Introduction to R Software

Data Frames

Shalabh

Department of Mathematics and Statistics

Indian Institute of Technology Kanpur

Data Frames

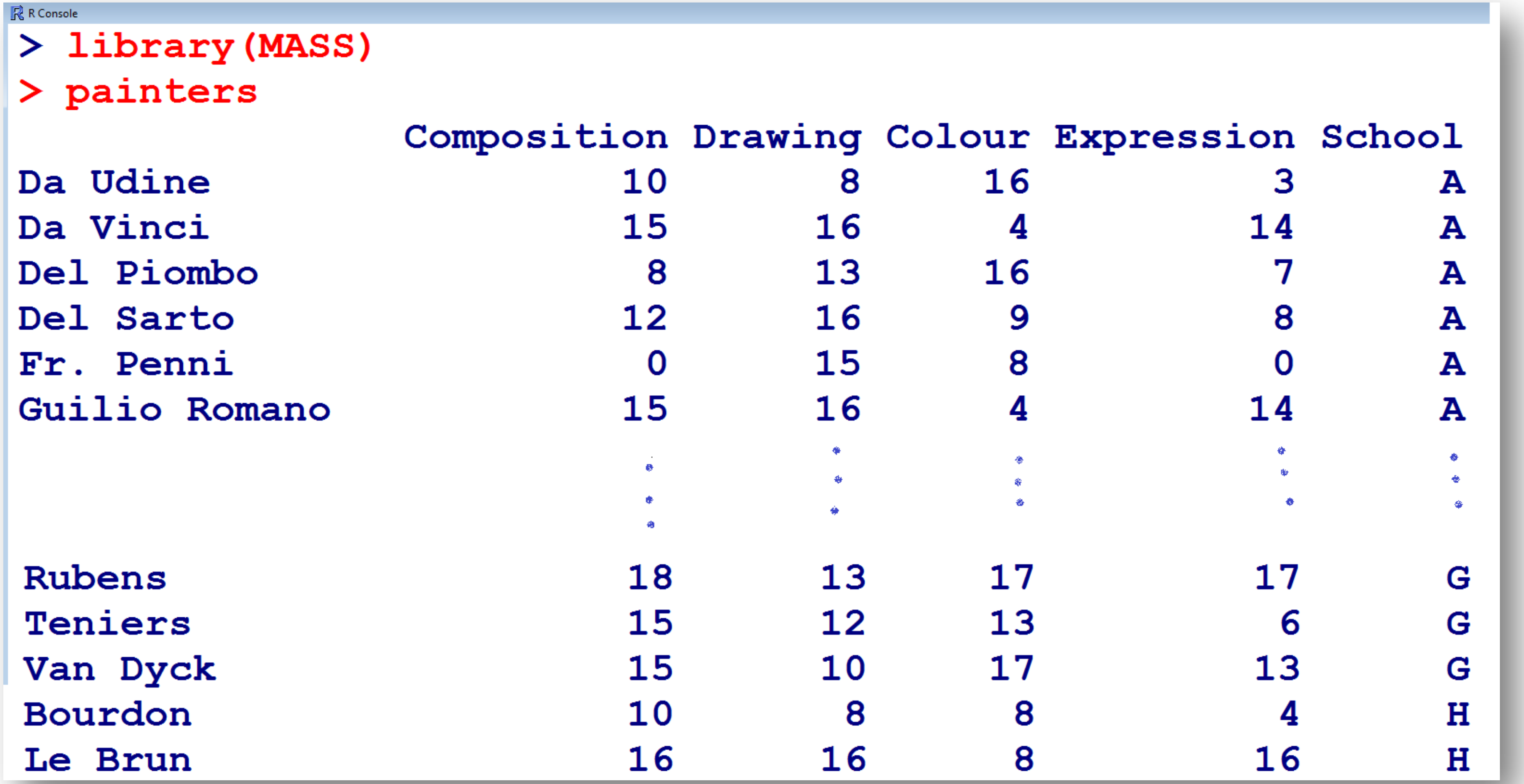
An example data frame `painters` is available in the library MASS (here only an excerpt of a data set):

```
> library(MASS)

> painters
          Composition Drawing Colour Expression School
Da Udine          10     8      16        3        A
Da Vinci          15    16       4       14        A
Del Piombo         8    13      16        7        A
Del Sarto         12    16       9        8        A
Fr. Penni          0    15       8        0        A
                   .     .       .        .        .
                   .     .       .        .        .
                   .     .       .        .        .
```

Here, the names of the painters serve as row identifications, i.e., every row is assigned to the name of the corresponding painter.

Data Frames

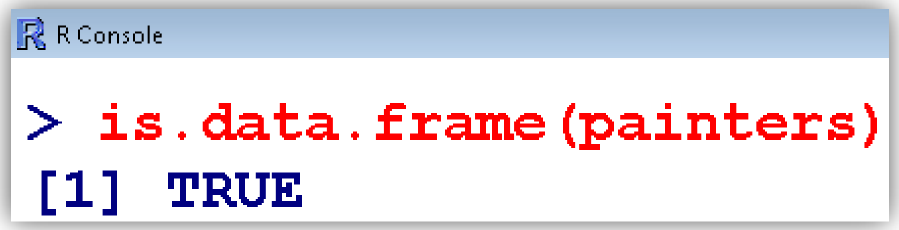


```
> library(MASS)
> painters
                Composition Drawing Colour Expression School
Da Udine                 10       8     16          3      A
Da Vinci                 15      16      4         14      A
Del Piombo                8      13     16          7      A
Del Sarto                12      16      9          8      A
Fr. Penni                 0      15      8          0      A
Guilio Romano            15      16      4         14      A
                          .       .      .          .      .
                          .       .      .          .      .
                          .       .      .          .      .
Rubens                   18      13     17         17      G
Teniers                  15      12     13          6      G
Van Dyck                 15      10     17         13      G
Bourdon                  10       8      8          4      H
Le Brun                  16      16      8         16      H
```

Data Frames

- Test if we are dealing with a data frame:

```
> is.data.frame(painters)
[1] TRUE
```



```
> is.data.frame(painters)
[1] TRUE
```

Data Frames

❑ **Creating Data Frames**

Use the `data.frame` function to create a data frame by adding column vectors to the data frame.

Example:

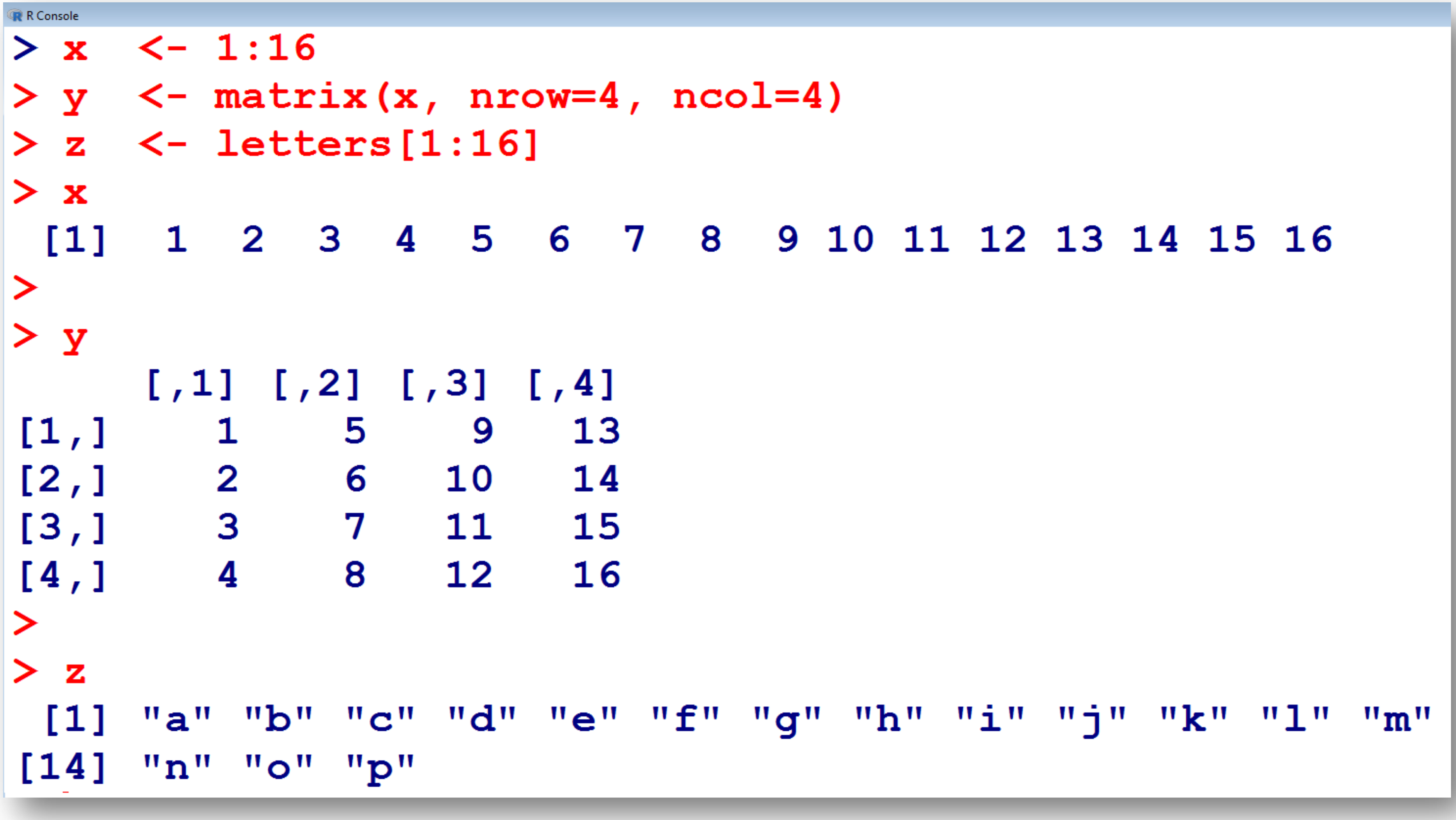
```
> x  <- 1:16                          # Vector
> y  <- matrix(x, nrow=4, ncol=4)     # 4 X 4 matrix
> z  <- letters[1:16]                 # lowercase alphabets

> x
 [1]  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16
> y
     [,1] [,2] [,3] [,4]
[1,]    1    5    9   13
[2,]    2    6   10   14
[3,]    3    7   11   15
[4,]    4    8   12   16
> z
 [1] "a" "b" "c" "d" "e" "f" "g" "h" "i" "j" "k" "l" "m"
"n" "o" "p"
```

Data Frames

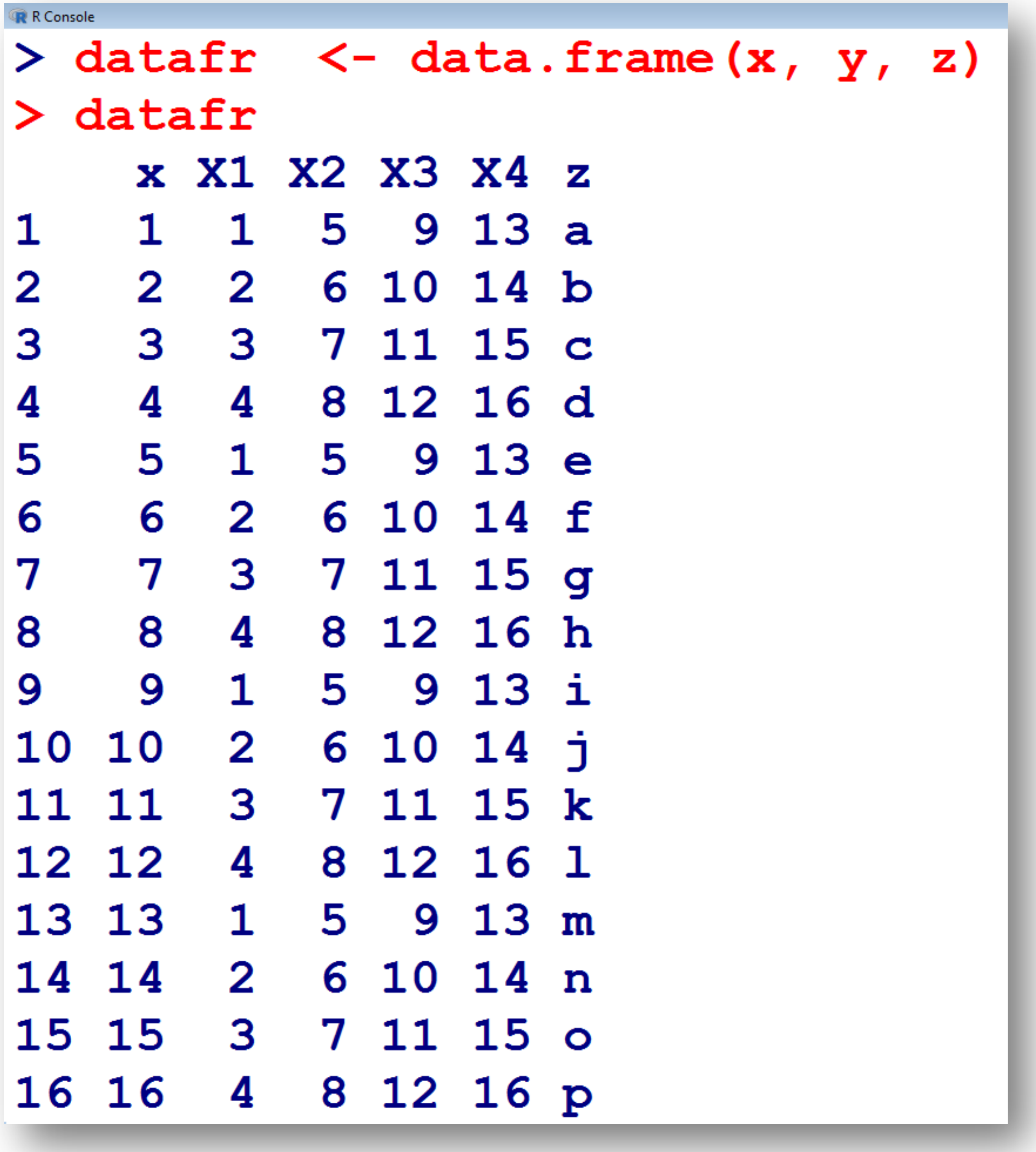
```
> datafr  <- data.frame(x, y, z)
> datafr
    x X1 X2 X3 X4 z
1   1  1  5  9 13 a
2   2  2  6 10 14 b
3   3  3  7 11 15 c
4   4  4  8 12 16 d
5   5  1  5  9 13 e
6   6  2  6 10 14 f
7   7  3  7 11 15 g
8   8  4  8 12 16 h
9   9  1  5  9 13 i
10 10  2  6 10 14 j
11 11  3  7 11 15 k
12 12  4  8 12 16 l
13 13  1  5  9 13 m
14 14  2  6 10 14 n
15 15  3  7 11 15 o
16 16  4  8 12 16 p
```

Data Frames



```
> x  <- 1:16
> y  <- matrix(x, nrow=4, ncol=4)
> z  <- letters[1:16]
> x
 [1]  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16
>
> y
     [,1] [,2] [,3] [,4]
[1,]    1    5    9   13
[2,]    2    6   10   14
[3,]    3    7   11   15
[4,]    4    8   12   16
>
> z
 [1] "a" "b" "c" "d" "e" "f" "g" "h" "i" "j" "k" "l" "m"
[14] "n" "o" "p"
```

Data Frames



```
> datafr  <- data.frame(x, y, z)
> datafr
    x X1 X2 X3 X4 z
1   1  1  5  9 13 a
2   2  2  6 10 14 b
3   3  3  7 11 15 c
4   4  4  8 12 16 d
5   5  1  5  9 13 e
6   6  2  6 10 14 f
7   7  3  7 11 15 g
8   8  4  8 12 16 h
9   9  1  5  9 13 i
10 10  2  6 10 14 j
11 11  3  7 11 15 k
12 12  4  8 12 16 l
13 13  1  5  9 13 m
14 14  2  6 10 14 n
15 15  3  7 11 15 o
16 16  4  8 12 16 p
```

Data Frames

❑ **Structure of the data:**

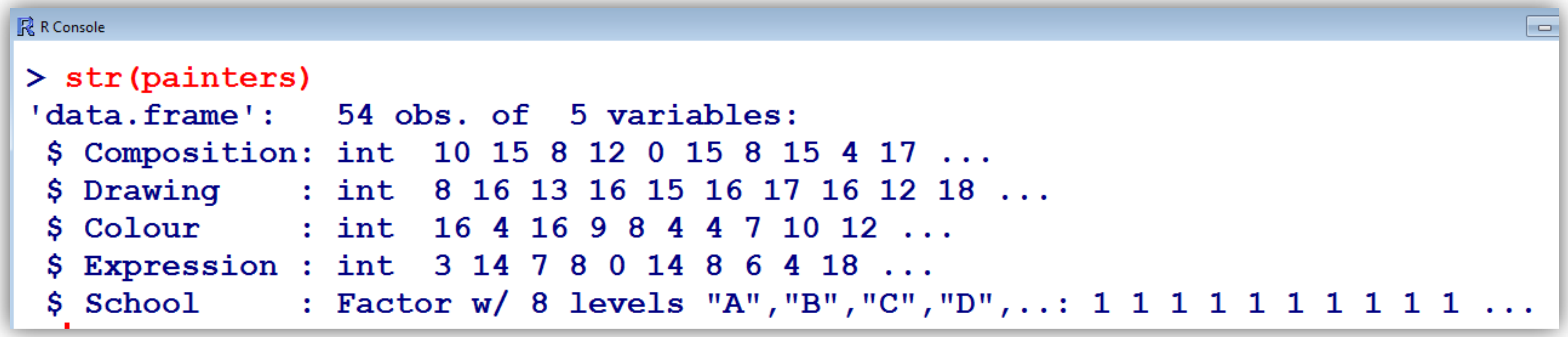
Display information about the structure of the data frame (`str`).

The result of `str` gives the dimension as well as the name and type of each variable.

```
> str(painters)
'data.frame'  :   54 obs. of  5 variables:
 $ Composition: int  10 15 8 12 0 15 8 15 4 17 ...
 $ Drawing    : int  8 16 13 16 15 16 17 16 12 18 ...
 $ Colour     : int  16 4 16 9 8 4 4 7 10 12 ...
 $ Expression : int  3 14 7 8 0 14 8 6 4 18 ...
 $ School     : Factor w/ 8 levels "A","B","C","D",..: 1
                                  1 1 1 1 1 1 1 1 1 ...
```

`int` means integer.

Data Frames



```
> str(painters)
'data.frame':   54 obs. of  5 variables:
 $ Composition: int  10 15 8 12 0 15 8 15 4 17 ...
 $ Drawing    : int  8 16 13 16 15 16 17 16 12 18 ...
 $ Colour     : int  16 4 16 9 8 4 4 7 10 12 ...
 $ Expression : int  3 14 7 8 0 14 8 6 4 18 ...
 $ School     : Factor w/ 8 levels "A","B","C","D",..: 1 1 1 1 1 1 1 1 1 1 ...
```

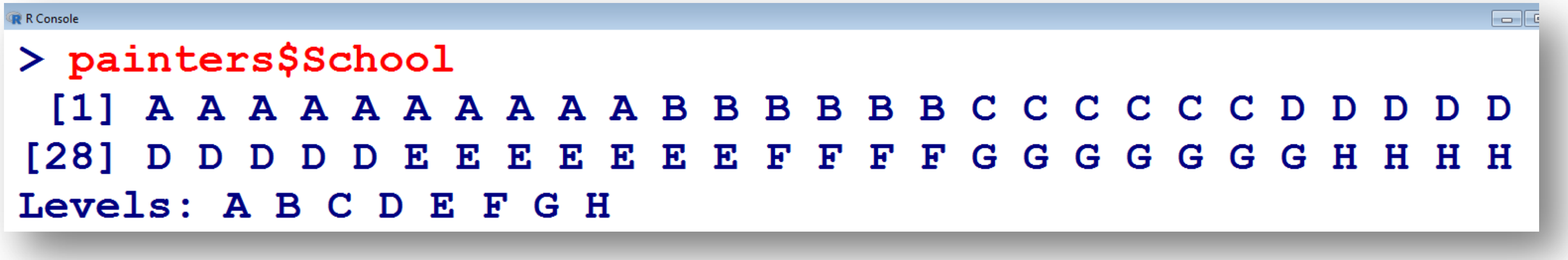
Data Frames

❑ Extract a variable from data frame using $

Variables can be extracted using the $ operator followed by the name of the variable.

Example: Suppose we want to extract information on variable `School` from the data set `painters.`

```
painters$School
 [1] A A A A A A A A A A B B B B B B C C C C C C D D D D D
[28] D D D D D E E E E E E E F F F F G G G G G G G H H H H
Levels: A B C D E F G H
```



```
> painters$School
 [1] A A A A A A A A A A B B B B B B C C C C C C D D D D D
[28] D D D D D E E E E E E E F F F F G G G G G G G H H H H
Levels: A B C D E F G H
```

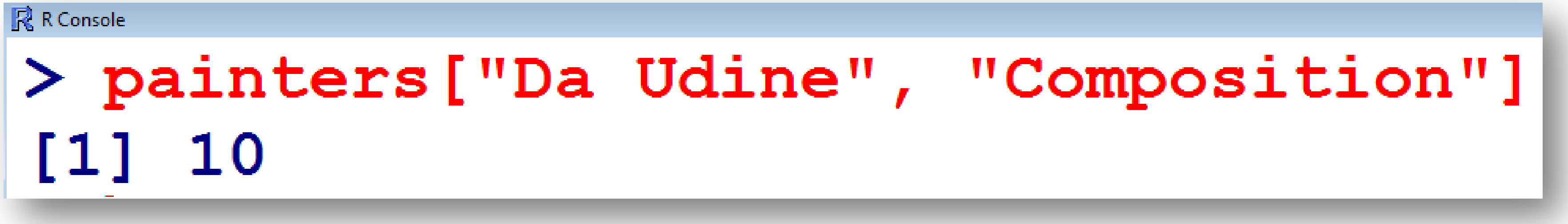
Data Frames

- ❑ Extract data from a data frame

The data from a data frame can be extracted by using the matrix-style `[row, column]` indexing.

Example: Suppose we want to extract information on the first painter `Da Udine` on the variable `Composition` from the data set `painters.`

```
> painters["Da Udine", "Composition"]
[1] 10
```

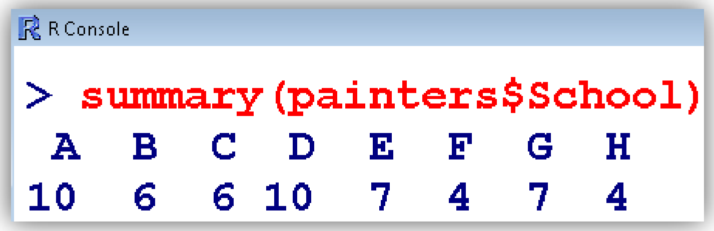


```
> painters["Da Udine", "Composition"]
[1] 10
```

Data Frames

The `summary` function for a categorical variable returns a detailed frequency table:

```
> summary(painters$School)
 A  B  C  D  E  F  G  H
10  6  6 10  7  4  7  4
```



```
> summary(painters$School)
 A  B  C  D  E  F  G  H
10  6  6 10  7  4  7  4
```

We will learn later:

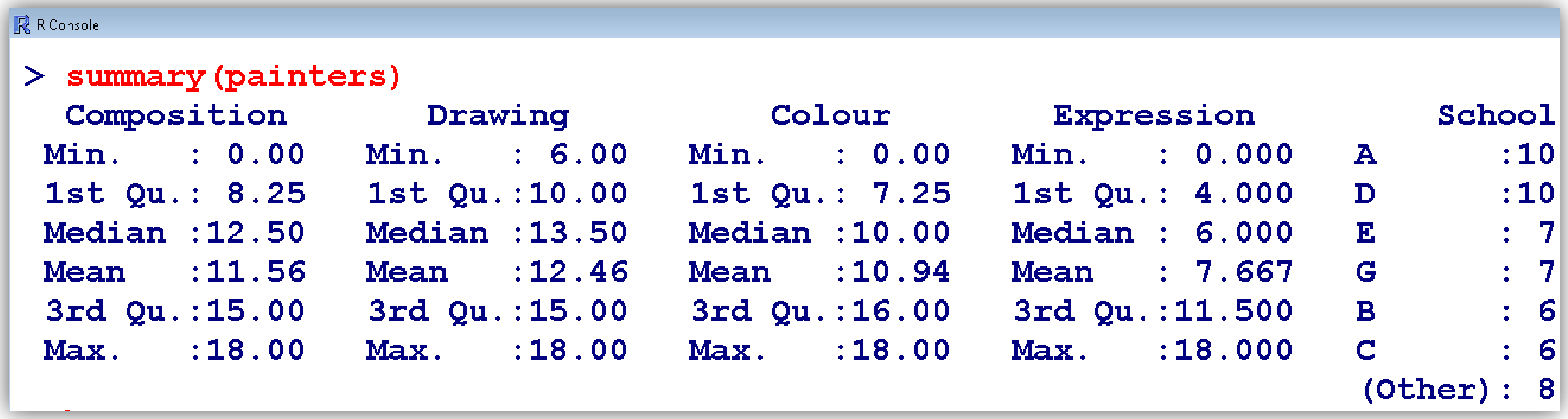
`summary` is a generic function used to produce result summaries of the results of various model fitting functions.

Data Frames

The `summary` function for a numeric variable returns an overview of descriptive measures for each variable: (*We will learn later*).

```
> summary(painters)
  Composition       Drawing          Colour        Expression      School
 Min.   : 0.00   Min.   : 6.00   Min.   : 0.00   Min.   : 0.000   A      :10
 1st Qu.: 8.25   1st Qu.:10.00   1st Qu.: 7.25   1st Qu.: 4.000   D      :10
 Median :12.50   Median :13.50   Median :10.00   Median : 6.000   E      : 7
 Mean   :11.56   Mean   :12.46   Mean   :10.94   Mean   : 7.667   G      : 7
 3rd Qu.:15.00   3rd Qu.:15.00   3rd Qu.:16.00   3rd Qu.:11.500   B      : 6
 Max.   :18.00   Max.   :18.00   Max.   :18.00   Max.   :18.000   C      : 6
                                                                  (Other): 8
```

Data Frames

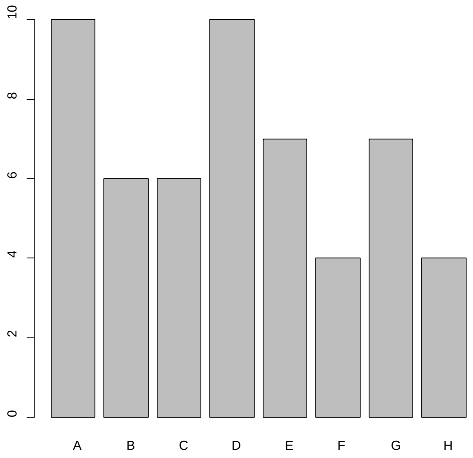


```
> summary(painters)
  Composition       Drawing          Colour        Expression         School 
 Min.   : 0.00   Min.   : 6.00   Min.   : 0.00   Min.   : 0.000   A      :10  
 1st Qu.: 8.25   1st Qu.:10.00   1st Qu.: 7.25   1st Qu.: 4.000   D      :10  
 Median :12.50   Median :13.50   Median :10.00   Median : 6.000   E      : 7  
 Mean   :11.56   Mean   :12.46   Mean   :10.94   Mean   : 7.667   G      : 7  
 3rd Qu.:15.00   3rd Qu.:15.00   3rd Qu.:16.00   3rd Qu.:11.500   B      : 6  
 Max.   :18.00   Max.   :18.00   Max.   :18.00   Max.   :18.000   C      : 6  
                                                                  (Other): 8  
```

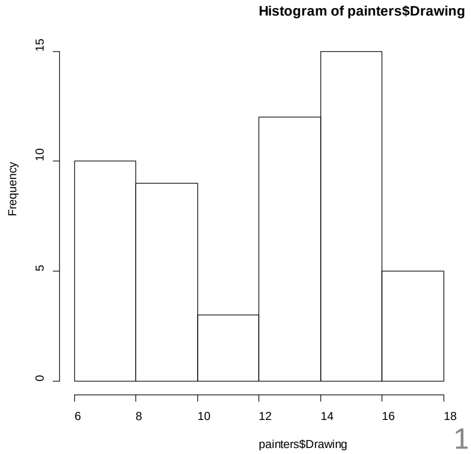
Data Frames

❑ Plot and graphics of the data

```
> plot(painters$School) #factor variable
```



```
> hist(painters$Drawing) #numeric variable
```


Histogram of painters$Drawing
Frequency
painters$Drawing