0.

1’ It has column names: Gender, Height, Weight, separator is “,”

1.

a’ several ways: use df[1:10,], or head(df, 10), tail(df, 10)

```
> df <- read.csv('heights_weights_genders2.csv',header = TRUE)
> View(df)
> head(df,10)
   Gender   Height   Weight
1  Female 67.38180 144.3067
2  Female 62.64413 118.1879
3    Male 69.20824 188.3849
4    Male 70.12350 186.7590
5    Male 74.22117 231.1302
6    Male 69.65044 198.8437
7    Male 72.38757 199.4469
8  Female 60.80842 110.9842
9  Female 61.96796 133.2625
10 Female 63.18299 147.9974
> tail(df,10)
      Gender   Height   Weight
9994    Male 71.58327 193.0855
9995    Male 67.77991 173.3880
9996    Male 64.23991 169.8508
9997    Male 74.78571 239.4643
9998  Female 63.86484 158.9401
9999    Male 73.81539 231.3745
10000   Male 66.87188 170.3240
10001 Female 65.87502 146.9627
10002 Female 62.72472 158.2855
10003 Female 64.38552 145.3501
```

When use as.is.

```
> df <- read.csv('heights_weights_genders2.csv',header = TRUE,as.is =
T,sep = ",")
> head(df,10)
   Gender   Height   Weight
1  Female 67.38180 144.3067
2  Female 62.64413 118.1879
3    Male 69.20824 188.3849
4    Male 70.12350 186.7590
5    Male 74.22117 231.1302
6    Male 69.65044 198.8437
7    Male 72.38757 199.4469
8  Female 60.80842 110.9842
9  Female 61.96796 133.2625
10 Female 63.18299 147.9974
> tail(df,10)
      Gender   Height   Weight
9994    Male 71.58327 193.0855
9995    Male 67.77991 173.3880
9996    Male 64.23991 169.8508
9997    Male 74.78571 239.4643
9998  Female 63.86484 158.9401
9999    Male 73.81539 231.3745
10000   Male 66.87188 170.3240
10001 Female 65.87502 146.9627
10002 Female 62.72472 158.2855
10003 Female 64.38552 145.3501
```

b’ 10003, 3

```
> str(df)
'data.frame':   10003 obs. of  3 variables:
 $ Gender: Factor w/ 2 levels "Female","Male": 1 1 2 2 2 2 2 1 1 1 ...
 $ Height: num  67.4 62.6 69.2 70.1 74.2 ...
 $ Weight: num  144 118 188 187 231 ...
> dim(df)
[1] 10003     3
> nrow(df)
[1] 10003
> ncol(df)
[1] 3
```

Since it is stange, that means Gender has some other value, when using as.is, it behaves good. But we can’ t find differences through this way.

```
> str(df)
'data.frame':   10003 obs. of  3 variables:
 $ Gender: chr  "Female" "Female" "Male" "Male" ...
 $ Height: num  67.4 62.6 69.2 70.1 74.2 ...
 $ Weight: num  144 118 188 187 231 ...
> dim(df)
[1] 10003     3
> nrow(df)
[1] 10003
> ncol(df)
[1] 3
```

So latter I think it’ s better for us to don’ t use as.is in this assignment.

C’ factor numeric numeric

```
> mode(df[,1])   > class(df[,1])
[1] "numeric"    [1] "factor"
> mode(df[,2])   > class(df[,2])
[1] "numeric"    [1] "numeric"
> mode(df[,3])   > class(df[,3])
[1] "numeric"    [1] "numeric"
```

ß

D’ Height.max = 79.00, Height.min = 54.26, Weight.max = 270.0, Weight.min = 64.7
When give as.is attribute.

```
> summary(df)
    Gender         Height          Weight     
 Female:5000   Min.   :54.26   Min.   : 64.7  
 Male  :5000   1st Qu.:63.51   1st Qu.:135.8  
 NA's  :   3   Median :66.32   Median :161.2  
               Mean   :66.37   Mean   :161.4  
               3rd Qu.:69.17   3rd Qu.:187.2  
               Max.   :79.00   Max.   :270.0  
```

E’ 4425, 5266, 9222

```
> df[!complete.cases(df),]
     Gender   Height    Weight
4425   <NA> 78.99874  64.70013
5266   <NA> 66.31807 161.21293
9222   <NA> 66.36756 161.44036
```

```
> which(!complete.cases(df$Gender))
[1] 4425 5266 9222

> which(complete.cases(df$Gender) == FALSE)
[1] 4425 5266 9222
```

F’ 3

```
> dfn <- na.omit(df)
> nrow(dfn)
[1] 10000
> summary(dfn)
    Gender          Height           Weight     
 Female:5000   Min.   :54.26   Min.   : 64.7  
 Male  :5000   1st Qu.:63.51   1st Qu.:135.8  
               Median :66.32   Median :161.2  
               Mean   :66.37   Mean   :161.4  
               3rd Qu.:69.17   3rd Qu.:187.2  
               Max.   :79.00   Max.   :270.0  
> 
```

G'

```
> print(dfn[order(dfn$Height),])
      Gender   Height    Weight
9555  Female 54.26313  64.70013
5291  Female 54.61686  71.39375
4895  Female 54.87373  78.60667
6525  Female 55.14856  88.81241
5155  Female 55.33649  88.36658
345   Female 55.65189  85.62178
9107  Female 55.66820  68.98253
2416  Female 55.73974 108.12197
1363  Female 55.85121 103.76714
8294  Female 55.97920  85.41753
7479  Female 56.06664  89.57120
9973  Female 56.07870  94.48837
80    Female 56.09825 104.95410
1674  Female 56.10537  87.29887
5824  Female 56.10890  80.53126
4391  Female 56.15946  90.81526
8098  Female 56.16730  77.89856
697   Female 56.44569  96.64024
8989  Female 56.53417  97.74390
4350  Female 56.54797  84.87212
```

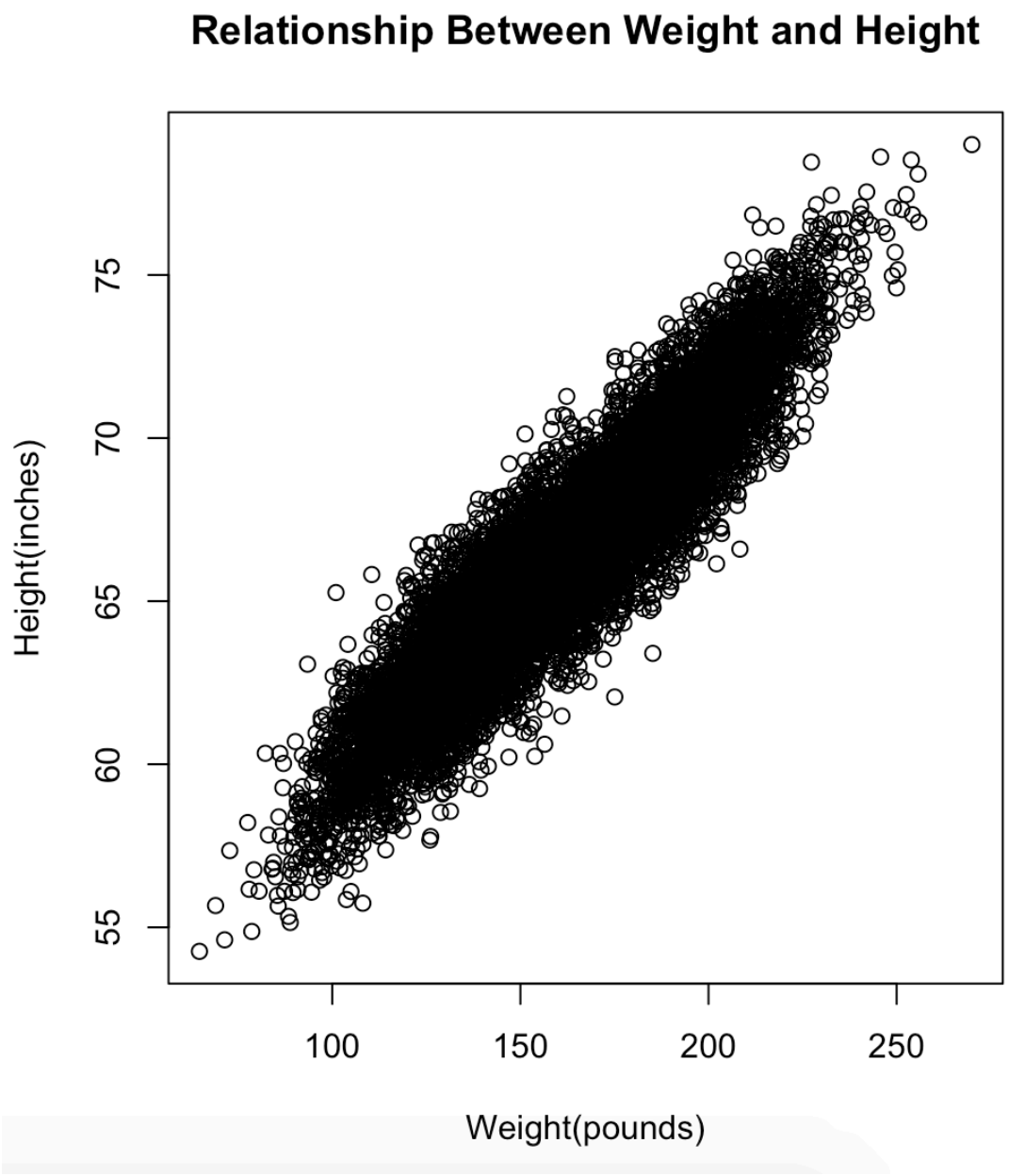
H'

```
> dfn$BMI <- dfn$Weight / (dfn$Height * dfn$Height)
> dfn
      Gender   Height    Weight        BMI
1     Female 67.38180 144.30671 0.03178347
2     Female 62.64413 118.18789 0.03011703
3       Male 69.20824 188.38488 0.03933059
4       Male 70.12350 186.75902 0.03797996
5       Male 74.22117 231.13019 0.04195667
6       Male 69.65044 198.84370 0.04098870
7       Male 72.38757 199.44690 0.03806267
8     Female 60.80842 110.98417 0.03001467
9     Female 61.96796 133.26253 0.03470354
10    Female 63.18299 147.99742 0.03707266
11    Female 67.48170 170.61708 0.03746714
12    Female 63.46075 127.30571 0.03161095
13    Female 65.96802 156.80208 0.03603172
14      Male 66.67790 174.74178 0.03930365
15      Male 68.30593 185.27962 0.03971100
16      Male 67.93687 188.88410 0.04092459
17      Male 68.52125 186.71880 0.03976834
18      Male 67.37249 187.19416 0.04124080
19      Male 67.45990 182.62935 0.04013093
```

2.

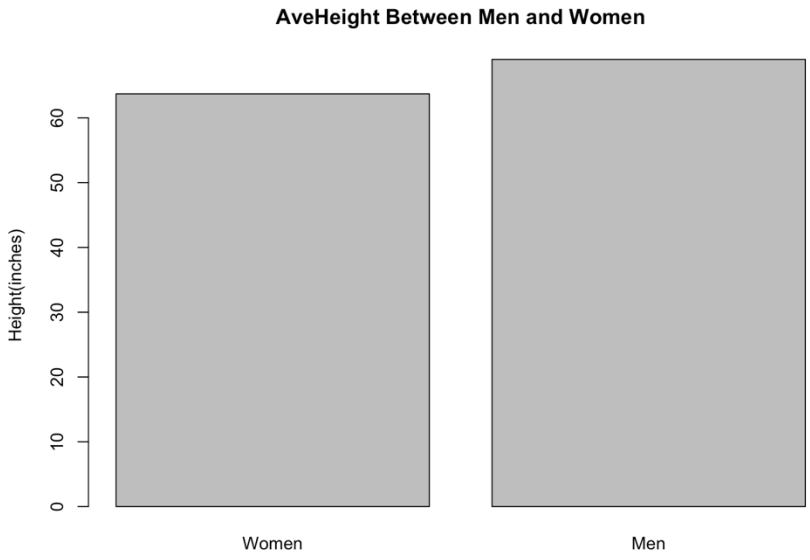
A’ . They are almost presenting a linear relationship.

```
> plot(dfn$Weight,dfn$Height,main= "Relationship Between Weight and He
ight", xlab = "Weight(pounds)", ylab = "Height(inches)")
```


Relationship Between Weight and Height
Height(inches)
55
60
65
70
75
100
150
200
250
Weight(pounds)

B’

```
> aveHeight.women <- mean(df[df$Gender == "Female",]$Height, trim = 0,na.rm = TRUE)
> aveHeight.men <- mean(df[df$Gender == "Male",]$Height, trim = 0,na.rm = TRUE)
> aveHeight.women
[1] 63.70877
> aveHeight.men
[1] 69.02635
> barplot(c(aveHeight.women,aveHeight.men),main = "AveHeight Between Men and Women", names.arg = c("Women","M
en"),  ylab = "Height(inches)")
```



C'

```
> barplot(c(aveHeight.women,aveHeight.men),main = "AveHeight Between Men and Women", names.arg = c("Women","m
en"),  ylab = "Height(inches)",col = c("blue","green"))
```