



Chapter 8: Data Frame và Tidyverse

Exercise 1: Tạo dataframe

- Tạo dataframe như hình

	Age	Height	Weight	Sex
Alex	25	177	57	F
Lilly	31	163	69	F
Mark	23	190	83	M
Oliver	52	179	75	M
Martha	76	163	70	F
Lucas	49	183	83	M
Caroline	26	164	53	F

- In các level của Sex
- Tạo dataframe mới như hình sau (có row.names là Name giống data frame trên):

	Working
Alex	Yes
Lilly	No
Mark	No
Oliver	Yes
Martha	Yes
Lucas	No
Caroline	Yes

- Tạo data frame mới chứa 2 data frame trên

	Age	Height	Weight	Sex	Working
Alex	25	177	57	F	Yes
Lilly	31	163	69	F	No
Mark	23	190	83	M	No
Oliver	52	179	75	M	Yes
Martha	76	163	70	F	Yes
Lucas	49	183	83	M	No
Caroline	26	164	53	F	Yes

- Cho biết data frame mới này có bao nhiêu dòng và bao nhiêu cột?
- Cho biết kiểu dữ liệu của từng cột

Exercise 2: Tạo và làm việc với dataframe

- Cho 3 vector: a <- (floor(runif(10, -10, 10))), b <- letters[4:13], c <- c("yes","no","no","no","no","yes","no","yes","yes","no")
- Tạo data frame từ 3 vector trên, in data frame



	a	b	c
1	-7	d	yes
2	-8	e	no
3	-4	f	no
4	-9	g	no
5	5	h	no
6	0	i	yes
7	-7	j	no
8	-8	k	yes
9	8	l	yes
10	-2	m	no

- In data frame với thứ tự giá trị được sắp tăng dần trong cột a

	a	b	c
4	-9	g	no
2	-8	e	no
8	-8	k	yes
1	-7	d	yes
7	-7	j	no
3	-4	f	no
10	-2	m	no
6	0	i	yes
5	5	h	no
9	8	l	yes

- Cho ma trận: matrix.data <- matrix(1:40, nrow = 10, ncol = 4)
- Tạo dataframe từ ma trận này
- Đặt tên cho cột và dòng của ma trận như hình dưới và in kết quả:

	variable_1	variable_2	variable_3	variable_4
id_1	1	11	21	31
id_2	2	12	22	32
id_3	3	13	23	33
id_4	4	14	24	34
id_5	5	15	25	35
id_6	6	16	26	36
id_7	7	17	27	37
id_8	8	18	28	38
id_9	9	19	29	39
id_10	10	20	30	40

Exercise 3: Làm việc với dataframe

- Sử dụng dữ liệu sẵn của R là state.x77
- In dữ liệu
- Cho biết kiểu của dữ liệu?
- Kiểu này có phải là dataframe không? Nếu không thì chuyển dữ liệu này thành dataframe
- Trong dataframe trên, cho biết có bao nhiêu state có income <4300. Đó là những state nào?
- Cho biết state nào có income cao nhất và là bao nhiêu?
- Cho biết state nào có Life.Expect cao nhất và là bao nhiêu?
- Cho biết state nào có Life.Expect thấp nhất và là bao nhiêu?



```
[1] "state has max income: Alaska 6315"
[1] "state has highest life.exp: Hawaii 73.6"
[1] "state has lowest life.exp: South Carolina 67.96"
```

Exercise 4: Làm việc với dataframe

- Tạo data frame từ các dữ liệu mà R cho sẵn như sau: state.abb, state.area, state.division, state.name, state.region. In head của dataframe này.

	state.abb	state.area	state.division	state.region	
Alabama	AL	51609	East	South Central	South
Alaska	AK	589757		Pacific	West
Arizona	AZ	113909		Mountain	West
Arkansas	AR	53104	West	South Central	South
California	CA	158693		Pacific	West
Colorado	CO	104247		Mountain	West

- Đổi tên cho tất cả các cột trong dataframe với tên chỉ chứa 3 ký tự sau dấu . của các tên cột đang có. In head dataframe sau khi đổi tên.

	abb	are	div	reg	
Alabama	AL	51609	East	South Central	South
Alaska	AK	589757		Pacific	West
Arizona	AZ	113909		Mountain	West
Arkansas	AR	53104	West	South Central	South
California	CA	158693		Pacific	West
Colorado	CO	104247		Mountain	West

- Tạo data frame mới chứa state.x77 và data frame vừa tạo. In head của data frame này.

	Population	Income	Illiteracy	Life Exp	Murder	HS Grad	Frost	Area	abb	are	div	reg
Alabama	3615	3624	2.1	69.05	15.1	41.3	20	50708	AL	51609	East South Central	South
Alaska	365	6315	1.5	69.31	11.3	66.7	152	566432	AK	589757		Pacific West
Arizona	2212	4530	1.8	70.55	7.8	58.1	15	113417	AZ	113909		Mountain West
Arkansas	2110	3378	1.9	70.66	10.1	39.9	65	51945	AR	53104	West South Central	South
California	21198	5114	1.1	71.71	10.3	62.6	20	156361	CA	158693		Pacific West
Colorado	2541	4884	0.7	72.06	6.8	63.9	166	103766	CO	104247		Mountain West

- Loại bỏ các cột div, Life Exp, HS Grad, Frost, abb, và are. In head của dataframe sau khi loại bỏ các cột.

	Population	Income	Illiteracy	Murder	Area	reg
Alabama	3615	3624	2.1	15.1	50708	South
Alaska	365	6315	1.5	11.3	566432	West
Arizona	2212	4530	1.8	7.8	113417	West
Arkansas	2110	3378	1.9	10.1	51945	South
California	21198	5114	1.1	10.3	156361	West
Colorado	2541	4884	0.7	6.8	103766	West

- Thêm một cột mới vào data frame với mô tả như sau: tên cột Illiteracy.Levels, giá trị của từng phần tử sẽ là Low nếu Illiteracy <1, là Some nếu Illiteracy <2, còn lại là High (gợi ý: dùng ifelse()). In head của dataframe này.

	Population	Income	Illiteracy	Murder	Area	reg	Illiteracy.Levels
Alabama	3615	3624	2.1	15.1	50708	South	High
Alaska	365	6315	1.5	11.3	566432	West	Some
Arizona	2212	4530	1.8	7.8	113417	West	Some
Arkansas	2110	3378	1.9	10.1	51945	South	Some
California	21198	5114	1.1	10.3	156361	West	Some
Colorado	2541	4884	0.7	6.8	103766	West	Low

- Cho biết trong data frame có bao nhiêu region và đó là những region nào?

[1] "Number of Regions: 4 . There are: Northeast, South, North Central, West"

Exercise 5: Sử dụng function trong package tidyverse

- Cho dữ liệu Obesity_data.csv. Đọc dữ liệu vào dataframe data. In head của data

	id	gender	height	weight	bmi	age	bmc	bmd	fat	lean	pcfat
1	1	F	150	49	21.8	53	1312	0.88	17802	28600	37.3
2	2	M	165	52	19.1	65	1309	0.84	8381	40229	16.8
3	3	F	157	57	23.1	64	1230	0.84	19221	36057	34.0
4	4	F	156	53	21.8	56	1171	0.80	17472	33094	33.8
5	5	M	160	51	19.9	54	1681	0.98	7336	40621	14.8
6	6	F	153	47	20.1	52	1358	0.91	14904	30068	32.2

- Tạo dataframe data1 chỉ chứa các cột id, gender, height, weight, bmi. In head của data1

	gender	height	weight	bmi	age
1	F	150	49	21.8	53
2	M	165	52	19.1	65
3	F	157	57	23.1	64
4	F	156	53	21.8	56
5	M	160	51	19.9	54
6	F	153	47	20.1	52

- Với data1, lọc ra những dòng có bmi ≥ 18.5 và ≤ 24.9 và đưa vào dataframe data2

	gender	height	weight	bmi	age
1	F	150	49	21.8	53
2	M	165	52	19.1	65
3	F	157	57	23.1	64
4	F	156	53	21.8	56
5	M	160	51	19.9	54
6	F	153	47	20.1	52

- Với data1, tạo ta biến mới height_m = height/100

	gender	height	weight	bmi	age	height_m
	F	150	49	21.8	53	1.50
	M	165	52	19.1	65	1.65
	F	157	57	23.1	64	1.57
	F	156	53	21.8	56	1.56
	M	160	51	19.9	54	1.60
	F	153	47	20.1	52	1.53

- Với data1, sắp tăng dần theo bmi



gender	height	weight	bmi	age	height_m
M	162	38	14.5	55	1.62
F	162	40	15.2	54	1.62
F	151	35	15.4	33	1.51
F	155	37	15.4	44	1.55
F	150	35	15.6	24	1.50
M	169	45	15.8	50	1.69

- Với data1, Tính giá trị trung bình của height, weight theo gender. Tạo ra các biến mới chứa giá trị trung bình là mean.height, mean.weight

gender	count	mean.height	mean.weight
F	862	153.2912	52.31090
M	355	165.0592	62.02254

- Với data1, đếm số lượng theo gender và age
- Chọn 10 mẫu ngẫu nhiên từ data1 và đưa vào data3
- Chọn 1% mẫu ngẫu nhiên từ data1 và đưa vào data4

Gợi ý

Exercise 1: Tạo dataframe

```
In [1]: Name <- c("Alex", "Lilly", "Mark", "Oliver", "Martha", "Lucas", "Caroline")
Age <- c(25, 31, 23, 52, 76, 49, 26)
Height <- c(177, 163, 190, 179, 163, 183, 164)
Weight <- c(57, 69, 83, 75, 70, 83, 53)
Sex <- as.factor(c("F", "F", "M", "M", "F", "M", "F"))
df <- data.frame (row.names = Name, Age, Height, Weight, Sex)
```

```
In [5]: print(df)
```

	Age	Height	Weight	Sex
Alex	25	177	57	F
Lilly	31	163	69	F
Mark	23	190	83	M
Oliver	52	179	75	M
Martha	76	163	70	F
Lucas	49	183	83	M
Caroline	26	164	53	F

```
In [3]: print(paste("Levels of Sex:",toString(levels(df$Sex))))
```

```
[1] "Levels of Sex: F, M"
```

```
In [6]: Working <- c("Yes", "No", "No", "Yes", "Yes", "No", "Yes")
df2 <- data.frame(row.names = Name, Working) #Name has been already defined in [5]
print(df2)
```

	Working
Alex	Yes
Lilly	No
Mark	No
Oliver	Yes
Martha	Yes
Lucas	No
Caroline	Yes

```
In [7]: # tao data frame moi chua thong tin cua 2 data frame tren
df <- cbind(df, df2)
print("Combining 2 data frame:")
print(df)
```

[1] "Combining 2 data frame:"

	Age	Height	Weight	Sex	Working
Alex	25	177	57	F	Yes
Lilly	31	163	69	F	No
Mark	23	190	83	M	No
Oliver	52	179	75	M	Yes
Martha	76	163	70	F	Yes
Lucas	49	183	83	M	No
Caroline	26	164	53	F	Yes

```
In [8]: #co bao nhieu dong va bao nhieu cot trong data frame nay?
print(paste("Dong:", dim(df)[1], ", Cot:", dim(df)[2]))
# cho biet kieu du lieu cua tung cot
print(str(df))
```

[1] "Dong: 7 , Cot: 5"

```
'data.frame': 7 obs. of 5 variables:
 $ Age    : num  25 31 23 52 76 49 26
 $ Height : num  177 163 190 179 163 183 164
 $ Weight : num  57 69 83 75 70 83 53
 $ Sex    : Factor w/ 2 levels "F","M": 1 1 2 2 1 2 1
 $ Working: Factor w/ 2 levels "No","Yes": 2 1 1 2 2 1 2
NULL
```

Exercise 2: Tạo và làm việc với dataframe



```
In [9]: a <- (floor(runif(10, -10, 10)))
b <- letters[4:13]
c <- c("yes", "no", "no", "no", "no", "yes", "no", "yes", "yes", "no")
#tao data frame
df3 <- data.frame(a,b,c)
print("Data frame:")
print(df3)
```

```
[1] "Data frame:"
  a b  c
1 -7 d yes
2 -8 e no
3 -4 f no
4 -9 g no
5  5 h no
6  0 i yes
7 -7 j no
8 -8 k yes
9  8 l yes
10 -2 m no
```

```
In [10]: #in data frame voi du lieu sap xep tang dan o cot a
print("Data frame with column a in order")
print(df3[with (df3, order(a)),] )
```

```
[1] "Data frame with column a in order"
  a b  c
4 -9 g no
2 -8 e no
8 -8 k yes
1 -7 d yes
7 -7 j no
3 -4 f no
10 -2 m no
6  0 i yes
5  5 h no
9  8 l yes
```



In [11]: #cho ma tran

```
matrix.data <- matrix(1:40, nrow = 10, ncol = 4)
#tao data frame tu ma tran
print("Data frame form matrix:")
df <- as.data.frame(matrix.data)
#dat ten cot va dong cho ma tran nhu sau
colnames(df) <- sub(" ", "", paste("variable_", 1:ncol(df)))
rownames(df) <- sub(" ", "", paste("id_", 1:nrow(df)))
print(df)
```

```
[1] "Data frame form matrix:"
  variable_1 variable_2 variable_3 variable_4
id_1          1          11          21          31
id_2          2          12          22          32
id_3          3          13          23          33
id_4          4          14          24          34
id_5          5          15          25          35
id_6          6          16          26          36
id_7          7          17          27          37
id_8          8          18          28          38
id_9          9          19          29          39
id_10        10          20          30          40
```

Exercise 3: Làm việc với dataframe

In [15]: print(head(state.x77))
print(paste("Data type:", class(state.x77)))

	Population	Income	Illiteracy	Life	Exp	Murder	HS	Grad	Frost	Area
Alabama	3615	3624	2.1	69.05	15.1	41.3	20	50708		
Alaska	365	6315	1.5	69.31	11.3	66.7	152	566432		
Arizona	2212	4530	1.8	70.55	7.8	58.1	15	113417		
Arkansas	2110	3378	1.9	70.66	10.1	39.9	65	51945		
California	21198	5114	1.1	71.71	10.3	62.6	20	156361		
Colorado	2541	4884	0.7	72.06	6.8	63.9	166	103766		

[1] "Data type: matrix"

In [16]: df <- data.frame(state.x77)
print(head(df))

	Population	Income	Illiteracy	Life.Exp	Murder	HS.Grad	Frost	Area
Alabama	3615	3624	2.1	69.05	15.1	41.3	20	50708
Alaska	365	6315	1.5	69.31	11.3	66.7	152	566432
Arizona	2212	4530	1.8	70.55	7.8	58.1	15	113417
Arkansas	2110	3378	1.9	70.66	10.1	39.9	65	51945
California	21198	5114	1.1	71.71	10.3	62.6	20	156361
Colorado	2541	4884	0.7	72.06	6.8	63.9	166	103766



In [18]: # Find out how many states have an income of Less than 4300.

```
print(paste("Number of states income < 4300:", nrow(df[df$Income < 4300,])))
# what are they?
df[df$Income < 4300,]
```

[1] "Number of states income < 4300: 20"

	Population	Income	Illiteracy	Life.Exp	Murder	HS.Grad	Frost	Area
Alabama	3615	3624	2.1	69.05	15.1	41.3	20	50708
Arkansas	2110	3378	1.9	70.66	10.1	39.9	65	51945
Georgia	4931	4091	2.0	68.54	13.9	40.6	60	58073
Idaho	813	4119	0.6	71.87	5.3	59.5	126	82677
Kentucky	3387	3712	1.6	70.10	10.6	38.5	95	39650
Louisiana	3806	3545	2.8	68.76	13.2	42.2	12	44930
Maine	1058	3694	0.7	70.39	2.7	54.7	161	30920
Mississippi	2341	3098	2.4	68.09	12.5	41.0	50	47296
Missouri	4767	4254	0.8	70.69	9.3	48.8	108	68995
New Hampshire	812	4281	0.7	71.23	3.3	57.6	174	9027
New Mexico	1144	3601	2.2	70.32	9.7	55.2	120	121412
North Carolina	5441	3875	1.8	69.21	11.1	38.5	80	48798
Oklahoma	2715	3983	1.1	71.42	6.4	51.6	82	68782
South Carolina	2816	3635	2.3	67.96	11.6	37.8	65	30225
South Dakota	681	4167	0.5	72.08	1.7	53.3	172	75955
Tennessee	4173	3821	1.7	70.11	11.0	41.8	70	41328
Texas	12237	4188	2.2	70.90	12.2	47.4	35	262134
Utah	1203	4022	0.6	72.90	4.5	67.3	137	82096
Vermont	472	3907	0.6	71.64	5.5	57.1	168	9267
West Virginia	1799	3617	1.4	69.48	6.7	41.6	100	24070



```
In [19]: # Find out which is the state with the highest income.
ma <- df[which.max(df$Income),][2]
print(paste("state has max income:",
            row.names(df[which.max(df$Income),]),
            toString(ma)))
# Find out which is the state with the highest Life.Exp.
ma <- df[which.max(df$Life.Exp),][4]
print(paste("state has highest life.exp:",
            row.names(df[which.max(df$Life.Exp),]),
            toString(ma)))
# Find out which is the state with the lowest Life.Exp.
ma <- df[which.min(df$Life.Exp),][4]
print(paste("state has lowest life.exp:",
            row.names(df[which.min(df$Life.Exp),]),
            toString(ma)))
```

```
[1] "state has max income: Alaska 6315"
[1] "state has highest life.exp: Hawaii 73.6"
[1] "state has lowest life.exp: South Carolina 67.96"
```

Exercise 4: Làm việc với dataframe

```
In [30]: # tao data frame tu state.abb, state.area, state.division, state.name, state.region
# row names la ten cua states.
df <- data.frame(state.abb, state.area, state.division, state.name, state.region,
                  row.names = state.name)
print("Data frame:")
print(head(df))
```

```
[1] "Data frame:"
      state.abb state.area     state.division state.name state.region
Alabama        AL      51609   East South Central       South
Alaska         AK      589757          Pacific       West
Arizona        AZ      113909          Mountain       West
Arkansas       AR      53104   West South Central       South
California    CA      158693          Pacific       West
Colorado       CO      104247          Mountain       West
```

```
In [31]: # doi ten tat cac cac cot chi chua 3 ky tu sau dau .
colnames(df) <- substr(colnames(df), 7, 9)
# sau khi doi ten
print("After rename:")
print(head(df))
```

```
[1] "After rename:"
      abb  are      div  reg
Alabama  AL  51609  East South Central South
Alaska   AK  589757          Pacific  West
Arizona  AZ  113909          Mountain West
Arkansas AR  53104  West South Central South
California CA  158693          Pacific  West
Colorado  CO  104247          Mountain West
```



```
In [32]: # tao data frame chua state.x77 va du lieu moi tao
df_new <- cbind(state.x77, df)
print("New data frame:")
print(head(df_new))
```

[1] "New data frame:"

	Population	Income	Illiteracy	Life Exp	Murder	HS	Grad	Frost	Area
Alabama	3615	3624	2.1	69.05	15.1	41.3	20	50708	
Alaska	365	6315	1.5	69.31	11.3	66.7	152	566432	
Arizona	2212	4530	1.8	70.55	7.8	58.1	15	113417	
Arkansas	2110	3378	1.9	70.66	10.1	39.9	65	51945	
California	21198	5114	1.1	71.71	10.3	62.6	20	156361	
Colorado	2541	4884	0.7	72.06	6.8	63.9	166	103766	
	abb	are		div	reg				
Alabama	AL	51609	East	South	Central	South			
Alaska	AK	589757			Pacific	West			
Arizona	AZ	113909			Mountain	West			
Arkansas	AR	53104	West	South	Central	South			
California	CA	158693			Pacific	West			
Colorado	CO	104247			Mountain	West			

```
In [33]: # Loai bo cot div
df_new$div <- NULL
# tiep tuc loai bo cac cot Life Exp, HS Grad, Frost, abb, va are
df_new <- subset(df_new,select = -c(4, 6, 7, 9, 10))

print("After drop div, Life Exp, HS Grad, Frost, abb, and are:")
print(head(df_new))
```

[1] "After drop div, Life Exp, HS Grad, Frost, abb, and are:"

	Population	Income	Illiteracy	Murder	Area	reg
Alabama	3615	3624	2.1	15.1	50708	South
Alaska	365	6315	1.5	11.3	566432	West
Arizona	2212	4530	1.8	7.8	113417	West
Arkansas	2110	3378	1.9	10.1	51945	South
California	21198	5114	1.1	10.3	156361	West
Colorado	2541	4884	0.7	6.8	103766	West



```
In [34]: # them cot phan Loai mu chu Illiteracy.Levels voi mo ta sau: <1 Low, <2 So
# con Lai la High
df_new$Illiteracy.Levels <- ifelse(df_new$Illiteracy >= 0 &
                                      df_new$Illiteracy < 1,
                                      "Low",
                                      ifelse(df_new$Illiteracy >= 1
                                            & df_new$Illiteracy < 2,
                                            "Some",
                                            "High"))

print("After insert Illiteracy.Levels:")
print(head(df_new))
```

[1] "After insert Illiteracy.Levels:"

	Population	Income	Illiteracy	Murder	Area	reg	Illiteracy.Levels	
Alabama	3615	3624		2.1	15.1	50708	South	High
Alaska	365	6315		1.5	11.3	566432	West	Some
Arizona	2212	4530		1.8	7.8	113417	West	Some
Arkansas	2110	3378		1.9	10.1	51945	South	Some
California	21198	5114		1.1	10.3	156361	West	Some
Colorado	2541	4884		0.7	6.8	103766	West	Low

```
In [35]: # cho biet co bao nhieu vung va la nhung vung nao?
print(paste("Number of Regions:", nlevels(df_new$reg), ". There are:",
           toString(levels(df_new$reg))))
```

[1] "Number of Regions: 4 . There are: Northeast, South, North Central, West"

Exercise 5: Sử dụng function trong package tidyverse

```
In [36]: library(tidyverse)
```

Loading tidyverse: ggplot2
 Loading tidyverse: tibble
 Loading tidyverse: tidyr
 Loading tidyverse: readr
 Loading tidyverse: purrr
 Loading tidyverse: dplyr

Conflicts with tidy packages -----

-
 filter(): dplyr, stats
 lag(): dplyr, stats



```
In [37]: data <- read.csv('Obesity_data.csv')
print(head(data))
```

	id	gender	height	weight	bmi	age	bmc	bmd	fat	lean	pcfat
1	1	F	150	49	21.8	53	1312	0.88	17802	28600	37.3
2	2	M	165	52	19.1	65	1309	0.84	8381	40229	16.8
3	3	F	157	57	23.1	64	1230	0.84	19221	36057	34.0
4	4	F	156	53	21.8	56	1171	0.80	17472	33094	33.8
5	5	M	160	51	19.9	54	1681	0.98	7336	40621	14.8
6	6	F	153	47	20.1	52	1358	0.91	14904	30068	32.2

```
In [39]: # c1
data1 = data %>% select(c(2,3,4,5,6))
print(head(data1))
```

	gender	height	weight	bmi	age
1	F	150	49	21.8	53
2	M	165	52	19.1	65
3	F	157	57	23.1	64
4	F	156	53	21.8	56
5	M	160	51	19.9	54
6	F	153	47	20.1	52

```
In [51]: # c2
data11 = data %>% select(gender, height, weight, bmi, age)
head(data11)
```

	gender	height	weight	bmi	age
	F	150	49	21.8	53
	M	165	52	19.1	65
	F	157	57	23.1	64
	F	156	53	21.8	56
	M	160	51	19.9	54
	F	153	47	20.1	52

```
In [52]: data2 = data1 %>% filter(bmi >=18.5, bmi <=24.9)
```

```
In [53]: print(head(data2))
```

	gender	height	weight	bmi	age
1	F	150	49	21.8	53
2	M	165	52	19.1	65
3	F	157	57	23.1	64
4	F	156	53	21.8	56
5	M	160	51	19.9	54
6	F	153	47	20.1	52

```
In [54]: print(paste("Rows have bmi >=18.5 and <=24.9: ", nrow(data2)))
```

[1] "Rows have bmi >=18.5 and <=24.9: 865"



In [55]: `data1 = mutate(data1, height_m = height/100)`
`head(data1)`

gender	height	weight	bmi	age	height_m
F	150	49	21.8	53	1.50
M	165	52	19.1	65	1.65
F	157	57	23.1	64	1.57
F	156	53	21.8	56	1.56
M	160	51	19.9	54	1.60
F	153	47	20.1	52	1.53

In [56]: `head(arrange(data1, bmi))`

gender	height	weight	bmi	age	height_m
M	162	38	14.5	55	1.62
F	162	40	15.2	54	1.62
F	151	35	15.4	33	1.51
F	155	37	15.4	44	1.55
F	150	35	15.6	24	1.50
M	169	45	15.8	50	1.69

In [59]: `group = group_by(data1, gender)`
`summary = summarize(group,`
`count = n(),`
`mean.height = mean(height, na.rm = T),`
`mean.weight = mean(weight, na.rm = T)`
`)`

In [60]: `summary`

gender	count	mean.height	mean.weight
F	862	153.2912	52.31090
M	355	165.0592	62.02254

In [62]: `count(data1, gender)`

gender	n
F	862
M	355



```
In [73]: group_gender_age = count(data1, gender, age)
head(group_gender_age)
```

gender	age	n
F	14	4
F	16	2
F	18	10
F	19	27
F	20	13
F	21	8

```
In [74]: tail(group_gender_age)
```

gender	age	n
M	82	1
M	83	1
M	84	3
M	85	1
M	87	1
M	88	1

```
In [66]: data3 = data1 %>% sample_n(10)
```

```
In [67]: data3
```

	gender	height	weight	bmi	age	height_m
1075	F	152	55	23.8	52	1.52
868	M	171	71	24.3	50	1.71
369	F	158	51	20.4	46	1.58
758	F	136	48	26.0	72	1.36
1198	M	150	45	20.0	59	1.50
318	M	166	52	18.9	26	1.66
789	F	145	45	21.4	80	1.45
672	F	159	50	19.8	22	1.59
57	F	154	43	18.1	48	1.54
181	M	160	54	21.1	65	1.60

```
In [70]: data4 = data %>% sample_frac(0.01) # 1 percentages
```



In [71]: data4

	id	gender	height	weight	bmi	age	bmc	bmd	fat	lean	pcfat
713	719	F	150	46	20.4	30	2055	1.26	15302	28551	33.3
1080	1090	F	139	36	18.6	79	1111	0.89	12075	23204	33.2
288	291	F	152	42	18.2	55	1491	0.96	8986	30893	21.7
76	76	F	151	57	25.0	54	1345	0.85	24067	30986	42.7
787	794	F	155	50	20.8	45	1718	1.05	16897	30004	34.8
842	849	F	161	56	21.6	23	1675	0.96	19429	34252	35.1
147	148	M	167	76	27.3	40	2184	1.08	25043	47546	33.5
776	782	F	155	59	24.6	46	1770	1.12	16907	30175	34.6
709	715	M	174	58	19.2	24	2193	1.12	10680	42376	19.3
628	634	M	183	95	28.4	18	2912	1.21	29944	58217	32.9
129	130	F	151	57	25.0	43	1689	1.02	21875	34088	37.9
91	92	F	151	51	22.4	61	1121	0.78	20464	33941	36.6