamin Pendidikan Jumlah_Bersaudara
1  Sinta    Perempuan      S2           2
2  Rini     Perempuan      S2           4
3  Dini     Perempuan      S2           5
> Data1 %>%
+ filter(Jenis_kelamin=="Perempuan",Pendidikan=="S2")
  Nama Jenis_kelamin Pendidikan Jumlah_Bersaudara
1  Sinta    Perempuan      S2           2
2  Rini     Perempuan      S2           4
3  Dini     Perempuan      S2           5
> |
```

Gambar 12.9

"Tugas:

- ⇒ Gunakan fungsi *filter()* untuk memilih responden berjenis kelamin perempuan dan berpendidikan S2.
- ⇒ Kemudian tampilkan informasi dari variabel nama, jenis kelamin, pendidikan, dan jumlah bersaudara."

Simbol "&" berarti "dan".

Perintah R

```
filter(Data1, Jenis_kelamin=="Perempuan" & Pendidikan=="S2")
```

Alternatif Perintah R

```
Data1 %>%
filter(Jenis_kelamin=="Perempuan", Pendidikan=="S2")
```

Praktek Ke-10: Menggunakan Fungsi `filter()`

```
Console ~/   
> Data1   
  Nama Jenis_Kelamin Pendidikan Jumlah_Bersaudara   
 1 Ugi Laki-Laki S1 2   
 2 Egi Laki-Laki S2 3   
 3 Andi Laki-Laki S3 1   
 4 Udin Laki-Laki S1 2   
 5 Boy Laki-Laki S2 5   
 6 Sinta Perempuan S2 2   
 7 Rini Perempuan S2 4   
 8 Rani Perempuan S1 2   
 9 Fitri Perempuan S3 3   
 10 Dini Perempuan S2 5   
> filter(Data1, Jenis_Kelamin=="Perempuan" & (Pendidikan=="S2" | Pendidikan=="S3") & Jumlah_Bersaudara > 2 )   
  Nama Jenis_Kelamin Pendidikan Jumlah_Bersaudara   
 1 Rini Perempuan S2 4   
 2 Fitri Perempuan S3 3   
 3 Dini Perempuan S2 5   
> Data1 %>%   
 + filter(Jenis_Kelamin=="Perempuan" & (Pendidikan=="S2" | Pendidikan=="S3") & Jumlah_Bersaudara > 2 )   
  Nama Jenis_Kelamin Pendidikan Jumlah_Bersaudara   
 1 Rini Perempuan S2 4   
 2 Fitri Perempuan S3 3   
 3 Dini Perempuan S2 5   
> |
```

Gambar 12.10

"Tugas:

- ⇒ Gunakan fungsi `filter()` untuk memilih responden berjenis kelamin perempuan, berpendidikan S2 atau S3, dengan jumlah bersaudara di atas 2.
- ⇒ Kemudian tampilkan informasi dari variabel nama, jenis kelamin, pendidikan, dan jumlah bersaudara."

Perintah R

```
filter(Data1, Jenis_Kelamin=="Perempuan" & (Pendidikan=="S2" | Pendidikan=="S3") & Jumlah_Bersaudara > 2 )
```

Alternatif Perintah R

```
Data1 %>%   
 filter(Jenis_Kelamin=="Perempuan" & (Pendidikan=="S2" | Pendidikan=="S3") &   
 Jumlah_Bersaudara > 2 )
```

Praktek Ke-11: Menggunakan Fungsi `select()`

```
Console ~/ |
> Data1
> Data1
  Nama Jenis_Kelamin Pendidikan Jumlah_Bersaudara
1  Ugi      Laki-Laki      S1           2
2  Egi      Laki-Laki      S2           3
3  Andi     Laki-Laki      S3           1
4  Udin     Laki-Laki      S1           2
5  Boy      Laki-Laki      S2           5
6  Sinta    Perempuan      S2           2
7  Rini     Perempuan      S2           4
8  Rani     Perempuan      S1           2
9  Fitri    Perempuan      S3           3
10 Dini     Perempuan      S2           5
> select(Data1, Nama)
  Nama
1  Ugi
2  Egi
3  Andi
4  Udin
5  Boy
6  Sinta
7  Rini
8  Rani
9  Fitri
10 Dini
> Data1 %>%
+ select(Nama)
  Nama
1  Ugi
2  Egi
3  Andi
4  Udin
5  Boy
6  Sinta
7  Rini
8  Rani
9  Fitri
10 Dini
> |
```

Gambar 12.11

"Tugas:

- ⇒ Gunakan fungsi `select()` untuk menampilkan informasi hanya pada variabel nama."

Pada Gambar 12.11, digunakan fungsi `select()` untuk hanya menampilkan variabel nama.

Perintah R

```
select(Data1, Nama)
```

Alternatif Perintah R

```
Data1 %>%
select(Nama)
```

Praktek Ke-12: Menggunakan Fungsi *select()*

```
Console -/
> Data1
  Nama Jenis_kelamin Pendidikan Jumlah_Bersaudara
1  Ugi      Laki-Laki      S1           2
2  Egi      Laki-Laki      S2           3
3  Andi     Laki-Laki      S3           1
4  Udin     Laki-Laki      S1           2
5  Boy      Laki-Laki      S2           5
6  Sinta    Perempuan      S2           2
7  Rini     Perempuan      S2           4
8  Rani     Perempuan      S1           2
9  Fitri    Perempuan      S3           3
10 Dini     Perempuan      S2           5
> select(Data1, Nama, Pendidikan)
  Nama Pendidikan
1  Ugi      S1
2  Egi      S2
3  Andi     S3
4  Udin     S1
5  Boy      S2
6  Sinta    S2
7  Rini     S2
8  Rani     S1
9  Fitri    S3
10 Dini     S2
> Data1 %>% select(Nama, Pendidikan)
  Nama Pendidikan
1  Ugi      S1
2  Egi      S2
3  Andi     S3
4  Udin     S1
5  Boy      S2
6  Sinta    S2
7  Rini     S2
8  Rani     S1
9  Fitri    S3
10 Dini     S2
>
```

Gambar 12.12

"Tugas:

⇒ Gunakan fungsi *select()* untuk menampilkan informasi hanya pada variabel **Nama** dan **Pendidikan**."

Berdasarkan Gambar 12.12, digunakan fungsi *select()*, untuk hanya menampilkan variabel **Nama** dan **Pendidikan**.

Perintah R

```
select(Data1, Nama, Pendidikan)
```

Alternatif Perintah R

```
Data1 %>% select(Nama, Pendidikan)
```

Praktek Ke-13: Kombinasi Penggunaan Fungsi *filter()* dan *select()*

```
Console -/ |
> Data1
  Nama Jenis_Kelamin Pendidikan Jumlah_Bersaudara
1  Ugi      Laki-Laki      S1          2
2  Egi      Laki-Laki      S2          3
3  Andi     Laki-Laki      S3          1
4  Udin     Laki-Laki      S1          2
5  Boy      Laki-Laki      S2          5
6  Sinta    Perempuan      S2          2
7  Rini     Perempuan      S2          4
8  Rani     Perempuan      S1          2
9  Fitri    Perempuan      S3          3
10 Dini     Perempuan      S2          5
> filter(select(Data1, Nama, Pendidikan), Pendidikan=="S1")
  Nama Pendidikan
1  Ugi      S1
2  Udin     S1
3  Rani     S1
> Data1 %>%
+ filter(Pendidikan=="S1") %>%
+ select(Nama, Pendidikan)
  Nama Pendidikan
1  Ugi      S1
2  Udin     S1
3  Rani     S1
> |
```

Gambar 12.13

"Tugas:

- ⇒ Gunakan fungsi *filter()* untuk memilih responden berpendidikan S1.
- ⇒ Gunakan fungsi *select()* untuk menampilkan informasi hanya pada variabel nama dan pendidikan."

Perintah R

```
filter(select(Data1, Nama, Pendidikan), Pendidikan=="S1")
```

Alternatif Perintah R

```
Data1 %>%
filter(Pendidikan=="S1") %>%
select(Nama, Pendidikan)
```

Praktek Ke-14: Penggunaan Fungsi *filter()*, *select()*, dan *arrange()*

```
Console ~/ |
> Data1
  Nama Jenis_Kelamin Pendidikan Jumlah_Bersaudara
1  Ugi      Laki-Laki      S1           2
2  Egi      Laki-Laki      S2           3
3  Andi     Laki-Laki      S3           1
4  Udin     Laki-Laki      S1           2
5  Boy      Laki-Laki      S2           5
6  Sinta   Perempuan     S2           2
7  Rini     Perempuan     S2           4
8  Rani     Perempuan     S1           2
9  Fitri   Perempuan     S3           3
10 Dini     Perempuan     S2           5
> arrange(filter(select(Data1, Nama, Pendidikan, Jumlah_Bersaudara), Pendidikan=="S2" | Pendidikan=="S3"), desc(Jumlah_Bersaudara))
  Nama Pendidikan Jumlah_Bersaudara
1  Boy      S2           5
2  Dini     S2           5
3  Rini     S2           4
4  Egi      S2           3
5  Fitri   S3           3
6  Sinta   S2           2
7  Andi     S3           1
> Data1 %>%
+ filter(Pendidikan=="S2" | Pendidikan=="S3") %>%
+ select(Nama, Pendidikan, Jumlah_Bersaudara) %>%
+ arrange(desc(Jumlah_Bersaudara))
  Nama Pendidikan Jumlah_Bersaudara
1  Boy      S2           5
2  Dini     S2           5
3  Rini     S2           4
4  Egi      S2           3
5  Fitri   S3           3
6  Sinta   S2           2
7  Andi     S3           1
> |
```

Gambar 12.14

"Tugas:

- ⇒ Gunakan fungsi *filter()* untuk memilih responden berpendidikan S2 atau S3.
- ⇒ Gunakan fungsi *select()* untuk menampilkan informasi hanya pada variabel nama, pendidikan, dan jumlah bersaudara.
- ⇒ Gunakan fungsi *arrange()* untuk mengurutkan data berdasarkan variabel jumlah bersaudara.
- ⇒ Gunakan fungsi *desc()* untuk mengurutkan data dari yang terbesar, sampai terkecil."

Perintah R

```
arrange(filter(select(Data1, Nama, Pendidikan, Jumlah_Bersaudara), Pendidikan=="S2" | Pendidikan=="S3"), desc(Jumlah_Bersaudara))
```

Alternatif Perintah R

```
Data1 %>%
filter(Pendidikan=="S2" | Pendidikan=="S3") %>%
select(Nama, Pendidikan, Jumlah_Bersaudara) %>%
arrange(desc(Jumlah_Bersaudara))
```

Praktek Ke-15: Menggunakan Fungsi *mutate()*

```
Console -/ ↗
> Nama <- c("Ugi", "Egi", "Andi", "Udin", "Boy", "Sinta", "Rini", "Rani", "Fitri", "Dini")
> Jenis_Kelamin <- c("Laki-Laki", "Laki-Laki", "Laki-Laki", "Laki-Laki", "Laki-Laki", "Perempuan", "Perempuan", "Perempuan", "Perempuan")
> Nilai_Matematika <- c(75, 60, 45, 40, 50, 75, 70, 80, 90, 80)
> Nilai_Kimia <- c(50, 70, 75, 50, 50, 85, 60, 90, 50, 70)
> Nilai_Biologi <- c(70, 50, 80, 80, 75, 80, 40, 80, 75, 90)
> Data1 <- data.frame(Nama, Jenis_Kelamin, Nilai_Matematika, Nilai_Kimia, Nilai_Biologi)
> Data1
  Nama Jenis_Kelamin Nilai_Matematika Nilai_Kimia Nilai_Biologi
1  Ugi      Laki-Laki           75           50           70
2  Egi      Laki-Laki           60           70           50
3  Andi     Laki-Laki           45           75           80
4  Udin     Laki-Laki           40           50           80
5  Boy      Laki-Laki           50           50           75
6  Sinta    Perempuan           75           85           80
7  Rini     Perempuan           70           60           40
8  Rani     Perempuan           80           90           80
9  Fitri    Perempuan           90           50           75
10 Dini     Perempuan           80           70           90
>
> mutate(Data1, Rata_Rata = (Nilai_Matematika + Nilai_Kimia + Nilai_Biologi)/3)
  Nama Jenis_Kelamin Nilai_Matematika Nilai_Kimia Nilai_Biologi Rata_Rata
1  Ugi      Laki-Laki           75           50           70  65.00000
2  Egi      Laki-Laki           60           70           50  60.00000
3  Andi     Laki-Laki           45           75           80  66.66667
4  Udin     Laki-Laki           40           50           80  56.66667
5  Boy      Laki-Laki           50           50           75  58.33333
6  Sinta    Perempuan           75           85           80  80.00000
7  Rini     Perempuan           70           60           40  56.66667
8  Rani     Perempuan           80           90           80  83.33333
9  Fitri    Perempuan           90           50           75  71.66667
10 Dini     Perempuan           80           70           90  80.00000
> |

>
> mutate(Data1, Rata_Rata = (Nilai_Matematika + Nilai_Kimia + Nilai_Biologi)/3)
  Nama Jenis_Kelamin Nilai_Matematika Nilai_Kimia Nilai_Biologi Rata_Rata
1  Ugi      Laki-Laki           75           50           70  65.00000
2  Egi      Laki-Laki           60           70           50  60.00000
3  Andi     Laki-Laki           45           75           80  66.66667
4  Udin     Laki-Laki           40           50           80  56.66667
5  Boy      Laki-Laki           50           50           75  58.33333
6  Sinta    Perempuan           75           85           80  80.00000
7  Rini     Perempuan           70           60           40  56.66667
8  Rani     Perempuan           80           90           80  83.33333
9  Fitri    Perempuan           90           50           75  71.66667
10 Dini     Perempuan           80           70           90  80.00000
> Data1 %>%
+ mutate(Rata_Rata = (Nilai_Matematika + Nilai_Kimia + Nilai_Biologi)/3)
  Nama Jenis_Kelamin Nilai_Matematika Nilai_Kimia Nilai_Biologi Rata_Rata
1  Ugi      Laki-Laki           75           50           70  65.00000
2  Egi      Laki-Laki           60           70           50  60.00000
3  Andi     Laki-Laki           45           75           80  66.66667
4  Udin     Laki-Laki           40           50           80  56.66667
5  Boy      Laki-Laki           50           50           75  58.33333
6  Sinta    Perempuan           75           85           80  80.00000
7  Rini     Perempuan           70           60           40  56.66667
8  Rani     Perempuan           80           90           80  83.33333
9  Fitri    Perempuan           90           50           75  71.66667
10 Dini     Perempuan           80           70           90  80.00000
> |
```

Gambar 12.15

"Tugas:

- ⇒ Gunakan fungsi *mutate()* untuk membentuk variabel baru bernama rata-rata, berdasarkan nilai ujian matematika, kimia, dan biologi, dari masing-masing responden.
- ⇒ Nilai rata-rata dihitung dengan rumus:

$$\text{rata - rata} = \frac{\text{nilai matematika} + \text{nilai kimia} + \text{nilai biologi}}{3}$$

⇒ Kemudian tampilkan informasi dari variabel nama, jenis kelamin, nilai matematika, nilai kimia, nilai biologi, dan nilai rata-rata.”

Perintah R

```
mutate(Data1, Rata_Rata = (Nilai_Matematika + Nilai_Kimia +  
Nilai_Biologi)/3)
```

Alternatif Perintah R

```
Data1 %>%  
mutate(Rata_Rata = (Nilai_Matematika + Nilai_Kimia + Nilai_Biologi)/3)
```

Praktek Ke-16: Menggunakan Fungsi *mutate()* dan *ifelse()*

```
Console ~/ |
> Data1
  Nama Jenis_Kelamin Nilai_Matematika Nilai_Kimia Nilai_Biologi
1 Ugi Laki-Laki 75 50 70
2 Egi Laki-Laki 60 70 50
3 Andi Laki-Laki 45 75 80
4 Udin Laki-Laki 40 50 80
5 Boy Laki-Laki 50 50 75
6 Sinta Perempuan 75 85 80
7 Rini Perempuan 70 60 40
8 Rani Perempuan 80 90 80
9 Fitri Perempuan 90 50 75
10 Dini Perempuan 80 70 90
> mutate(Data1, Kelulusan_Matematika = ifelse(Nilai_Matematika<60, "TL", "L"))
  Nama Jenis_Kelamin Nilai_Matematika Nilai_Kimia Nilai_Biologi Kelulusan_Matematika
1 Ugi Laki-Laki 75 50 70 L
2 Egi Laki-Laki 60 70 50 L
3 Andi Laki-Laki 45 75 80 TL
4 Udin Laki-Laki 40 50 80 TL
5 Boy Laki-Laki 50 50 75 TL
6 Sinta Perempuan 75 85 80 L
7 Rini Perempuan 70 60 40 L
8 Rani Perempuan 80 90 80 L
9 Fitri Perempuan 90 50 75 L
10 Dini Perempuan 80 70 90 L
> Data1 %>%
+ mutate(Kelulusan_Matematika = ifelse(Nilai_Matematika<60, "TL", "L"))
  Nama Jenis_Kelamin Nilai_Matematika Nilai_Kimia Nilai_Biologi Kelulusan_Matematika
1 Ugi Laki-Laki 75 50 70 L
2 Egi Laki-Laki 60 70 50 L
3 Andi Laki-Laki 45 75 80 TL
4 Udin Laki-Laki 40 50 80 TL
5 Boy Laki-Laki 50 50 75 TL
6 Sinta Perempuan 75 85 80 L
7 Rini Perempuan 70 60 40 L
8 Rani Perempuan 80 90 80 L
9 Fitri Perempuan 90 50 75 L
10 Dini Perempuan 80 70 90 L
> |
```

Gambar 12.16

"Tugas:

- ⇒ Gunakan fungsi *mutate()* untuk membentuk variabel baru bernama kelulusan matematika.
- ⇒ Gunakan fungsi *ifelse()* untuk membuat kriteria dari variabel kelulusan matematika sebagai berikut.
 - Jika nilai matematika < 60, maka tidak lulus "TL".
 - Jika nilai matematika > 60, maka lulus "L".

Perintah R

```
mutate(Data1, Kelulusan_Matematika = ifelse(Nilai_Matematika<60, "TL", "L"))
```

Alternatif Perintah R

```
Data1 %>%
mutate(Kelulusan_Matematika = ifelse(Nilai_Matematika<60, "TL", "L"))
```

Praktek Ke-17: Menggunakan Fungsi *mutate()* dan *ifelse()*

```
Console -/ ↵
> Data1
  Nama Jenis_Kelamin Nilai_Matematika Nilai_Kimia Nilai_Biologi
1  Ugi Laki-Laki 75 50 70
2  Egi Laki-Laki 60 70 50
3  Andi Laki-Laki 45 75 80
4  Udin Laki-Laki 40 50 80
5  Boy Laki-Laki 50 50 75
6  Sinta Perempuan 75 85 80
7  Rini Perempuan 70 60 40
8  Rani Perempuan 80 90 80
9  Fitri Perempuan 90 50 75
10 Dini Perempuan 80 70 90
> ekstrak <- select(
+   mutate(Data1, Rata_Rata = (Nilai_Matematika+Nilai_Kimia+Nilai_Biologi)/3), Nama, Rata_Rata)
> ekstrak
  Nama Rata_Rata
1  Ugi 65.00000
2  Egi 60.00000
3  Andi 66.66667
4  Udin 56.66667
5  Boy 58.33333
6  Sinta 80.00000
7  Rini 56.66667
8  Rani 83.33333
9  Fitri 71.66667
10 Dini 80.00000
>
> Nilai <- mutate(ekstrak, Nilai =
+   ifelse(Rata_Rata < 60, "D",
+   ifelse(Rata_Rata < 70, "C",
+   ifelse(Rata_Rata < 80, "B", "A"))
+   )
+ )
> Nilai
  Nama Rata_Rata Nilai
1  Ugi 65.00000 C
2  Egi 60.00000 C
3  Andi 66.66667 C
4  Udin 56.66667 D
5  Boy 58.33333 D
6  Sinta 80.00000 A
7  Rini 56.66667 D
8  Rani 83.33333 A
9  Fitri 71.66667 B
10 Dini 80.00000 A
>
> Nilai <- mutate(ekstrak, Nilai =
+   ifelse(Rata_Rata < 60, "D",
+   ifelse(Rata_Rata < 70, "C",
+   ifelse(Rata_Rata < 80, "B", "A"))
+   )
+ )
> Nilai
  Nama Rata_Rata Nilai
1  Ugi 65.00000 C
2  Egi 60.00000 C
3  Andi 66.66667 C
4  Udin 56.66667 D
5  Boy 58.33333 D
6  Sinta 80.00000 A
7  Rini 56.66667 D
8  Rani 83.33333 A
9  Fitri 71.66667 B
10 Dini 80.00000 A
>
> |
```

```

Console ~/
> ekstrak <- Data1 %>%
+   mutate(Rata_Rata = (Nilai_Matematika+Nilai_Kimia+Nilai_Biologi)/3) %>%
+   select>Nama, Rata_Rata)
> ekstrak
  Nama Rata_Rata
1   Ugi 65.00000
2   Egi 60.00000
3  Andi 66.66667
4  Udin 56.66667
5   Boy 58.33333
6  Sinta 80.00000
7   Rini 56.66667
8   Rani 83.33333
9  Fitri 71.66667
10  Dini 80.00000
>
> nilai <- ekstrak %>%
+   mutate(Nilai = ifelse(Rata_Rata < 60, "D",
+                         ifelse(Rata_Rata < 70, "C",
+                         ifelse(Rata_Rata < 80, "B", "A"))))
> nilai
  Nama Rata_Rata Nilai
1   Ugi 65.00000    C
2   Egi 60.00000    C
3  Andi 66.66667    C
4  Udin 56.66667    D
5   Boy 58.33333    D
6  Sinta 80.00000    A
7   Rini 56.66667    D
8   Rani 83.33333    A
9  Fitri 71.66667    B
10  Dini 80.00000    A
>

```

Gambar 12.17

"Tugas:

Tahap 1

- ⇒ Gunakan fungsi `mutate()` pada *data frame* `Data1` untuk membentuk variabel baru bernama `rata-rata`.
- ⇒ Nilai rata-rata dihitung dengan rumus:

$$rata - rata = \frac{nilai\ matematika + nilai\ kimia + nilai\ biologi}{3}$$

- ⇒ Kemudian gunakan fungsi `select()` untuk memilih hanya variabel nama dan rata-rata, dan kemudian disimpan dalam *data frame* `ekstrak`.

Tahap2

- ⇒ Gunakan fungsi `mutate()` pada *data frame* `ekstrak` untuk membentuk variabel baru bernama `nilai`.
- ⇒ Gunakan fungsi `ifelse()` untuk membuat kriteria dari variabel `nilai` sebagai berikut.
 - Jika nilai rata-rata < 60, maka beri nilai D.
 - Jika nilai rata-rata < 70, maka beri nilai C.
 - Jika nilai rata-rata < 80, maka beri nilai B.
 - Jika nilai rata-rata ≥ 80, maka beri nilai A.

⇒ Kemudian gunakan fungsi `select()` untuk memilih hanya variabel nama, rata-rata dan nilai, yang disimpan dalam *data frame* nilai.

Perintah R

```
ekstrak <- select(
  mutate(Data1, Rata_Rata =
    (Nilai_Matematika+Nilai_Kimia+Nilai_Biologi)/3), Nama, Rata_Rata)
ekstrak

Nilai <- mutate(ekstrak, Nilai =
  ifelse(Rata_Rata < 60, "D",
        ifelse(Rata_Rata < 70, "C",
              ifelse(Rata_Rata < 80, "B", "A")))
)
Nilai
```

Alternatif Perintah R

```
ekstrak <- Data1 %>%
  mutate(Rata_Rata = (Nilai_Matematika+Nilai_Kimia+Nilai_Biologi)/3) %>%
  select>Nama, Rata_Rata)
ekstrak

nilai <- ekstrak %>%
  mutate(Nilai = ifelse(Rata_Rata < 60, "D",
                      ifelse(Rata_Rata < 70, "C",
                            ifelse(Rata_Rata < 80, "B", "A"))))
nilai
```

Praktek Ke-18: Menggunakan Fungsi *mutate()* dan *ifelse()*

```
Console ~/ |
> Data1
  Nama Jenis_Kelamin Nilai_Matematika Nilai_Kimia Nilai_Biologi
1  Ugi      Laki-Laki           75           50           70
2  Egi      Laki-Laki           60           70           50
3  Andi     Laki-Laki           45           75           80
4  Udin     Laki-Laki           40           50           80
5  Boy      Laki-Laki           50           50           75
6  Sinta    Perempuan           75           85           80
7  Rini     Perempuan           70           60           40
8  Rani     Perempuan           80           90           80
9  Fitri    Perempuan           90           50           75
10 Dini     Perempuan           80           70           90
> Data1 %>%
+   mutate(
+     kelulusan_Matematika = ifelse(Nilai_Matematika < 60, "TL", "L"),
+     kelulusan_Kimia = ifelse(Nilai_Kimia < 60, "TL", "L"),
+     kelulusan_Biologi = ifelse(Nilai_Biologi < 60, "TL", "L"))
  Nama Jenis_Kelamin Nilai_Matematika Nilai_Kimia Nilai_Biologi kelulusan_Matematika kelulusan_Kimia kelulusan_Biologi
1  Ugi      Laki-Laki           75           50           70              L              TL              L
2  Egi      Laki-Laki           60           70           50              L              L              TL
3  Andi     Laki-Laki           45           75           80              TL              L              L
4  Udin     Laki-Laki           40           50           80              TL              TL              L
5  Boy      Laki-Laki           50           50           75              TL              TL              L
6  Sinta    Perempuan           75           85           80              L              L              L
7  Rini     Perempuan           70           60           40              L              L              TL
8  Rani     Perempuan           80           90           80              L              L              L
9  Fitri    Perempuan           90           50           75              L              TL              L
10 Dini     Perempuan           80           70           90              L              L              L
> |
```

Gambar 12.18

"Tugas:

- ⇒ Gunakan fungsi *mutate()* untuk membentuk variabel kelulusan matematika.
- ⇒ Gunakan fungsi *ifelse()* untuk membuat syarat dari variabel kelulusan matematika. Syarat dari variabel kelulusan matematika adalah sebagai berikut.
 - Jika nilai matematika < 60, maka tidak lulus "TL".
 - Jika nilai matematika > 60, maka lulus "L".
- ⇒ Langkah yang sama untuk membuat variabel kelulusan kimia dan biologi."

Perintah R

```
Data1 %>%
  mutate(
    kelulusan_Matematika = ifelse(Nilai_Matematika < 60, "TL", "L"),
    kelulusan_Kimia = ifelse(Nilai_Kimia < 60, "TL", "L"),
    kelulusan_Biologi = ifelse(Nilai_Biologi < 60, "TL", "L"))
```

Praktek Ke-19: Menggunakan Fungsi *mutate()*, *ifelse()*, dan *select()*

```
Console ~/ |
> Data1
  Nama Jenis_Kelamin Nilai_Matematika Nilai_Kimia Nilai_Biologi
1 Ugi Laki-Laki 75 50 70
2 Egi Laki-Laki 60 70 50
3 Andi Laki-Laki 45 75 80
4 Udin Laki-Laki 40 50 80
5 Boy Laki-Laki 50 50 75
6 Sinta Perempuan 75 85 80
7 Rini Perempuan 70 60 40
8 Rani Perempuan 80 90 80
9 Fitri Perempuan 90 50 75
10 Dini Perempuan 80 70 90
> |

Console ~/ |
> simpan <- Data1 %>%
+ mutate(
+ kelulusan_Matematika = ifelse(Nilai_Matematika < 60, 0, 1),
+ kelulusan_Kimia = ifelse(Nilai_Kimia < 60, 0, 1),
+ kelulusan_Biologi = ifelse(Nilai_Biologi < 60, 0, 1))
>
> simpan
  Nama Jenis_Kelamin Nilai_Matematika Nilai_Kimia Nilai_Biologi kelulusan_Matematika kelulusan_Kimia kelulusan_Biologi
1 Ugi Laki-Laki 75 50 70 1 0 1
2 Egi Laki-Laki 60 70 50 1 1 0
3 Andi Laki-Laki 45 75 80 0 1 1
4 Udin Laki-Laki 40 50 80 0 0 1
5 Boy Laki-Laki 50 50 75 0 0 1
6 Sinta Perempuan 75 85 80 1 1 1
7 Rini Perempuan 70 60 40 1 1 0
8 Rani Perempuan 80 90 80 1 1 1
9 Fitri Perempuan 90 50 75 1 0 1
10 Dini Perempuan 80 70 90 1 1 1
>
> simpan %>%
+ mutate(Jumlah_Pelajaran_yang_Lulus = kelulusan_Matematika + kelulusan_Kimia + kelulusan_Biologi ) %>%
+ select>Nama, Jumlah_Pelajaran_yang_Lulus)
  Nama Jumlah_Pelajaran_yang_Lulus
1 Ugi 2
2 Egi 2
3 Andi 2
4 Udin 1
5 Boy 1
6 Sinta 3
7 Rini 2
8 Rani 3
9 Fitri 2
10 Dini 3
>
```

Gambar 12.19

"Tugas:

Tahap 1

- ⇒ Gunakan fungsi *mutate()* pada data frame *Data1* untuk membentuk variabel kelulusan matematika.
- ⇒ Gunakan fungsi *ifelse()* untuk membuat syarat dari variabel kelulusan matematika. Syarat dari variabel kelulusan matematika adalah sebagai berikut.
 - Jika nilai matematika < 60, maka tidak lulus, diberi angka 0.
 - Jika nilai matematika > 60, maka lulus, diberi angka 1.
- ⇒ Langkah yang sama untuk membuat variabel kelulusan kimia dan biologi.
- ⇒ Selanjutnya simpan ke dalam data frame *simpan*.

Tahap 2

- ⇒ Gunakan fungsi `mutate()` pada *data frame* `simpan` untuk membentuk variabel jumlah pelajaran yang lulus.
- ⇒ Jumlah pelajaran yang lulus dari tiap-tiap responden dihitung dengan menjumlahkan

kelulusan matematika + kelulusan kimia + kelulusan biologi

- ⇒ Gunakan fungsi `select()` untuk hanya menampilkan informasi dari variabel nama dan jumlah pelajaran yang lulus.

Perintah R

```
simpan <- Data1 %>%
```

```
  mutate(  
    kelulusan_Matematika = ifelse(Nilai_Matematika < 60, 0, 1),  
    kelulusan_Kimia = ifelse(Nilai_Kimia < 60, 0, 1),  
    kelulusan_Biologi = ifelse(Nilai_Biologi < 60, 0, 1))
```

```
simpan
```

```
simpan %>%
```

```
mutate(Jumlah_Pelajaran_yang_Lulus = kelulusan_Matematika + kelulusan_Kimia  
+ kelulusan_Biologi ) %>%  
select>Nama, Jumlah_Pelajaran_yang_Lulus)
```

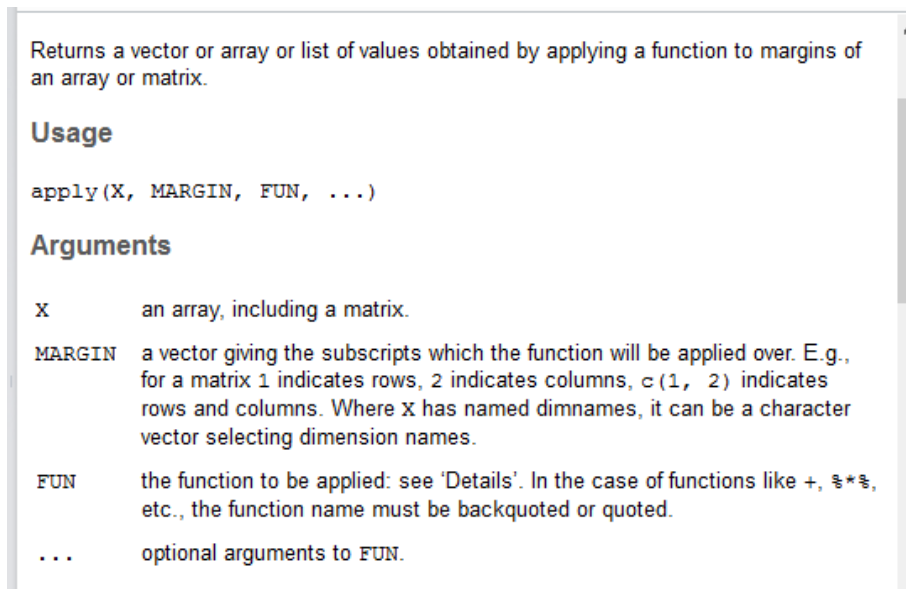

Praktek Ke-20: Menggunakan Fungsi *mutate()*, *select()*, dan *apply()*

```
Console -/ ↻
> Data1
  Nama Jenis_Kelamin Nilai_Matematika Nilai_Kimia Nilai_Biologi
1 Ugi Laki-Laki 75 50 70
2 Egi Laki-Laki 60 70 50
3 Andi Laki-Laki 45 75 80
4 Udin Laki-Laki 40 50 80
5 Boy Laki-Laki 50 50 75
6 Sinta Perempuan 75 85 80
7 Rini Perempuan 70 60 40
8 Rani Perempuan 80 90 80
9 Fitri Perempuan 90 50 75
10 Dini Perempuan 80 70 90
>
> pelajaran <- select(Data1, Nilai_Matematika, Nilai_Kimia, Nilai_Biologi)
> pelajaran
  Nilai_Matematika Nilai_Kimia Nilai_Biologi
1 75 50 70
2 60 70 50
3 45 75 80
4 40 50 80
5 50 50 75
6 75 85 80
7 70 60 40
8 80 90 80
9 90 50 75
10 80 70 90
>
<
> Data1 %>%
+ mutate(
+   Minimum = apply(pelajaran, 1, FUN=min),
+   Maksimum = apply(pelajaran, 1, FUN=max),
+   Rata_Rata = apply(pelajaran, 1, FUN=mean),
+   Median = apply(pelajaran, 1, FUN=median),
+   standardeviasi = apply(pelajaran, 1, FUN=sd))
1 Nama Jenis_Kelamin Nilai_Matematika Nilai_Kimia Nilai_Biologi Minimum Maksimum Rata_Rata Median Standardeviasi
2 Ugi Laki-Laki 75 50 70 50 75 65.00000 70 13.228757
3 Egi Laki-Laki 60 70 50 50 70 60.00000 60 10.000000
4 Andi Laki-Laki 45 75 80 45 80 66.66667 75 18.929694
5 Udin Laki-Laki 40 50 80 40 80 56.66667 50 20.816660
6 Boy Laki-Laki 50 50 75 50 75 58.33333 50 14.433757
7 Sinta Perempuan 75 85 80 75 85 80.00000 80 5.000000
8 Rini Perempuan 70 60 40 40 70 56.66667 60 15.275252
9 Rani Perempuan 80 90 80 80 90 83.33333 80 5.773503
10 Fitri Perempuan 90 50 75 50 90 71.66667 75 20.207259
11 Dini Perempuan 80 70 90 70 90 80.00000 80 10.000000
>
```

Gambar 12.20

"Tugas:

- ⇒ Gunakan fungsi *select()* pada *data frame* *Data1* untuk memilih variabel nilai matematika, nilai kimia, dan nilai biologi dan simpan ke dalam *data frame* *pelajaran*.
- ⇒ Gunakan *mutate()* pada *data frame* *Data1* untuk membentuk variabel minimum, maksimum, rata-rata, median, dan standar deviasi.
- ⇒ Pada pembentukan variabel minimum, maksimum, rata-rata, median, dan standar deviasi, digunakan fungsi *apply()*.



Gambar 12.21

- ⇒ Pada penggunaan fungsi *apply()*, terdapat argumen *MARGIN*, yang mana nilai 1 menunjukkan baris, dan nilai 2 menunjukkan kolom (Gambar 21).

Perintah R

Data1

```
pelajaran <- select(Data1, Nilai_Matematika, Nilai_Kimia, Nilai_Biologi)
pelajaran
```

```
Data1 %>%
mutate(
  Minimum = apply(pelajaran, 1, FUN=min),
  Maksimum = apply(pelajaran, 1, FUN=max),
  Rata_Rata = apply(pelajaran, 1, FUN=mean),
  Median = apply(pelajaran, 1, FUN=median),
  Standardeviasi = apply(pelajaran, 1, FUN=sd))
```

Praktek Ke-21: Menggunakan Fungsi *mutate()*, *select()*, dan *apply()*

```

Console ~/
> Data1
  > Data1
    Nama Jenis_Kelamin Nilai_Matematika Nilai_Kimia Nilai_Biologi
  1   Ugi      Laki-Laki           75           50           NA
  2   Egi      Laki-Laki           NA           70           50
  3   Andi     Laki-Laki           45           75           NA
  4   Udin     Laki-Laki           40           NA           NA
  5   Boy      Laki-Laki           50           50           75
  6   Sinta    Perempuan           75           85           80
  7   Rini     Perempuan           70           NA           40
  8   Rani     Perempuan           80           90           80
  9   Fitri    Perempuan           90           NA           NA
 10  Dini     Perempuan           80           70           90
  > pelajaran <- select(Data1,Nilai_Matematika, Nilai_Kimia, Nilai_Biologi)
  > pelajaran
    Nilai_Matematika Nilai_Kimia Nilai_Biologi
  1             75           50           NA
  2             NA           70           50
  3             45           75           NA
  4             40           NA           NA
  5             50           50           75
  6             75           85           80
  7             70           NA           40
  8             80           90           80
  9             90           NA           NA
 10            80           70           90
  > Data1 %>%
  +   mutate(
  +     rata_rata = apply(pelajaran, 1, FUN = mean, na.rm=TRUE),
  +     rata_rata2 = apply(pelajaran, 1, FUN = mean))
    Nama Jenis_Kelamin Nilai_Matematika Nilai_Kimia Nilai_Biologi rata_rata rata_rata2
  1   Ugi      Laki-Laki           75           50           NA 62.50000      NA
  2   Egi      Laki-Laki           NA           70           50 60.00000      NA
  3   Andi     Laki-Laki           45           75           NA 60.00000      NA
  4   Udin     Laki-Laki           40           NA           NA 40.00000      NA
  5   Boy      Laki-Laki           50           50           75 58.33333     58.33333
  6   Sinta    Perempuan           75           85           80 80.00000     80.00000
  7   Rini     Perempuan           70           NA           40 55.00000      NA
  8   Rani     Perempuan           80           90           80 83.33333     83.33333
  9   Fitri    Perempuan           90           NA           NA 90.00000      NA
 10  Dini     Perempuan           80           70           90 80.00000     80.00000
  > pelajaran <- select(Data1,Nilai_Matematika, Nilai_Kimia, Nilai_Biologi)
  > pelajaran
    Nilai_Matematika Nilai_Kimia Nilai_Biologi
  1             75           50           NA
  2             NA           70           50
  3             45           75           NA
  4             40           NA           NA
  5             50           50           75
  6             75           85           80
  7             70           NA           40
  8             80           90           80
  9             90           NA           NA
 10            80           70           90
  > Data1 %>%
  +   mutate(
  +     rata_rata = apply(pelajaran, 1, FUN = mean, na.rm=TRUE),
  +     rata_rata2 = apply(pelajaran, 1, FUN = mean))
    Nama Jenis_Kelamin Nilai_Matematika Nilai_Kimia Nilai_Biologi rata_rata rata_rata2
  1   Ugi      Laki-Laki           75           50           NA 62.50000      NA
  2   Egi      Laki-Laki           NA           70           50 60.00000      NA
  3   Andi     Laki-Laki           45           75           NA 60.00000      NA
  4   Udin     Laki-Laki           40           NA           NA 40.00000      NA
  5   Boy      Laki-Laki           50           50           75 58.33333     58.33333
  6   Sinta    Perempuan           75           85           80 80.00000     80.00000
  7   Rini     Perempuan           70           NA           40 55.00000      NA
  8   Rani     Perempuan           80           90           80 83.33333     83.33333
  9   Fitri    Perempuan           90           NA           NA 90.00000      NA
 10  Dini     Perempuan           80           70           90 80.00000     80.00000
  >
  
```

Gambar 12.22

"Tugas:

- ⇒ Gunakan fungsi *select()* pada *data frame* *Data1* untuk memilih variabel nilai matematika, nilai kimia, dan nilai biologi dan simpan ke dalam *data frame* *pelajaran*.

- ⇒ Gunakan `mutate()` pada *data frame* `Data1` untuk membentuk variabel rata-rata dan `rata-rata2`.
- ⇒ Pada pembentukan variabel rata-rata dan `rata-rata2` digunakan fungsi `apply()`.
- ⇒ Pada pembentukan variabel rata-rata, melibatkan argumen `na.rm=TRUE`, sementara pada pembentukan variabel `rata-rata2`, tidak melibatkan argumen `na.rm=TRUE`.
- ⇒ Perhatikan perbedaan hasilnya.

Perintah R

`Data1`

```
pelajaran <- select(Data1, Nilai_Matematika, Nilai_Kimia, Nilai_Biologi)
pelajaran
```

```
Data1 %>%
  mutate(
    rata_rata = apply(pelajaran, 1, FUN = mean, na.rm=TRUE),
    rata_rata2 = apply(pelajaran, 1, FUN = mean))
```

Praktek Ke-22: Menggunakan Fungsi *mutate()*, *select()*, dan *apply()*

```

Console -1
> Data1
  Nama Jenis_Kelamin Nilai_Matematika Nilai_Kimia Nilai_Biologi
1   Ugi      Laki-Laki           75           50           NA
2   Egi      Laki-Laki           NA           70           50
3   Andi     Laki-Laki           45           75           NA
4   Udin     Laki-Laki           40           NA           NA
5   Boy      Laki-Laki           50           50           75
6   Sinta    Perempuan           75           85           80
7   Rini     Perempuan           70           NA           40
8   Rani     Perempuan           80           90           80
9   Fitri    Perempuan           90           NA           NA
10  Dini     Perempuan           80           70           90
> pelajaran <- select(Data1, Nilai_Matematika, Nilai_Kimia, Nilai_Biologi)
> pelajaran
  Nilai_Matematika Nilai_Kimia Nilai_Biologi
1             75             50             NA
2             NA             70             50
3             45             75             NA
4             40             NA             NA
5             50             50             75
6             75             85             80
7             70             NA             40
8             80             90             80
9             90             NA             NA
10            80             70             90
> Data1 %>%
+   mutate(
+     jumlah_ujian_yang_diikuti = apply(pelajaran, 1, function(x){ length(v
+     rata_rata = apply(pelajaran, 1, FUN = mean, na.rm=TRUE)
+
> Data1 %>%
+   mutate(
+     jumlah_ujian_yang_diikuti = apply(pelajaran, 1, function(x){ length(which(!is.na(x))) } ),
+     rata_rata = apply(pelajaran, 1, FUN = mean, na.rm=TRUE),
+     standar_deviasi = apply(pelajaran, 1, FUN = sd, na.rm=TRUE),
+     median = apply(pelajaran, 1, FUN = median, na.rm=TRUE),
+     minimum = apply(pelajaran, 1, FUN = min, na.rm=TRUE),
+     maksimum = apply(pelajaran, 1, FUN = mean, na.rm=TRUE),
+     jumlah = apply(pelajaran, 1, FUN = sum, na.rm=TRUE)
+   )
+ )
  Nama Jenis_Kelamin Nilai_Matematika Nilai_Kimia Nilai_Biologi jumlah_ujian_yang_diikuti rata_rata standar_deviasi median minimum maksimum jumlah
1   Ugi      Laki-Laki           75           50           NA                2 62.50000    17.677670  62.5    50 62.50000    125
2   Egi      Laki-Laki           NA           70           50                2 60.00000    14.142136  60.0    50 60.00000    120
3   Andi     Laki-Laki           45           75           NA                2 60.00000    21.213203  60.0    45 60.00000    120
4   Udin     Laki-Laki           40           NA           NA                1 40.00000         NA  40.0    40 40.00000     40
5   Boy      Laki-Laki           50           50           75                3 58.33333    14.433757  50.0    50 58.33333    175
6   Sinta    Perempuan           75           85           80                3 80.00000     5.000000  80.0    75 80.00000    240
7   Rini     Perempuan           70           NA           40                2 55.00000    21.213203  55.0    40 55.00000    110
8   Rani     Perempuan           80           90           80                3 83.33333     5.773503  80.0    80 83.33333    250
9   Fitri    Perempuan           90           NA           NA                1 90.00000         NA  90.0    90 90.00000     90
10  Dini     Perempuan           80           70           90                3 80.00000    10.000000  80.0    70 80.00000    240
> |

```

Gambar 12.23

"Tugas:

- ⇒ Gunakan fungsi *select()* pada *data frame* *Data1* untuk memilih variabel nilai matematika, nilai kimia, dan nilai biologi dan simpan ke dalam *data frame* *pelajaran*.
- ⇒ Gunakan *mutate()* pada *data frame* *Data1* untuk membentuk variabel jumlah ujian yang diikuti, rata-rata, standar deviasi, median, minimum, maksimum, dan jumlah.
- ⇒ Pada pembentukan variabel jumlah ujian yang diikuti, rata-rata, standar deviasi, median, minimum, maksimum, dan jumlah, digunakan fungsi *apply()*.
- ⇒ Pada penggunaan fungsi *apply()*, argumen *na.rm=TRUE* bertujuan untuk mengabaikan data yang kosong.

Perintah R

Data1

```
pelajaran <- select(Data1, Nilai_Matematika, Nilai_Kimia, Nilai_Biologi)
pelajaran
```

```
Data1 %>%
  mutate(
    jumlah_ujian_yang_diikuti = apply(pelajaran, 1, function(x){
length(which(!is.na(x))) } ),
    rata_rata = apply(pelajaran, 1, FUN = mean, na.rm=TRUE),
    standar_deviasi = apply(pelajaran, 1, FUN = sd, na.rm=TRUE),
    median = apply(pelajaran, 1, FUN = median, na.rm=TRUE),
    minimum = apply(pelajaran, 1, FUN = min, na.rm=TRUE),
    maksimum = apply(pelajaran, 1, FUN = max, na.rm=TRUE),
    jumlah = apply(pelajaran, 1, FUN = sum, na.rm=TRUE)
  )
```

Praktek Ke-23: Menggunakan Fungsi *summarise()*

```
Console -/ |
> Data1
  Nama Jenis_Kelamin Nilai_Matematika Nilai_Kimia Nilai_Biologi
1 Ugi Laki-Laki 75 50 70
2 Egi Laki-Laki 60 70 50
3 Andi Laki-Laki 45 75 80
4 Udin Laki-Laki 40 50 80
5 Boy Laki-Laki 50 50 75
6 Sinta Perempuan 75 85 80
7 Rini Perempuan 70 60 40
8 Rani Perempuan 80 90 80
9 Fitri Perempuan 90 50 75
10 Dini Perempuan 80 70 90
> Data1 %>%
+ summarise(
+ rata_rata_matematika = mean(Nilai_Matematika),
+ rata_rata_kimia = mean(Nilai_Kimia),
+ rata_rata_biologi = mean(Nilai_Biologi),
+ standar_deviasi_matematika = sd(Nilai_Matematika),
+ standar_deviasi_kimia = sd(Nilai_Kimia),
+ standar_deviasi_biologi = sd(Nilai_Biologi),
+ minimum_matematika = min(Nilai_Matematika),
+ minimum_kimia = min(Nilai_Kimia),
+ minimum_biologi = min(Nilai_Biologi),
+ maksimum_matematika = max(Nilai_Matematika),
+ maksimum_kimia = max(Nilai_Kimia),
+ maksimum_biologi = max(Nilai_Biologi)
+ )
  rata_rata_matematika rata_rata_kimia rata_rata_biologi standar_deviasi_matematika standar_deviasi_kimia standar_deviasi_biologi minimum_matematika
1 66.5 65 72 16.84076 15.27525 15.31158 40
  minimum_kimia minimum_biologi maksimum_matematika maksimum_kimia maksimum_biologi
1 50 40 90 90 90
> |
```

Gambar 12.24

"Tugas:

- ⇒ **Gunakan fungsi *summarise()* untuk menghitung nilai rata-rata, standar deviasi, minimum, dan maksimum, dari variabel nilai matematika, kimia, dan biologi."**

Perintah R

```
Data1 %>%
summarise(
rata_rata_matematika = mean(Nilai_Matematika),
rata_rata_kimia = mean(Nilai_Kimia),
rata_rata_biologi = mean(Nilai_Biologi),
standar_deviasi_matematika = sd(Nilai_Matematika),
standar_deviasi_kimia = sd(Nilai_Kimia),
standar_deviasi_biologi = sd(Nilai_Biologi),
minimum_matematika = min(Nilai_Matematika),
minimum_kimia = min(Nilai_Kimia),
minimum_biologi = min(Nilai_Biologi),
maksimum_matematika = max(Nilai_Matematika),
maksimum_kimia = max(Nilai_Kimia),
maksimum_biologi = max(Nilai_Biologi)
)
```

Praktek Ke-24: Menggunakan Fungsi *summarise()*

```
Console ~/ /
> Data1
  Nama Jenis_Kelamin Nilai_Matematika Nilai_Kimia Nilai_Biologi
1  Ugi      Laki-Laki           75           50           NA
2  Egi      Laki-Laki           NA           70           50
3  Andi     Laki-Laki           45           75           NA
4  Udin     Laki-Laki           40           NA           NA
5  Boy      Laki-Laki           50           50           75
6  Sinta    Perempuan           75           85           80
7  Rini     Perempuan           70           NA           40
8  Rani     Perempuan           80           90           80
9  Fitri    Perempuan           90           NA           NA
10 Dini     Perempuan           80           70           90
> Data1 %>%
+ summarise(
+ rata_rata_matematika = mean(Nilai_Matematika),
+ rata_rata_matematika2 = mean(Nilai_Matematika, na.rm=TRUE))
  rata_rata_matematika rata_rata_matematika2
1                   NA                   67.22222
>
```

Gambar 12.25

"Tugas:

- ⇒ Gunakan fungsi *summarise()* untuk menghitung nilai rata-rata matematika.
- ⇒ Karena data pada variabel nilai matematika memuat NA, maka beri tambahan argumen *na.rm=TRUE*."

Perintah R

```
Data1 %>%
summarise(
rata_rata_matematika = mean(Nilai_Matematika),
rata_rata_matematika2 = mean(Nilai_Matematika, na.rm=TRUE))
```


Praktek Ke-25: Menggunakan Fungsi *summarise()*

```
Console ~1
> Data1
  Nama Jenis_Kelamin Nilai_Matematika Nilai_Kimia Nilai_Biologi
1  Ugi      Laki-Laki           75           50           NA
2  Egi      Laki-Laki           NA           70           50
3  Andi     Laki-Laki           45           75           NA
4  Udin     Laki-Laki           40           NA           NA
5  Boy      Laki-Laki           50           50           75
6  Sinta    Perempuan           75           85           80
7  Rini     Perempuan           70           NA           40
8  Rani     Perempuan           80           90           80
9  Fitri    Perempuan           90           NA           NA
10 Dini     Perempuan           80           70           90
> Data1 %>%
+   summarise(
+     rata_rata_matematika = mean(Nilai_Matematika, na.rm=TRUE),
+     median_matematika = median(Nilai_Matematika, na.rm=TRUE),
+     standar_deviasi_matematika = sd(Nilai_Matematika, na.rm=TRUE),
+     minimum_matematika = min(Nilai_Matematika, na.rm=TRUE),
+     maksimum_matematika = max(Nilai_Matematika, na.rm=TRUE),
+     jumlah_seluruh_nilai = sum(Nilai_Matematika, na.rm=TRUE),
+     banyaknya_nilai = length(na.omit(Nilai_Matematika)),
+     banyaknya_nilai_yang_berbeda = n_distinct(na.omit(Nilai_Matematika)))
  rata_rata_matematika median_matematika standar_deviasi_matematika minimum_matem
atika maksimum_matematika
1           67.22222           75           17.6973
  jumlah_seluruh_nilai banyaknya_nilai banyaknya_nilai_yang_berbeda
1           605           9           7
>
```

Gambar 12.26




"Tugas:

- ⇒ Gunakan fungsi *summarise()* untuk menghitung nilai rata-rata, median, modus, standar deviasi, minimum, maksimum, jumlah seluruh nilai, banyaknya nilai, dan banyaknya nilai yang berbeda, dari variabel nilai matematika.

Perintah R

```
Data1 %>%
  summarise(
    rata_rata_matematika = mean(Nilai_Matematika, na.rm=TRUE),
    median_matematika = median(Nilai_Matematika, na.rm=TRUE),
    standar_deviasi_matematika = sd(Nilai_Matematika, na.rm=TRUE),
    minimum_matematika = min(Nilai_Matematika, na.rm=TRUE),
    maksimum_matematika = max(Nilai_Matematika, na.rm=TRUE),
    jumlah_seluruh_nilai = sum(Nilai_Matematika, na.rm=TRUE),
    banyaknya_nilai = length(na.omit(Nilai_Matematika)),
    banyaknya_nilai_yang_berbeda =
n_distinct(na.omit(Nilai_Matematika)))
```

Praktek Ke-26: Menghitung Modus dari Suatu Data Menggunakan Fungsi *summarise()* dan *mfv()*

```
Console ~/     
> Data1  
> Data1  
  Nama Jenis_Kelamin Nilai_Matematika Nilai_Kimia Nilai_Biologi  
1 Ugi Laki-Laki 75 50 NA  
2 Egi Laki-Laki NA 70 50  
3 Andi Laki-Laki 45 75 NA  
4 Udin Laki-Laki 40 NA NA  
5 Boy Laki-Laki 50 50 75  
6 Sinta Perempuan 75 85 80  
7 Rini Perempuan 70 NA 40  
8 Rani Perempuan 80 90 80  
9 Fitri Perempuan 90 NA NA  
10 Dini Perempuan 80 70 90  
> library(modeest)  
> mfv(na.omit(Data1$Nilai_Matematika)) #Modus dari data nilai matematika  
[1] 75 80  
> mfv(Data1$Nilai_Matematika, na.rm=TRUE) #Modus dari data nilai matematika  
[1] 75 80  
> mfv(na.omit(Data1$Nilai_Kimia)) #Modus dari data nilai kimia  
[1] 50 70  
> mfv(Data1$Nilai_Kimia, na.rm=TRUE) #Modus dari data nilai kimia  
[1] 50 70  
> mfv(na.omit(Data1$Nilai_Biologi)) #Modus dari data nilai biologi  
[1] 80  
> mfv(Data1$Nilai_Biologi, na.rm=TRUE) #Modus dari data nilai biologi  
[1] 80  
>  
> Data1 %>%  
+ summarise(  
+   modus_matematika = mlv(Nilai_Matematika, na.rm=TRUE, method='mfv')[['M']]  
+ ),  
+   modus_kimia = mlv(Nilai_Kimia, na.rm=TRUE, method='mfv')[['M']],  
+   modus_biologi = mlv(Nilai_Biologi, na.rm=TRUE, method='mfv')[['M']]  
+ )  
modus_matematika modus_kimia modus_biologi  
1 77.5 60 80  
>
```

Gambar 12.27

"Tugas:

- ⇒ Gunakan fungsi *summarise()* untuk menghitung nilai rata-rata, median, modus, standar deviasi, minimum, maksimum, jumlah seluruh nilai, banyaknya nilai, dan banyaknya nilai yang berbeda, dari variabel nilai matematika.

Perintah R

```
Data1  
library(modeest)  
mfv(na.omit(Data1$Nilai_Matematika)) #Modus dari data nilai matematika  
mfv(Data1$Nilai_Matematika, na.rm=TRUE) #Modus dari data nilai matematika  
mfv(na.omit(Data1$Nilai_Kimia)) #Modus dari data nilai kimia  
mfv(Data1$Nilai_Kimia, na.rm=TRUE) #Modus dari data nilai kimia  
mfv(na.omit(Data1$Nilai_Biologi)) #Modus dari data nilai biologi  
mfv(Data1$Nilai_Biologi, na.rm=TRUE) #Modus dari data nilai biologi
```

```
Data1 %>%  
  summarise(  
    modus_matematika = mlv(Nilai_Matematika, na.rm=TRUE,  
method='mfv')[['M']],  
    modus_kimia = mlv(Nilai_Kimia, na.rm=TRUE, method='mfv')[['M']],  
    modus_biologi = mlv(Nilai_Biologi, na.rm=TRUE,  
method='mfv')[['M']])
```

Praktek Ke-27: Menggunakan Fungsi *distinct()*, *n_distinct()*, dan *summarise()*

```
Console ~/ /
> Data1
  Nama Jenis_Kelamin Pendidikan Jumlah_Bersaudara
1   Ugi      Laki-Laki      S1             2
2   Egi      Laki-Laki      S2             3
3   Andi     Laki-Laki      S3             1
4   Udin     Laki-Laki      S1             2
5   Boy      Laki-Laki      S2             5
6   Sintra   Perempuan      S2             2
7   Rini     Perempuan      S2             4
8   Rani     Perempuan      S1             2
9   Fitri    Perempuan      S3             3
10  Dini     Perempuan      S2             5
> Data1 %>%
+ distinct(Jenis_Kelamin)
  Jenis_Kelamin
1     Laki-Laki
2     Perempuan
> Data1 %>%
+ distinct(Pendidikan)
  Pendidikan
1     S1
2     S2
3     S3
> Data1 %>%
+ distinct(Jumlah_Bersaudara)
  Jumlah_Bersaudara
1     2
2     3
3     1
4     5
5     4
> Data1 %>%
+ summarise(n_distinct(Jumlah_Bersaudara))
  n_distinct(Jumlah_Bersaudara)
1                               5
> Data1 %>%
+ summarise(jumlah_kategori_pendidikan = n_distinct(Pendidikan))
  jumlah_kategori_pendidikan
1                               3
```

Gambar 12.28

"Tugas:

- ⇒ Gunakan fungsi *distinct()* untuk menentukan data-data yang berbeda (*unique values*) dari variabel jenis kelamin, pendidikan, dan jumlah bersaudara.
- ⇒ Gunakan fungsi *n_distinct()* untuk menentukan jumlah data yang berbeda.

Perintah R

```
Data1
```

```
Data1 %>%
distinct(Jenis_Kelamin)
```

```
Data1 %>%
distinct(Pendidikan)
```

```
Data1 %>%
distinct(Jumlah_Bersaudara)
```

```
Data1 %>%
summarise(n_distinct(Jumlah_Bersaudara))
```

```
Data1 %>%
summarise(jumlah_kategori_pendidikan = n_distinct(Pendidikan))
```

Praktek Ke-28: Menggunakan Fungsi *transmute()* dan *mutate()*

```
Console ~/ |
> Data1
  Nama Jenis_Kelamin Nilai_Matematika Nilai_Kimia Nilai_Biologi
1  Ugi      Laki-Laki           75           50           70
2  Egi      Laki-Laki           60           70           50
3  Andi     Laki-Laki           45           75           80
4  Udin     Laki-Laki           40           50           80
5  Boy      Laki-Laki           50           50           75
6  Sinta    Perempuan           75           85           80
7  Rini     Perempuan           70           60           40
8  Rani     Perempuan           80           90           80
9  Fitri    Perempuan           90           50           75
10 Dini     Perempuan           80           70           90
> Data1 %>%
+ mutate(Rata_Rata = (Nilai_Matematika+Nilai_Kimia+Nilai_Biologi)/3)
  Nama Jenis_Kelamin Nilai_Matematika Nilai_Kimia Nilai_Biologi Rata_Rata
1  Ugi      Laki-Laki           75           50           70 65.00000
2  Egi      Laki-Laki           60           70           50 60.00000
3  Andi     Laki-Laki           45           75           80 66.66667
4  Udin     Laki-Laki           40           50           80 56.66667
5  Boy      Laki-Laki           50           50           75 58.33333
6  Sinta    Perempuan           75           85           80 80.00000
7  Rini     Perempuan           70           60           40 56.66667
8  Rani     Perempuan           80           90           80 83.33333
9  Fitri    Perempuan           90           50           75 71.66667
10 Dini     Perempuan           80           70           90 80.00000
> Data1 %>%
+ transmute(Rata_Rata = (Nilai_Matematika+Nilai_Kimia+Nilai_Biologi)/3)
  Rata_Rata
1 65.00000
2 60.00000
3 66.66667
4 56.66667
5 58.33333
6 80.00000
7 56.66667
8 83.33333
9 71.66667
10 80.00000
> |
```

Gambar 12.29

Perintah R

```
Data1
```

```
Data1 %>%
```

```
mutate(Rata_Rata = (Nilai_Matematika+Nilai_Kimia+Nilai_Biologi)/3)
```

```
Data1 %>%
```

```
transmute(Rata_Rata = (Nilai_Matematika+Nilai_Kimia+Nilai_Biologi)/3)
```

Praktek Ke-29: Menggunakan Fungsi `group_by()` dan `n()`

```
Console ~/ | ↻
> Data1
  Nama Jenis_kelamin Pendidikan Jumlah_Bersaudara
1   Ugi          <NA>          <NA>              2
2   Egi     Laki-Laki         S2              3
3   Andi     Laki-Laki         S3              1
4   Udin     Laki-Laki         S1              2
5   Boy      Laki-Laki         <NA>              5
6   Sinta          <NA>          <NA>              2
7   Rini          <NA>         S2              4
8   Rani     Perempuan         S1              2
9   Fitri     Perempuan         S3              3
10  Dini     Perempuan         S2              5
> grup <- group_by(Data1, Jenis_kelamin)
> grup %>%
+ summarise(
+ distribusi_frekuensi_jenis_kelamin = n() )
# A tibble: 3 × 2
  Jenis_kelamin distribusi_frekuensi_jenis_kelamin
  <fctr>          <int>
1   Laki-Laki          4
2   Perempuan          3
3      NA              3
> |
```

Gambar 12.30

Berdasarkan Gambar 12.30, diketahui jumlah laki-laki sebanyak 4, jumlah perempuan sebanyak 3, dan NA sebanyak 3.

Perintah R

```
grup <- group_by(Data1, Jenis_kelamin)
grup %>%
summarise(
distribusi_frekuensi_jenis_kelamin = n() )
```

Praktek Ke-30: Menggunakan Fungsi `group_by()` dan `n()`

```
Console ~/ |
> Data1
  Nama Jenis_Kelamin Pendidikan Jumlah_Bersaudara
1   Ugi           <NA>         <NA>              2
2   Egi           Laki-Laki     S2              3
3   Andi          Laki-Laki     S3              1
4   Udin          Laki-Laki     S1              2
5   Boy           Laki-Laki     <NA>           5
6   Sinta         <NA>         <NA>           2
7   Rini          <NA>         S2              4
8   Rani          Perempuan    S1              2
9   Fitri         Perempuan    S3              3
10  Dini          Perempuan    S2              5
> grup <- group_by(Data1, Jenis_Kelamin) %>%
+ filter(Jenis_Kelamin != "NA" )
> grup
Source: local data frame [7 x 4]
Groups: Jenis_Kelamin [2]

  Nama Jenis_Kelamin Pendidikan Jumlah_Bersaudara
<fctr> <fctr> <fctr> <dbl>
1   Egi           Laki-Laki     S2              3
2   Andi          Laki-Laki     S3              1
3   udin          Laki-Laki     S1              2
4   Boy           Laki-Laki     NA              5
5   Rani          Perempuan    S1              2
6   Fitri         Perempuan    S3              3
7   Dini          Perempuan    S2              5
> grup %>%
+ summarise(
+ distribusi_frekuensi_jenis_kelamin = n())
# A tibble: 2 x 2
  Jenis_Kelamin distribusi_frekuensi_jenis_kelamin
  <fctr> <int>
1   Laki-Laki              4
2   Perempuan             3
> |
```

Gambar 12.31

Berdasarkan Gambar 12.31, diketahui jumlah laki-laki sebanyak 4, dan jumlah perempuan sebanyak 3.

Perintah R

```
Data1
```

```
grup <- group_by(Data1, Jenis_Kelamin) %>%
filter(Jenis_Kelamin != "NA" )
grup
```

```
grup %>%
summarise(
distribusi_frekuensi_jenis_kelamin = n())
```

Praktek Ke-31: Menggunakan Fungsi *na.exclude()*

```
Console ~/ |
> Data1
  Nama Jenis_Kelamin Pendidikan Jumlah_Bersaudara
1  Ugi             <NA>         <NA>          2
2  Egi             Laki-Laki         S2             3
3  Andi            Laki-Laki         S3             1
4  Udin            Laki-Laki         S1             2
5  Boy             Laki-Laki         <NA>          5
6  Sinta           <NA>         <NA>          2
7  Rini            <NA>         S2             4
8  Rani            Perempuan         S1             2
9  Fitri           Perempuan         S3             3
10 Dini            Perempuan         S2             5
> na.exclude(Data1)
  Nama Jenis_Kelamin Pendidikan Jumlah_Bersaudara
2  Egi             Laki-Laki         S2             3
3  Andi            Laki-Laki         S3             1
4  Udin            Laki-Laki         S1             2
8  Rani            Perempuan         S1             2
9  Fitri           Perempuan         S3             3
10 Dini            Perempuan         S2             5
> |
```

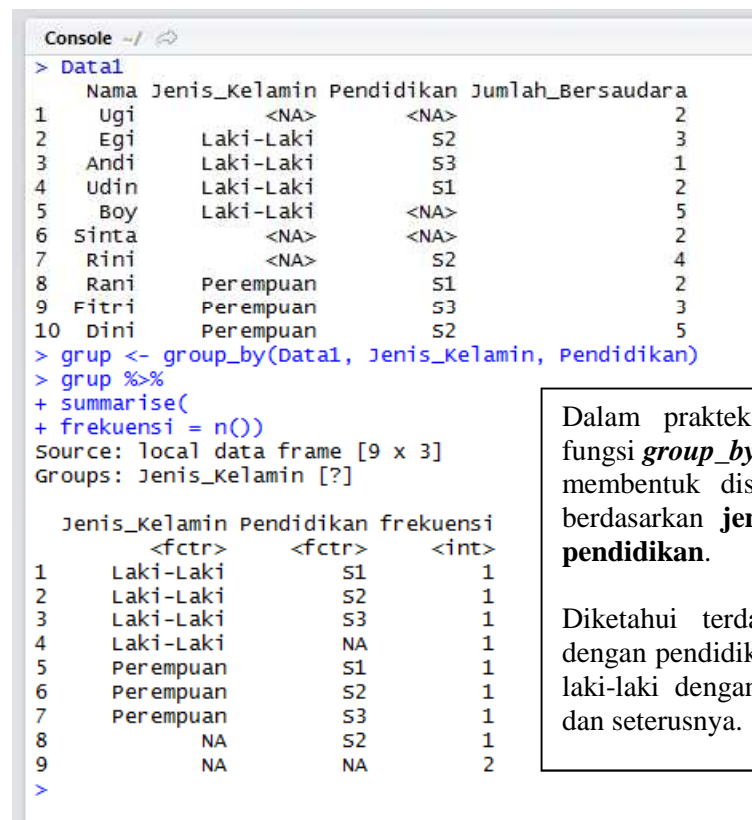
Gambar 12.32

Berdasarkan Gambar 12.32, digunakan fungsi *na.exclude()* untuk mengeliminasi baris yang mengandung NA atau mengeliminasi *cell* yang kosong (tidak terisi / NA / *not available*)

Perintah R

```
Data1
na.exclude(Data1)
```

Praktek Ke-32: Menggunakan Fungsi `group_by()` dan `n()`



```
Console ~/   
> Data1  
  Nama Jenis_Kelamin Pendidikan Jumlah_Bersaudara  
1   Ugi           <NA>         <NA>             2  
2   Egi      Laki-Laki         S2             3  
3   Andi      Laki-Laki         S3             1  
4   Udin      Laki-Laki         S1             2  
5   Boy       Laki-Laki         <NA>           5  
6   Sinta     <NA>             <NA>           2  
7   Rini     <NA>             S2             4  
8   Rani     Perempuan         S1             2  
9   Fitri     Perempuan         S3             3  
10  Dini     Perempuan         S2             5  
> grup <- group_by(Data1, Jenis_Kelamin, Pendidikan)  
> grup %>%  
+ summarise(  
+ frekuensi = n())  
Source: local data frame [9 x 3]  
Groups: Jenis_Kelamin [?]  
  
  Jenis_Kelamin Pendidikan frekuensi  
  <fctr>         <fctr>         <int>  
1   Laki-Laki     S1             1  
2   Laki-Laki     S2             1  
3   Laki-Laki     S3             1  
4   Laki-Laki     NA             1  
5   Perempuan     S1             1  
6   Perempuan     S2             1  
7   Perempuan     S3             1  
8   NA           S2             1  
9   NA           NA             2  
>
```

Dalam praktek ini, digunakan fungsi `group_by()` dan `n()` untuk membentuk distribusi frekuensi berdasarkan **jenis kelamin** dan **pendidikan**.

Diketahui terdapat 1 laki-laki dengan pendidikan S1, terdapat 1 laki-laki dengan pendidikan S2, dan seterusnya.

Gambar 12.33

Perintah R

```
Data1  
grup <- group_by(Data1, Jenis_Kelamin, Pendidikan)  
grup %>%  
summarise(  
frekuensi = n())
```


Praktek Ke-33: Menggunakan Fungsi `group_by()` dan `n()`

```
Console -1 ↻
> Data1
  Nama Jenis_Kelamin Kelas Nilai_Matematika Nilai_Kimia
1 Ugi Laki-Laki I 75 50
2 Egi Laki-Laki II 60 70
3 Andi Laki-Laki III 45 75
4 Udin Laki-Laki I 75 50
5 Boy Laki-Laki II 50 50
6 Sinta Perempuan III 75 85
7 Rini Perempuan I 70 60
8 Rani Perempuan II 80 90
9 Fitri Perempuan III 90 50
10 Dini Perempuan I 80 70
>
> grup <- group_by(Data1, Jenis_Kelamin, Kelas)
> grup %>%
+ summarise(
+ rata_rata_matematika = mean(Nilai_Matematika),
+ rata_rata_kimia = mean(Nilai_Kimia),
+ nilai_berbeda_matematika = n_distinct(Nilai_Matematika),
+ nilai_berbeda_kimia = n_distinct(Nilai_Kimia),
+ minimum_matematika = min(Nilai_Matematika),
+ minimum_kimia = min(Nilai_Kimia),
+ jumlah_siswa = n())
Source: local data frame [6 x 9]
Groups: Jenis_Kelamin [?]

  Jenis_Kelamin Kelas rata_rata_matematika rata_rata_kimia nilai_berbeda_matematika nilai_berbeda_kimia minimum_matematika minimum_kimia jumlah_siswa
  <fctr> <fctr> <dbl> <dbl> <int> <int> <dbl> <dbl> <int>
1 Laki-Laki I 75.0 50.0 1 1 75 50 2
2 Laki-Laki II 55.0 60.0 2 2 50 50 2
3 Laki-Laki III 45.0 75.0 1 1 45 75 1
4 Perempuan I 75.0 65.0 2 2 70 60 2
5 Perempuan II 80.0 90.0 1 1 80 90 1
6 Perempuan III 82.5 67.5 2 2 75 50 2
>
> |
```

Dalam praktek ini, penggunaan fungsi `group_by()` dan `n()` bertujuan untuk membentuk distribusi frekuensi berdasarkan **jenis kelamin** dan **kelas**. Serta ingin diketahui rata-rata nilai berdasarkan **jenis kelamin** dan **kelas**.

→ Diketahui jumlah siswa dengan jenis kelamin laki-laki dari kelas I sebanyak 2. Nilai rata-rata laki-laki dari kelas I, untuk pelajaran matematika 75, untuk pelajaran kimia 50.

Gambar 12.34

Perintah R

Data1

```
grup <- group_by(Data1, Jenis_Kelamin, kelas)
grup %>%
summarise(
rata_rata_matematika = mean(Nilai_Matematika),
rata_rata_kimia = mean(Nilai_Kimia),
nilai_berbeda_matematika = n_distinct(Nilai_Matematika),
nilai_berbeda_kimia = n_distinct(Nilai_Kimia),
minimum_matematika = min(Nilai_Matematika),
minimum_kimia = min(Nilai_Kimia),
jumlah_siswa = n())
```

Praktek Ke-34: Menggunakan Fungsi `join()` dan `data.frame()`

```
> Data1
  Nama Jenis_kelamin
1 Ugi      Laki-Laki
2 Egi      Laki-Laki
3 Andi     Laki-Laki
> Data3
  Nama Nilai_Kimia
1 Ugi      50
2 Egi      70
3 Andi     75
> full_join(Data1,Data3)
Joining, by = "Nama"
  Nama Jenis_kelamin Nilai_Kimia
1 Ugi      Laki-Laki      50
2 Egi      Laki-Laki      70
3 Andi     Laki-Laki      75
> |
```

Terdapat 2 data frame, yakni **Data1** dan **Data2**, yang selanjutnya digabung dengan fungsi `join()`.

Gambar 12.35

```
Console ~| ↻
> Data1
  Nama Jenis_kelamin
1 Ugi      Laki-Laki
2 Egi      Laki-Laki
3 Andi     Laki-Laki
> Data2
  Kelas Nilai_Matematika
1 I      75
2 II     60
3 III    45
> data.frame(Data1,Data2)
  Nama Jenis_kelamin Kelas Nilai_Matematika
1 Ugi      Laki-Laki   I      75
2 Egi      Laki-Laki  II     60
3 Andi     Laki-Laki  III    45
> |
```

Terdapat 2 data frame, yakni **Data1** dan **Data2**, yang selanjutnya digabung dengan fungsi `data.frame()`.

Gambar 12.36

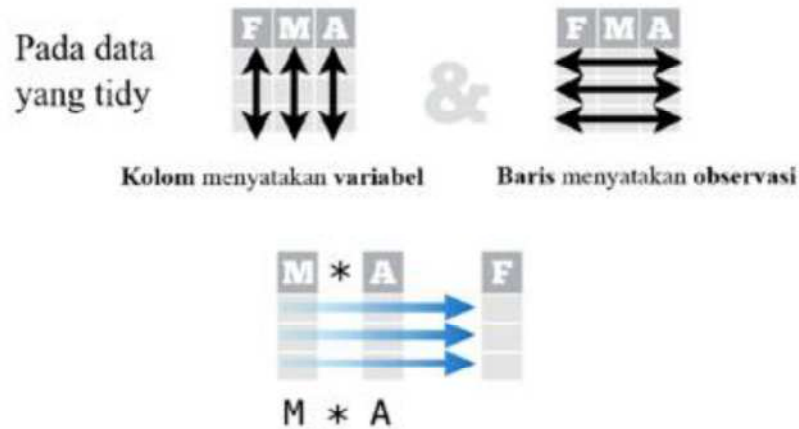
Perintah R

`Data1`

```
grup <- group_by(Data1, Jenis_kelamin, kelas)
grup %>%
summarise(
rata_rata_matematika = mean(Nilai_Matematika),
rata_rata_kimia = mean(Nilai_Kimia),
nilai_berbeda_matematika = n_distinct(Nilai_Matematika),
nilai_berbeda_kimia = n_distinct(Nilai_Kimia),
minimum_matematika = min(Nilai_Matematika),
minimum_kimia = min(Nilai_Kimia),
jumlah_siswa = n())
```

Tidy Data

Data yang rapi (*tidy data*) merupakan pondasi utama dalam memanipulasi data menggunakan *package dplyr* dan *tidyr*.

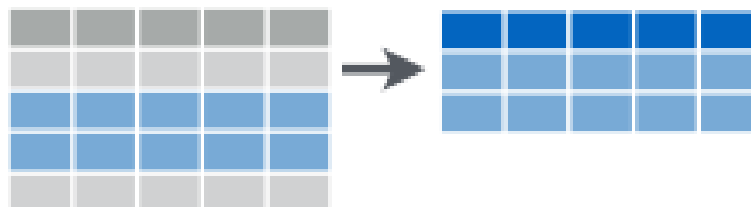


Gambar 12.37

Merubah Tampilan Daset

- `data_frame(a=1:3,b=4:6)`
Mengombinasikan vektors menjadi data frame
- `arrange(mtcars,mpg)`
Mengurutkan baris dari rendah ke tinggi berdasarkan variabel mpg
- `arrange(mtcars, desc(mpg))`
Mengurutkan baris dari tinggi ke rendah
- `rename(tb,y = year)`
Mengganti nama kolom atau variabel year menjadi y

Subset Observasi



Gambar 12.38

- `filter(iris,Sepal.Length > 7)`
Mengekstrak baris data sesuai kriteria yang ditentukan

- `distinct(iris)`
Menghilangkan baris yang nilai-nilainya sama
- `sample_frac(iris, 0.5, replace=TRUE)`
Mengambil random setengah baris data secara random
- `sample_n(iris, 10, replace=TRUE)`
Memilih n baris data secara random
- `slice(iris, 10:15)`
Mengambil baris data sesuai posisi yang diinginkan.
- `top_n(df, 2, x)`
Memilih dan mengurutkan 2 observasi dataset df berdasarkan variabel x

Subset Variabel



Gambar 12.39

- `select(iris, Sepal.Width, Petal.Length, Species)`
Memilih kolom berdasarkan nama atau fungsi bantuan (*helper*):
- `select(iris, contains("."))`
Memilih kolom yang memiliki nama variabelnya memiliki unsur karakter yang diinginkan. Pada kasus ini titik(.)
- `select(iris, ends_with("Length"))`
Memilih kolom yang memiliki nama berakhiran *Length*
- `select(iris, everything())`
Memilih semua kolom
- `select(iris, matches(".t."))`
Memilih kolom yang namanya ada huruf *t*.
- `iris, num_range("x", 1:5)`
Memilih kolom yang bernama x_1, x_2, x_3, x_4, x_5
- `select(iris, one_of(c("Species", "Genus")))`
Memilih kolom dari salah satu *Species* atau *Genus*
- `select(iris, start_with("Sepal"))`
Memilih kolom yang namanya dimulai dengan kata *Sepal*

- `select (iris, Sepal.Length:Petal.Width)`
Memilih semua kolom diantara *Sepal.Length* dan *Petal.Width*
- `select (iris, -Species)`
Memilih semua kolom selain *Species*

Summarise Data



Gambar 13.40

- `summarise (iris, avg=mean (Sepal.Length))`
Meringkas data menjadi menjadi satu baris
 - `summarise_each (iris, funs (mean))`
Menerapkan fungsi *summary* yakni mean ke masing-masing kolom
 - `count (iris, Species, wt=Sepal.Length)`
Mencacah banyaknya kategori species dengan kriteria *Sepal.Length*
- Beberapa fungsi *summary* yang sering digunakan,

first	min
last	max
nth	median
n	mean
n_distinct	var
IQR	sd

Membuat Variabel Baru



Gambar 13.41

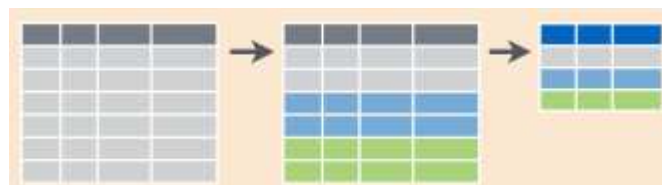
- `mutate(iris, sepal = Sepal.Length + Sepal.Width)`
Membuat dan menggabungkan variabel *sepal* yang merupakan jumlahan antara Length dan Width
- `mutate_each(iris, funs(min_rank))`
Menerapkan fungsi *window* ke masing-masing kolom
- `transmute(iris, sepal=Sepal.Length + Sepal.Width)`
Membuat variabel baru *sepal* dan menghilangkan kolom lainnya

Mutate menggunakan fungsi *window*, fungsi yang mengambil vektor sebagai masukan dan menghasilkan vektor juga, seperti

lead	cumall
lag	cumany
dense_rank	cummean
min_rank	cumsum
percent_rank	cummax
row_number	cummin
ntile	cumprod
between	pmax
cume_dist	pmin

Grouping Data

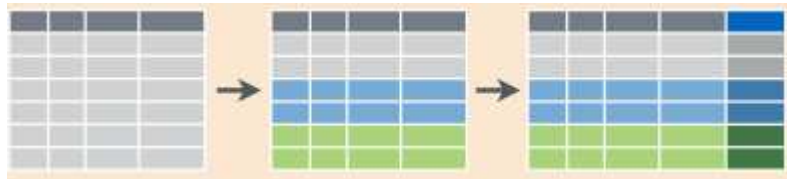
- `group_by(iris, Species)`
Mengelompokkan data menjadi baris yang bersesuaian dengan data Species
- `ungroup(iris)`
Menghilangkan informasi pengelompokan dari dataframe
- `iris %>% group_by(Species) %>% summarise(...)`



Gambar 13.42

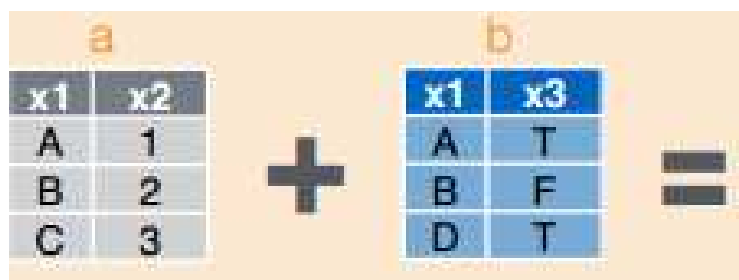
Mengelompokkan data iris berdasarkan Species dan diringkas berdasarkan fungsi *summarise*

- iris %>% group_by(Species) %>% mutate(...)



Gambar 13.43

Combine Datasets



Gambar 13.44

⇒ Mutating Join

- left_join(a,b,by="x1")
Menggabungkan baris dari data b ke a\

x1	x2	x3
A	1	T
B	2	F
C	3	NA

- right_join(a,b,by="x1")
Menggabungkan baris dari a ke b

x1	x3	x2
A	T	1
B	F	2
D	T	NA

- inner_join(a,b,by="x1")
Menggabungkan data yang beririsan.

x1	x2	x3
A	1	T
B	2	F

- full_join(a,b,by="x1")
Menggabungkan data. Semua data yang ada.

x1	x2	x3
A	1	T
B	2	F
C	3	NA
D	NA	T

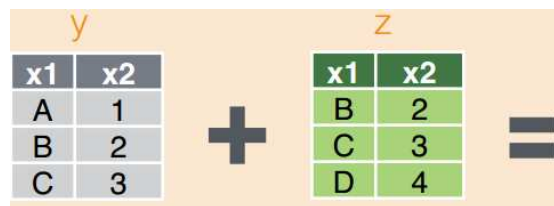
⇒ Filtering Join

- `semi_join(a, b, by="x1")`
Semua baris di a yang memiliki sama dengan b
- `anti_join(a, b, by="x1")`
Semua data yang tidak beririsan ditampilkan.

x1	x2
A	1
B	2

x1	x2
C	3

⇒ Operasi Himpunan



Gambar 13.45

- `intersect(y, z)`
Baris yang ada tepat di y dan z
- `union(y, z)`
Menampilkan dengan menggabungkan dua-duanya
- `setdiff(y, z)`
Baris yang ada di y tapi tidak di z
- `bind_rows(y, z)`
Menambahkan z ke y sebagai baris baru
- `bind_cols(y, z)`
Menambahkan z ke y sebagai kolom baru

x1	x2
B	2
C	3

x1	x2
A	1
B	2
C	3
D	4

x1	x2
A	1

x1	x2
A	1
B	2
C	3
B	2
C	3
D	4

x1	x2	x1	x2
A	1	B	2
B	2	C	3
C	3	D	4