Week1_R

Akhila

5/2/2020

R Console Input and Evaluation

```
x <- 5
print(x) #explicit printing
```

```
## [1] 5
```

```
x #autoprinting
```

```
## [1] 5
```

```
msg <- "hello"
msg
```

```
## [1] "hello"
```

```
#create sequence (integers) with :
x1 <- 1:20
x1
```

```
##  [1]  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20
```

```
x2<- 0:10
x2
```

```
##  [1]  0  1  2  3  4  5  6  7  8  9 10
```

```
class(x2)
```

```
## [1] "integer"
```

```
as.numeric(x2)
```

```
##  [1]  0  1  2  3  4  5  6  7  8  9 10
```

```
as.logical(x2)
```

```
##  [1] FALSE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
```

```
as.character(x2)
```

```
##  [1] "0"  "1"  "2"  "3"  "4"  "5"  "6"  "7"  "8"  "9"  "10"
```

```
x3<-c("a","b","c")
as.numeric(x3)
```

```
## Warning: NAs introduced by coercion
```

```
## [1] NA NA NA
```

```r
x4<- list(1,"a",FALSE,1+4i)

x4
```

```
## [[1]]
## [1] 1
##
## [[2]]
## [1] "a"
##
## [[3]]
## [1] FALSE
##
## [[4]]
## [1] 1+4i
```

```r
x4[2]
```

```
## [[1]]
## [1] "a"
```

Matrices

```r
mat<- matrix(nrow=3,ncol=3)
mat
```

```
##      [,1] [,2] [,3]
## [1,]   NA   NA   NA
## [2,]   NA   NA   NA
## [3,]   NA   NA   NA
```

```r
mat1 <- matrix(1:9,nrow=3,ncol=3)
mat1
```

```
##      [,1] [,2] [,3]
## [1,]    1    4    7
## [2,]    2    5    8
## [3,]    3    6    9
```

```r
mat1[3,3]
```

```
## [1] 9
```

```r
dim(mat1)
```

```
## [1] 3 3
```

```r
attributes(mat1)
```

```
## $dim
## [1] 3 3
```

```r
mat2 <- 1:9
mat2
```

```
## [1] 1 2 3 4 5 6 7 8 9
```

```r
dim(mat2) <- c(3,3)
mat2
```

```
##      [,1] [,2] [,3]
## [1,]    1    4    7
## [2,]    2    5    8
## [3,]    3    6    9
```

```r
vect1<- 1:3
vect2<- 10:12

cbind(vect1,vect2)
```

```
##      vect1 vect2
## [1,]     1    10
## [2,]     2    11
## [3,]     3    12
```

```r
rbind(vect1,vect2)
```

```
##       [,1] [,2] [,3]
## vect1    1    2    3
## vect2   10   11   12
```

```r
cbind(mat2,vect2)
```

```
##            vect2
## [1,] 1 4 7    10
## [2,] 2 5 8    11
## [3,] 3 6 9    12
```

Factors

```r
f <- factor(c("y","y","n","y","n","y"))
f
```

```
## [1] y y n y n y
## Levels: n y
```

```r
table(f)
```

```
## f
## n y
## 2 4
```

```r
unclass(f)
```

```
## [1] 2 2 1 2 1 2
## attr(,"levels")
## [1] "n" "y"
```

```r
f1 <- factor(c("y","y","n","y","n","y"),levels=c("y","n"))
f1
```

```
## [1] y y n y n y
## Levels: y n
```

Missing values NA and NaNs

```
miss1 <- c(1,2,NA,4)
miss1
```

```
## [1]  1  2 NA  4
```

```
is.na(miss1)
```

```
## [1] FALSE FALSE  TRUE FALSE
```

```
is.nan(miss1)
```

```
## [1] FALSE FALSE FALSE FALSE
```

```
miss2 <- c(1,2,NaN,6,NA,4)
is.na(miss2)
```

```
## [1] FALSE FALSE  TRUE FALSE  TRUE FALSE
```

```
is.nan(miss2)
```

```
## [1] FALSE FALSE  TRUE FALSE FALSE FALSE
```

Attributes

```
df <- data.frame(f=0:3,b=c(F,T,T,T))
df
```

```
##   f     b
## 1 0 FALSE
## 2 1  TRUE
## 3 2  TRUE
## 4 3  TRUE
```

```
ncol(df)
```

```
## [1] 2
```

```
nrow(df)
```

```
## [1] 4
```

```
row.names.data.frame(df)
```

```
## [1] "1" "2" "3" "4"
```

```
df
```

```
##   f     b
## 1 0 FALSE
## 2 1  TRUE
## 3 2  TRUE
## 4 3  TRUE
```

```
nam <- c(1,2,3)
nam
```

```
## [1] 1 2 3
```

```
names(nam)
```

```
## NULL
```

```
names(nam)<- c("1","2","3")
nam1 <- list(a=1,b=3,c=5)
nam1
```

```
## $a
## [1] 1
##
## $b
## [1] 3
##
## $c
## [1] 5
```

```
mat1
```

```
##      [,1] [,2] [,3]
## [1,]    1    4    7
## [2,]    2    5    8
## [3,]    3    6    9
```

```
dimnames(mat1)<- list(c(1,2,3),c(10,11,12))
dimnames(mat1)
```

```
## [[1]]
## [1] "1" "2" "3"
##
## [[2]]
## [1] "10" "11" "12"
```

Reading and writing data

```
t1<- read.table("foo.txt",sep=",",header = F)
t1
```

```
##    V1         V2 V3
## 1   1     Akhila 21
## 2   2     Aarthy 20
## 3   3   Susheela 49
## 4   4 Padmanaban 56
## 5   5  Keerthana 20
## 6   6   Gayathri 21
## 7   7   Sandhiya 21
## 8   8   Prahanya 21
## 9   9   Jayasree 21
## 10 10   Priyanka 21
```

```
t2 <- read.table("foo.txt",sep=",",header = F,nrows=5)
c2<- sapply(t2,class)
tfinal <- read.table("foo.txt",sep=",",header = F, colClasses = c("integer","factor","numeric"))
tfinal
```

```
##    V1         V2 V3
```

```
## 1   1     Akhila 21
## 2   2     Aarthy 20
## 3   3   Susheela 49
## 4   4 Padmanaban 56
## 5   5  Keerthana 20
## 6   6   Gayathri 21
## 7   7   Sandhiya 21
## 8   8   Prahanya 21
## 9   9   Jayasree 21
## 10 10   Priyanka 21
```

```
c2
```

```
##        V1        V2        V3
## "integer"  "factor" "integer"
```

```
library(readr)
t3 <- read_csv("foo.txt",col_types = "ici",progress = T)
t3
```

```
## # A tibble: 9 x 3
##     `1` Akhila      `21`
##   <int> <chr>      <int>
## 1     2 Aarthy        20
## 2     3 Susheela      49
## 3     4 Padmanaban    56
## 4     5 Keerthana     20
## 5     6 Gayathri      21
## 6     7 Sandhiya      21
## 7     8 Prahanya      21
## 8     9 Jayasree      21
## 9    10 Priyanka      21
```

```
df
```

```
##   f     b
## 1 0 FALSE
## 2 1  TRUE
## 3 2  TRUE
## 4 3  TRUE
```

```
dput(df, "df.R")
new.df<- dget("df.R")
new.df
```

```
##   f     b
## 1 0 FALSE
## 2 1  TRUE
## 3 2  TRUE
## 4 3  TRUE
```

```
tfinal
```

```
##    V1         V2 V3
## 1   1     Akhila 21
## 2   2     Aarthy 20
## 3   3   Susheela 49
## 4   4 Padmanaban 56
## 5   5  Keerthana 20
```

```
## 6   6   Gayathri 21
## 7   7   Sandhiya 21
## 8   8   Prahanya 21
## 9   9   Jayasree 21
## 10 10   Priyanka 21
```

```
dump(c("tfinal","df"),file="df1.R")
rm(tfinal,df)
#tfinal
source("df1.R")
tfinal
```

```
##    V1         V2 V3
## 1   1     Akhila 21
## 2   2     Aarthy 20
## 3   3   Susheela 49
## 4   4 Padmanaban 56
## 5   5  Keerthana 20
## 6   6   Gayathri 21
## 7   7   Sandhiya 21
## 8   8   Prahanya 21
## 9   9   Jayasree 21
## 10 10   Priyanka 21
```

```
df
```

```
##   f     b
## 1 0 FALSE
## 2 1  TRUE
## 3 2  TRUE
## 4 3  TRUE
```

```
a <- data.frame(x = rnorm(100), y = runif(100))
b <- c(3, 4.4, 1 / 3)
a
```

```
##                x           y
## 1   -0.837802267 0.345489648
## 2    0.989038613 0.136529503
## 3   -0.397434440 0.415620409
## 4   -0.612886787 0.562410563
## 5    1.386448998 0.872511169
## 6    0.121295771 0.613567946
## 7    0.757515701 0.824738606
## 8   -0.347161759 0.857381343
## 9   -0.884843495 0.716037904
## 10  -0.984335333 0.967471538
## 11   0.438820354 0.169012549
## 12  -1.384517020 0.535297646
## 13   0.029606016 0.001950593
## 14  -1.394060738 0.377893453
## 15   0.212132563 0.839981135
## 16   1.327357702 0.053743226
## 17   1.672263653 0.313008683
## 18   0.184118683 0.769008568
## 19  -0.723628761 0.635735085
## 20  -0.582158084 0.436065426
```

```
## 21   0.236060179 0.405507138
## 22  -0.661183745 0.707500598
## 23   0.488220467 0.887143185
## 24   0.303275767 0.055772969
## 25   0.229194093 0.431730162
## 26  -1.925651145 0.077462175
## 27  -0.144046766 0.434821868
## 28  -0.014767843 0.335772324
## 29  -1.836982473 0.946563334
## 30  -0.377977781 0.588993004
## 31   1.692586816 0.975131270
## 32  -1.427454839 0.774064931
## 33  -3.225668143 0.891755613
## 34  -0.064903091 0.147901755
## 35  -0.653401787 0.446903353
## 36  -0.385643635 0.045203023
## 37  -0.072559897 0.031499791
## 38  -0.016333782 0.192338284
## 39  -1.100838677 0.195685755
## 40  -0.721266747 0.294973489
## 41   0.384430504 0.288373830
## 42   0.291366640 0.032620433
## 43  -0.643655194 0.146140232
## 44  -0.696446713 0.964565727
## 45   1.238061738 0.026601439
## 46   0.760258675 0.600592335
## 47  -0.029073780 0.582907897
## 48  -0.123670313 0.645017540
## 49  -0.108276602 0.744096377
## 50  -0.346549935 0.289391084
## 51   0.543543990 0.552265203
## 52   0.482107817 0.940146691
## 53  -1.320549914 0.230404194
## 54  -1.677198486 0.430649429
## 55   1.440282310 0.988790524
## 56   1.022943095 0.348650817
## 57   1.547814753 0.719865941
## 58   0.632057294 0.493821534
## 59   0.039849048 0.766033001
## 60  -0.356875672 0.062552764
## 61   0.170926145 0.769645875
## 62   1.402631795 0.924069031
## 63   0.468984737 0.714338980
## 64   0.387516177 0.719322243
## 65  -0.099437621 0.859535373
## 66  -0.227549078 0.171193709
## 67  -1.653564705 0.750180131
## 68  -0.410000879 0.121671815
## 69  -1.014713223 0.773781553
## 70   0.377800954 0.828641249
## 71   1.262714886 0.602048828
## 72   0.211659884 0.211314563
## 73  -0.032063060 0.362473505
## 74  -0.345256536 0.457762720
```

```
## 75  -1.026569524 0.696176335
## 76   0.835265568 0.603381027
## 77   0.147899709 0.002129533
## 78  -0.056345745 0.454066851
## 79  -0.116647728 0.514302555
## 80  -0.826554909 0.303212389
## 81  -0.458453234 0.980852609
## 82  -0.850696641 0.504879727
## 83  -0.213478931 0.285312673
## 84   0.150440021 0.759404889
## 85  -0.206555958 0.093845251
## 86   0.669001209 0.293732033
## 87   2.272002140 0.793346806
## 88   0.628200334 0.403941645
## 89  -0.002047917 0.967523502
## 90   0.818071319 0.112065395
## 91   0.194047800 0.872395358
## 92  -0.130964427 0.140082847
## 93  -1.217901761 0.529291230
## 94  -0.065248227 0.400714733
## 95   0.960811956 0.794007441
## 96  -1.124872603 0.672744813
## 97  -0.505648061 0.411487702
## 98   0.951289330 0.706302525
## 99  -0.759399565 0.612477406
## 100 -2.170500809 0.651304036
```

```
b
```

```
## [1] 3.0000000 4.4000000 0.3333333
```

```
save(a, b, file = "mydata.rda")
rm(a,b)
#a
load("mydata.rda")
a
```

```
##                x           y
## 1   -0.837802267 0.345489648
## 2    0.989038613 0.136529503
## 3   -0.397434440 0.415620409
## 4   -0.612886787 0.562410563
## 5    1.386448998 0.872511169
## 6    0.121295771 0.613567946
## 7    0.757515701 0.824738606
## 8   -0.347161759 0.857381343
## 9   -0.884843495 0.716037904
## 10  -0.984335333 0.967471538
## 11   0.438820354 0.169012549
## 12  -1.384517020 0.535297646
## 13   0.029606016 0.001950593
## 14  -1.394060738 0.377893453
## 15   0.212132563 0.839981135
## 16   1.327357702 0.053743226
## 17   1.672263653 0.313008683
## 18   0.184118683 0.769008568
```

```
## 19  -0.723628761 0.635735085
## 20  -0.582158084 0.436065426
## 21   0.236060179 0.405507138
## 22  -0.661183745 0.707500598
## 23   0.488220467 0.887143185
## 24   0.303275767 0.055772969
## 25   0.229194093 0.431730162
## 26  -1.925651145 0.077462175
## 27  -0.144046766 0.434821868
## 28  -0.014767843 0.335772324
## 29  -1.836982473 0.946563334
## 30  -0.377977781 0.588993004
## 31   1.692586816 0.975131270
## 32  -1.427454839 0.774064931
## 33  -3.225668143 0.891755613
## 34  -0.064903091 0.147901755
## 35  -0.653401787 0.446903353
## 36  -0.385643635 0.045203023
## 37  -0.072559897 0.031499791
## 38  -0.016333782 0.192338284
## 39  -1.100838677 0.195685755
## 40  -0.721266747 0.294973489
## 41   0.384430504 0.288373830
## 42   0.291366640 0.032620433
## 43  -0.643655194 0.146140232
## 44  -0.696446713 0.964565727
## 45   1.238061738 0.026601439
## 46   0.760258675 0.600592335
## 47  -0.029073780 0.582907897
## 48  -0.123670313 0.645017540
## 49  -0.108276602 0.744096377
## 50  -0.346549935 0.289391084
## 51   0.543543990 0.552265203
## 52   0.482107817 0.940146691
## 53  -1.320549914 0.230404194
## 54  -1.677198486 0.430649429
## 55   1.440282310 0.988790524
## 56   1.022943095 0.348650817
## 57   1.547814753 0.719865941
## 58   0.632057294 0.493821534
## 59   0.039849048 0.766033001
## 60  -0.356875672 0.062552764
## 61   0.170926145 0.769645875
## 62   1.402631795 0.924069031
## 63   0.468984737 0.714338980
## 64   0.387516177 0.719322243
## 65  -0.099437621 0.859535373
## 66  -0.227549078 0.171193709
## 67  -1.653564705 0.750180131
## 68  -0.410000879 0.121671815
## 69  -1.014713223 0.773781553
## 70   0.377800954 0.828641249
## 71   1.262714886 0.602048828
## 72   0.211659884 0.211314563
```

```
## 73  -0.032063060 0.362473505
## 74  -0.345256536 0.457762720
## 75  -1.026569524 0.696176335
## 76   0.835265568 0.603381027
## 77   0.147899709 0.002129533
## 78  -0.056345745 0.454066851
## 79  -0.116647728 0.514302555
## 80  -0.826554909 0.303212389
## 81  -0.458453234 0.980852609
## 82  -0.850696641 0.504879727
## 83  -0.213478931 0.285312673
## 84   0.150440021 0.759404889
## 85  -0.206555958 0.093845251
## 86   0.669001209 0.293732033
## 87   2.272002140 0.793346806
## 88   0.628200334 0.403941645
## 89  -0.002047917 0.967523502
## 90   0.818071319 0.112065395
## 91   0.194047800 0.872395358
## 92  -0.130964427 0.140082847
## 93  -1.217901761 0.529291230
## 94  -0.065248227 0.400714733
## 95   0.960811956 0.794007441
## 96  -1.124872603 0.672744813
## 97  -0.505648061 0.411487702
## 98   0.951289330 0.706302525
## 99  -0.759399565 0.612477406
## 100 -2.170500809 0.651304036
```

Connections

```
str(file)
```

```
## function (description = "", open = "", blocking = TRUE, encoding = getOption("encoding"),
##     raw = FALSE)
```

Subsetting

```
subset1 <- c("a", "b", "c", "c", "d", "a")
subset1[1:3]
```

```
## [1] "a" "b" "c"
```

```
subset1[c(1,3,5)]
```

```
## [1] "a" "c" "d"
```

```
subset1[subset1>"b"]
```

```
## [1] "c" "c" "d"
```

```r
subsetmat <- matrix(1:9,nrow=3,ncol=3)

subsetmat
```

```
##      [,1] [,2] [,3]
## [1,]    1    4    7
## [2,]    2    5    8
## [3,]    3    6    9
```

```r
subsetmat[2,2]
```

```
## [1] 5
```

```r
class(subsetmat[2,2])
```

```
## [1] "integer"
```

```r
subsetmat[2,2,drop=F]
```

```
##      [,1]
## [1,]    5
```

```r
class(subsetmat[2,2,drop=F])
```

```
## [1] "matrix"
```

```r
subsetmat[1,]
```

```
## [1] 1 4 7
```

```r
class(subsetmat[1,])
```

```
## [1] "integer"
```

```r
subsetmat[1, ,drop=F]
```

```
##      [,1] [,2] [,3]
## [1,]    1    4    7
```

```r
class(subsetmat[1, ,drop=F])
```

```
## [1] "matrix"
```

```r
subsetmat[,1]
```

```
## [1] 1 2 3
```

```r
class(subsetmat[,1])
```

```
## [1] "integer"
```

```r
subsetmat[ ,1,drop=F]
```

```
##      [,1]
## [1,]    1
## [2,]    2
## [3,]    3
```

```r
class(subsetmat[ ,1,drop=F])
```

```
## [1] "matrix"
```

```r
subsetlist <- list(foo = 1:4, bar = 0.7)
subsetlist
```

```
## $foo
## [1] 1 2 3 4
##
## $bar
## [1] 0.7
```

```r
subsetlist$foo
```

```
## [1] 1 2 3 4
```

```r
subsetlist[["foo"]]
```

```
## [1] 1 2 3 4
```

```r
#[[ ]] can be used with computed ones whereas $ can be used only with literal names
indices <- "foo"
subsetlist$indices
```

```
## NULL
```

```r
subsetlist[[indices]]
```

```
## [1] 1 2 3 4
```

```r
subsetnestedlist <- list(a = list(10, 12, 14), b = c(3.14, 2.81))
class(subsetnestedlist$a)
```

```
## [1] "list"
```

```r
subsetnestedlist[[c(1,3)]]
```

```
## [1] 14
```

```r
class(subsetnestedlist[[c(1,3)]])
```

```
## [1] "numeric"
```

```r
subsetnestedlist[[1]][[3]]
```

```
## [1] 14
```

```r
subsetnestedlist[[c(2,1)]]
```

```
## [1] 3.14
```

```r
class(subsetnestedlist[[2]][[1]])
```

```
## [1] "numeric"
```

```r
subsetnestedlist[[2]][[1]]
```

```
## [1] 3.14
```

```r
subsetnestedlist1 <- list(a = list(10, 12, 14), b = c(3.14, 2.81), d=c(T,F,T,T))

subsetnestedlist1[c(1,3)]
```

```
## $a
## $a[[1]]
## [1] 10
##
```

```
## $a[[2]]
## [1] 12
##
## $a[[3]]
## [1] 14
##
##
## $d
## [1]  TRUE FALSE  TRUE  TRUE
```

```
class(subsetnestedlist1[c(1,3)])
```

```
## [1] "list"
```

```
subsetnestedlist1[[c(1,3)]]
```

```
## [1] 14
```

Partial matching

```
pm <- list(aarthy=c(23,04,2000))
pm$a
```

```
## [1]   23    4 2000
```

```
pm[["a"]]
```

```
## NULL
```

```
pm[["a",exact=F]]
```

```
## [1]   23    4 2000
```

Removing NAN values

```
missnan <- c(1,2,NA,6,NA,4)
bad <- is.na(missnan)
bad
```

```
## [1] FALSE FALSE  TRUE FALSE  TRUE FALSE
```

```
missnan[!bad]
```

```
## [1] 1 2 6 4
```

```
#help("complete.cases")
#Return a logical vector indicating which cases are complete, i.e., have no missing values.
good <- complete.cases(missnan)
good
```

```
## [1]  TRUE  TRUE FALSE  TRUE FALSE  TRUE
```

```
missnan[good]
```

```
## [1] 1 2 6 4
```

```r
missnan1 <- c(1,2,NA,6,NA,4)
missnan2 <- c(NA,"b","a",NA,"c",NA)
#OR ing of 1 and NA results in FALSE, OR ing of 2 and "b" returns TRUE
good1 <- complete.cases(missnan1,missnan2)
good1
```

```
## [1] FALSE  TRUE FALSE FALSE FALSE FALSE
```

```r
missnan1[good1]
```

```
## [1] 2
```

```r
missnan2[good1]
```

```
## [1] "b"
```

```r
is.na(head(airquality))
```

```
##   Ozone Solar.R  Wind  Temp Month   Day
## 1 FALSE   FALSE FALSE FALSE FALSE FALSE
## 2 FALSE   FALSE FALSE FALSE FALSE FALSE
## 3 FALSE   FALSE FALSE FALSE FALSE FALSE
## 4 FALSE   FALSE FALSE FALSE FALSE FALSE
## 5  TRUE    TRUE FALSE FALSE FALSE FALSE
## 6 FALSE    TRUE FALSE FALSE FALSE FALSE
```

```r
complete.cases(head(airquality))
```

```
## [1]  TRUE  TRUE  TRUE  TRUE FALSE FALSE
```

```r
good2 <- complete.cases(airquality)
head(airquality[good2, ])
```

```
##   Ozone Solar.R Wind Temp Month Day
## 1    41     190  7.4   67     5   1
## 2    36     118  8.0   72     5   2
## 3    12     149 12.6   74     5   3
## 4    18     313 11.5   62     5   4
## 7    23     299  8.6   65     5   7
## 8    19      99 13.8   59     5   8
```

```r
head(airquality[!good2,])
```

```
##    Ozone Solar.R Wind Temp Month Day
## 5     NA      NA 14.3   56     5   5
## 6     28      NA 14.9   66     5   6
## 10    NA     194  8.6   69     5  10
## 11     7      NA  6.9   74     5  11
## 25    NA      66 16.6   57     5  25
## 26    NA     266 14.9   58     5  26
```

Vectorized

```r
v1 <- 1:4
v2 <- 5:8
v1
```

```
## [1] 1 2 3 4
```

```
v2
```

```
## [1] 5 6 7 8
```

```
v1+v2
```

```
## [1]  6  8 10 12
```

```
v1-v2
```

```
## [1] -4 -4 -4 -4
```

```
v1*v2
```

```
## [1]  5 12 21 32
```

```
v1/v2
```

```
## [1] 0.2000000 0.3333333 0.4285714 0.5000000
```

```
v1>2
```

```
## [1] FALSE FALSE  TRUE  TRUE
```

```
v2==8
```

```
## [1] FALSE FALSE FALSE  TRUE
```

```
mat1
```

```
##   10 11 12
## 1  1  4  7
## 2  2  5  8
## 3  3  6  9
```

```
mat2<- matrix(rep(10,9),3,3)
mat2
```

```
##      [,1] [,2] [,3]
## [1,]   10   10   10
## [2,]   10   10   10
## [3,]   10   10   10
```

```
mat1+mat2
```

```
##   10 11 12
## 1 11 14 17
## 2 12 15 18
## 3 13 16 19
```

```
mat1-mat2
```

```
##   10 11 12
## 1 -9 -6 -3
## 2 -8 -5 -2
## 3 -7 -4 -1
```

```
mat1*mat2
```

```
##   10 11 12
## 1 10 40 70
## 2 20 50 80
## 3 30 60 90
```

```
mat1/mat2
```

```
##    10  11  12
## 1 0.1 0.4 0.7
## 2 0.2 0.5 0.8
## 3 0.3 0.6 0.9
```

```
mat1%*%mat2
```

```
##   [,1] [,2] [,3]
## 1  120  120  120
## 2  150  150  150
## 3  180  180  180
```

Quiz

```
x <- 4L
x <- c(4,TRUE)
x
```

```
## [1] 4 1
```

```
class(x)
```

```
## [1] "numeric"
```

```
x<-c(1,3,5)
y<-c(3,2,10)
class(cbind(x,y))
```

```
## [1] "matrix"
```

```
x<- list(2, "a", "b", TRUE)
length(x[[2]])
```

```
## [1] 1
```

```
class(x[[2]])
```

```
## [1] "character"
```

```
x <- 1:4
y <- 2
x
```

```
## [1] 1 2 3 4
```

```
y
```

```
## [1] 2
```

```
x+y
```

```
## [1] 3 4 5 6
```

```
x <- c(3, 5, 1, 10, 12, 6)
x[x<6]<-0
x
```

```
## [1]  0  0  0 10 12  6
```

```r
x <- c(3, 5, 1, 10, 12, 6)
x[x<6]==0
```

```
## [1] FALSE FALSE FALSE
```

```r
x
```

```
## [1]  3  5  1 10 12  6
```

```r
x <- c(3, 5, 1, 10, 12, 6)
x[x>6]<-0
x
```

```
## [1] 3 5 1 0 0 6
```

```r
x <- c(3, 5, 1, 10, 12, 6)
x[x<=5]<-0
x
```

```
## [1]  0  0  0 10 12  6
```

```r
x <- c(3, 5, 1, 10, 12, 6)
x[x%in%1:5]<-0
x
```

```
## [1]  0  0  0 10 12  6
```

```r
x <- c(3, 5, 1, 10, 12, 6)
x[x!=6]<-0
x
```

```
## [1] 0 0 0 0 0 6
```

```r
x <- c(3, 5, 1, 10, 12, 6)
x[x==6]<-0
x
```

```
## [1]  3  5  1 10 12  0
```

```r
x <- c(3, 5, 1, 10, 12, 6)
x[x>=6]<-0
x
```

```
## [1] 3 5 1 0 0 0
```

```r
x <- c(3, 5, 1, 10, 12, 6)
x[x==0]<-6
x
```

```
## [1]  3  5  1 10 12  6
```

```r
x <- c(3, 5, 1, 10, 12, 6)
x[x==0]<-6
x
```

```
## [1]  3  5  1 10 12  6
```

11-20

```r
hw1 <- read.csv("hw1_data.csv",header=T)
tail(hw1)
```

```
##     Ozone Solar.R Wind Temp Month Day
## 148    14      20 16.6   63     9  25
## 149    30     193  6.9   70     9  26
## 150    NA     145 13.2   77     9  27
## 151    14     191 14.3   75     9  28
## 152    18     131  8.0   76     9  29
## 153    20     223 11.5   68     9  30
```

```r
hw1[47, ]
```

```
##    Ozone Solar.R Wind Temp Month Day
## 47    21     191 14.9   77     6  16
```

```r
bad <- is.na(hw1[["Ozone"]])
hw1[!complete.cases(hw1$Ozone), ]
```

```
##     Ozone Solar.R Wind Temp Month Day
## 5      NA      NA 14.3   56     5   5
## 10     NA     194  8.6   69     5  10
## 25     NA      66 16.6   57     5  25
## 26     NA     266 14.9   58     5  26
## 27     NA      NA  8.0   57     5  27
## 32     NA     286  8.6   78     6   1
## 33     NA     287  9.7   74     6   2
## 34     NA     242 16.1   67     6   3
## 35     NA     186  9.2   84     6   4
## 36     NA     220  8.6   85     6   5
## 37     NA     264 14.3   79     6   6
## 39     NA     273  6.9   87     6   8
## 42     NA     259 10.9   93     6  11
## 43     NA     250  9.2   92     6  12
## 45     NA     332 13.8   80     6  14
## 46     NA     322 11.5   79     6  15
## 52     NA     150  6.3   77     6  21
## 53     NA      59  1.7   76     6  22
## 54     NA      91  4.6   76     6  23
## 55     NA     250  6.3   76     6  24
## 56     NA     135  8.0   75     6  25
## 57     NA     127  8.0   78     6  26
## 58     NA      47 10.3   73     6  27
## 59     NA      98 11.5   80     6  28
## 60     NA      31 14.9   77     6  29
## 61     NA     138  8.0   83     6  30
## 65     NA     101 10.9   84     7   4
## 72     NA     139  8.6   82     7  11
## 75     NA     291 14.9   91     7  14
## 83     NA     258  9.7   81     7  22
## 84     NA     295 11.5   82     7  23
## 102    NA     222  8.6   92     8  10
## 103    NA     137 11.5   86     8  11
## 107    NA      64 11.5   79     8  15
## 115    NA     255 12.6   75     8  23
```

```
## 119    NA     153  5.7   88     8  27
## 150    NA     145 13.2   77     9  27
```

```
hw2<-hw1[complete.cases(hw1$Ozone), ]
mean(hw2$Ozone)
```

```
## [1] 42.12931
```

```
#hw3 <- hw1[hw1$Ozone>31 & hw1$Temp>90, ]
hw3 <- hw1[which(hw1$Ozone>31 & hw1$Temp>90), ]
hw3
```

```
##     Ozone Solar.R Wind Temp Month Day
## 69     97     267  6.3   92     7   8
## 70     97     272  5.7   92     7   9
## 120    76     203  9.7   97     8  28
## 121   118     225  2.3   94     8  29
## 122    84     237  6.3   96     8  30
## 123    85     188  6.3   94     8  31
## 124    96     167  6.9   91     9   1
## 125    78     197  5.1   92     9   2
## 126    73     183  2.8   93     9   3
## 127    91     189  4.6   93     9   4
```

```
mean(hw3$Solar.R)
```

```
## [1] 212.8
```

```
hw4<- hw1[which(hw1$Month==6), ]
mean(hw4$Temp)
```

```
## [1] 79.1
```

```
hw5<- hw1[which(hw1$Month==5), ]
max(hw5$Ozone,na.rm=T)
```

```
## [1] 115
```