

Foundations of R Software
Prof. Shalabh
Department of Mathematics and Statistics
Indian Institute of Technology, Kanpur

Lecture - 42
Data frames: Creations and Operation

Hello friend, welcome to the course foundations of R software and you can recall that in the last lecture we initiated a discussion on the Data frame. So, on this topic we had learned that what is this with data frame. And, in short means I can say, the type of file that you try to create from Microsoft Excel and which you popularly call as MS Excel or a Spreadsheet. This is equivalent to the concept of data frame in R software.

So, now as you know that in a spreadsheet you try to make different types of operations and so, our objective is that we want to learn that how we can make the same type of operation in the R software also. So, in this lecture today, we are going to continue on the topic of data frame and first we are trying to learn that how you can create a data frame and then how you can do different types of operations on it, right.

And, as, we have used the package MASS in the last lecture and we had used the data set painters in the last lecture. So, we are going to continue to use the same and the same data set in this lecture also for illustrating different type of operations.

(Refer Slide Time: 01:47)

Data Frames

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

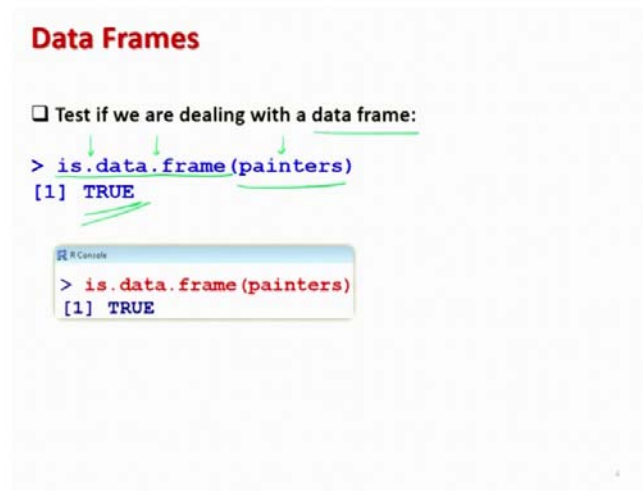
```
> library(MASS)
> painters
```

	Composition	Drawing	Colour	Expression	School
Da Udine	10	8	16	3	A
Da Vinci	15	16	4	14	A
Del Piombc	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.

So, let us begin our lecture, ok. So, if you can recall that we had used the library mass to upload the package mass MA double S and after that we have used the data set painters, which had different types of column names like as composition, drawing, colour expression, a school and there were different row names which are the names of different painters, right.

(Refer Slide Time: 02:27)



So, this is only a small piece excepts of that data set. And, it will look like this when you try to look it on the R console, ok. Now, the first question comes here that whenever you are going to get a file from some external sources you do not know is it data frame or something else, right.

So, if you want to test that whether the file which you have obtained is a data frame or not. So, for that we use the command here is dot d a t a dot f r a m e, right. So, you can see here that, there are two dots here or two full stops here, one after is and another after theta. So, is this is dot theta dot frame, right. So, it is equivalent to like as other commands which you have used in the past like is dot numeric is dot character etc.

So, in order to use it you have to write this command and inside the parentheses you have to write down the name. So, for example, we want to test here whether this data set painters is a data frame or not, so I try to write down here p a i n t e r s all in lower case alphabets and it comes out to be here true, right.

So, this means, this is a theta set which is in the format of data frame, right. Now, before I move forward in order to explain about different functions related to the data frame the first question comes here, how you can create a data frame?

(Refer Slide Time: 03:41)

```
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"
```

For example, here I am taking the example which is based on the built-in data frame, but suppose you have got different data set and if you want to create data frame, how you can do it?, right. So, for that we have a command here `data.frame`. So, this is a function which is used to create the data frame by adding column vectors to the data frame. So, suppose if I try to take here 3 different types of objects.

Say, I try to take here the numbers 1 to 16 and I store them in the variable x then I try to take here a matrix based on this 16 values and I try to arrange them in the 4 by 4 matrix, in which there are 4 rows and 4 columns and this is here y and then I try to take the alphabets, lower case alphabets from a to up to here p which are indexed from 1 to 16 and I try to store this under the variable name here z. So, you can see here x is like here like this 1 to 16 numbers.

(Refer Slide Time: 05:09)

```
Data Frames  
> datafr = data.frame(x, y, z)  
> datafr
```

x	x1	x2	x3	x4	z
1	1	5	9	13	a
2	2	6	10	14	b
3	3	7	11	15	c
4	4	8	12	16	d
5	1	5	9	13	e
6	2	6	10	14	f
7	3	7	11	15	g
8	4	8	12	16	h
9	1	5	9	13	i
10	2	6	10	14	j
11	3	7	11	15	k
12	4	8	12	16	l
13	1	5	9	13	m
14	2	6	10	14	n
15	3	7	11	15	o
16	4	8	12	16	p

Then y here is this matrix and z here is the 16 alphabets. And, now, I try to create here a data frame of x, y, z. So, what you have to see is that, I write down here data.frame and inside the parentheses I try to write down all the variables separated by comma, right. So, now you can observe what is really happening. So, this is your x and this is your y matrix which is your y and this is your z.

So, if you try to see here x, it is the set of numbers from 1 to 16 and this is exactly the same what you have given here, right. Similarly, if you try to see here this z, this is here the set of lower-case alphabets from a to p and this is the same set which we have taken here under z.

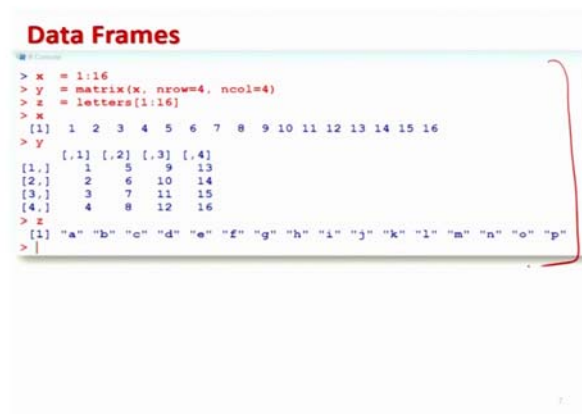
Now, if you try to look here at this matrix here y. So, this is here like this in which you can see the numbers from 1 to 16 are arranged column wise, from 1 2 3 4 and then it goes here, then 5 6 7 8, then come here then 9 10 11 12 and then come here then 13 14 15 16.

Now, if you try to see what is happening in this data frame? So, in this data frame, if you try to see here first, I am trying to make here a block; this is here your matrix y. But, after this if you try to see this matrix is repeated here 4 times, because in the data frame the number of data values for each of the column remain the same. So, what it has done that there were 4 observations in a row 1 2 3 4.

And then, in another column there are 16 observations. So, it has repeated the matrix y 4 times here, right. So, that is what is happening here, ok. So, that is what you have to be watchful when you are trying to create different data frames, right.

Because, in data frame there is a condition that the number of elements in every column should be the same and even the number of elements in the rows for different row names could be the same. Otherwise, R will try to adjust it automatically and the way it is trying to adjust that is the thing which we have to understand, right.

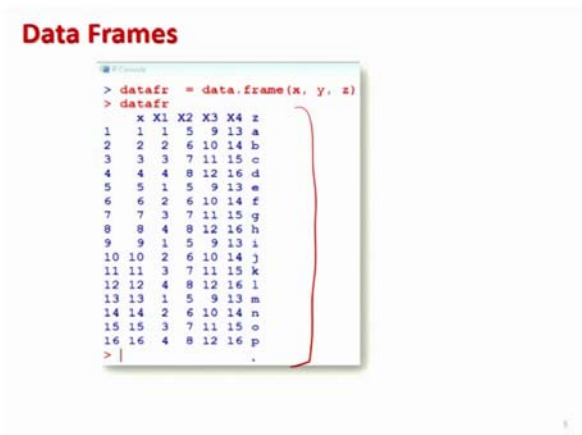
(Refer Slide Time: 07:57)



```
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" "n" "o" "p"
> |
```

So, you have to be extremely careful when you are trying to create that data frame. So, you can see here this is the screenshot of the same outcome.

(Refer Slide Time: 08:01)



```
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
> |
```

And this is here the data frame that we have just created, right. So, let me try to show you these operations first on the R software and then I try to move forward, right. So, I try to create here this, 3 variables here. So, you can see here x is here like this, y here is like this and z here is like this, right.

(Refer Slide Time: 08:23)

```

RGui (64-bit)
File Edit View Misc Packages Windows Help
R Console
> 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"
[16] "p"
> |
  
```

And then, I try to create here a data frame and I try to store it in a name d a t a f r. So, data frame and then I try to write down here x, y and here z, right. And, then if you try to see, what is this value? It is here like this. So, that is exactly the same thing which I shown you on this screenshot also, right here, ok.

(Refer Slide Time: 08:59)

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 ...
  
```

int means integer.

So, now after this, I come to another aspect and we try to consider some more operation which can be done on our data frame.

So, for that, I am just going to use data set on painters. So, first question I try to address here is that if that, how can you see the structure of the data?, right. The structure of the data frame. For example, if you want to know what type of variables are there? How many observations are there, etc., in a given data frame so the command here is str. str is the short form structure. So, this will become here str and inside the parentheses you have to write here painters p a i n t e r s and you will get here this type of output.

If you try to see what is telling you here, the first line is indicating that this painters is a data frame and it has 54 observations on 5 variables, right. And then, it is trying to give you the name of those variables which are here composition drawing, colour, expression and a school. And then, after that it is trying to give you the type of observation these variables are having. For example, it is written here i n t. So, i n t means here integer, right.

So, you can see here the data on the composition is integer, the data on drawing, colour, expression is integer, but the data on the school is factor. And, if you try to see this is exactly what we had learnt in the last lecture also, right. And after that, it is trying to give you briefly the some values of the data. So, that you can understand that what type of data is available, right. So, this command str will give you the structure of the data frame.

(Refer Slide Time: 10:53)

```
> 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 ...
```

And if you can see here this is the screenshot of the same operation, ok.

(Refer Slide Time: 10:59)

Data Frames

❑ Extract a variable from data frame using \$

Variables can be extracted using the \$ operator followed by the name of the variable. *name of the dataframe \$ Variable name*

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 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 H H H H
Levels: A B C D E F G H
```

After this, if you want to extract a variable from a data set how to get it done? And, if you try to recall that we had very briefly discussed this operation in the last in lecture also, but here I would now like to give you a formal explanation. So, the rule is very simple; whenever you want to extract a particular variable from a data frame, you have to write down the name of the data frame.

And then, you have to write down here the sign, colour operator and after that you have to write down the variable name. That is all, as simple as that. For example, if you want to extract the variable say school from the data set or data frame painters, you have to simply write down here the name of the data frame p a i n t e r s and then dollar operator and then a school. And then, yeah, you have to keep in mind that this variable name has to be exactly in the same way as it is mentioned, right.

So, if you really want to know what is the exact spelling, best is to first use the operator column names and c o l m n a m e s to find out the column names and then try to use these names over here, right. So, you can see here these are the values that are stored in this variable and this is here the screenshot of the same operation, ok.

(Refer Slide Time: 12:39)

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
```

And, now, in case if you want to extract a particular data set from a data frame, then how to get it done?

So, this operation is something like as you have done a similar operation in the matrix, that when you wanted to access a particular element in the matrix, a particular value which is located at a particular row and a particular column inside the matrix, right. So, in the. So, similarly in the case of data frame also you have to write down the square brackets and within the square brackets you have to write the address of the row and the address of the column. And which are separated by comma, right.

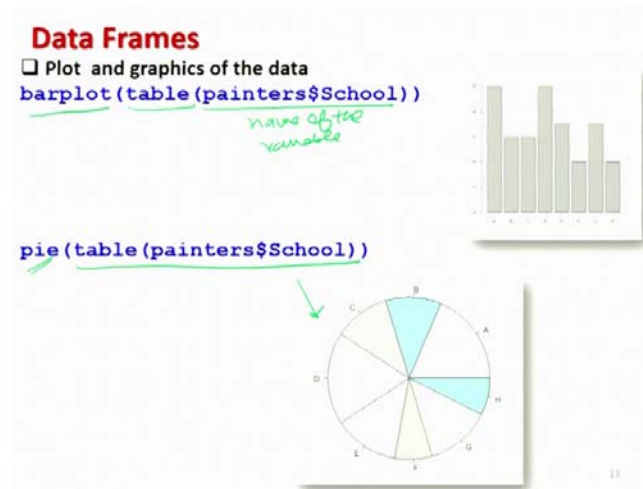
And then, you try to enter. So, it will try to give you the value which is available at that particular row and particular columns intersection, right. So, suppose you want to extract the information on the first painter which is here Da Udine and you want to see the value of the composition. So, if you try to recall this Da Udine, it is in the row and composition is here in the column.

So, if you try to write down here the name of the data frame as `painters` and then, inside the square brackets try to write down this row name here. So, now, this is here a name not a number. So, you have to write down the name in the double quotes `Da Udine` and yeah, that is the same name which is given in the data set. And then you have to write down the name of the column which is here `composition`.

And if you try to see here, it will come out of here like this 10. And this, you can see here this is the screenshot and if you really want to see you can see here in this one, this was the value here if you try to see like this. So, this was here like this. So, this is here the row name Da Udine and then it is here composition column and its value here is 10. So, that is exactly I have shown you here.

But without going into the screenshot, I have shown you with the R commands, right.

(Refer Slide Time: 15:01)



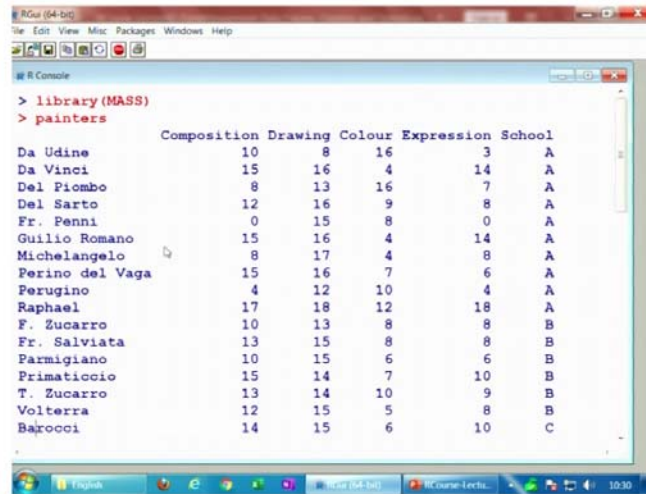
And, the advantage of finding out these variable is that, you can use them just like any other variable. For example, if you want to create a plots and graphic on such variable for example, if you simply want to create a bar plot, although we have not discussed it we will try to discuss different types of graphics soon in the forthcoming lectures.

But this is the command here that you have to write down here the command table t a b l e and inside the parentheses you have to write the name of the variable. So, now, you can see here the name of the variable comes out to be like this painters dollar a school and then you can create the bar plot here like this. And similarly, if you want to create a pie chart for that we will try to discuss it in more detail in the forthcoming lecture.

But you can use here the command here pie and then the same table and the variable name. So, you can see here you get the pie chart also, right. So, that is the advantage of

using such names, but now before going into more detail, let me try to first show you these operations on the R console, right.

(Refer Slide Time: 16:15)



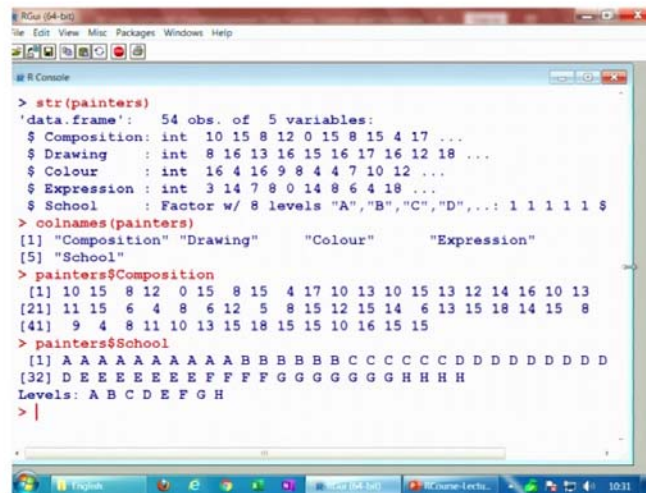
```
R Console
> library(MASS)
> painters

  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
Giulio Romano 15     16      4         14     A
Michelangelo   8     17      4          8     A
Perino del Vaga 15     16      7          6     A
Perugino       4     12     10          4     A
Raphael       17     18     12         18     A
F. Zucarro    10     13      8          8     B
Fr. Salviata  13     15      8          8     B
Parmigiano    10     15      6          6     B
Primaticcio   15     14      7         10     B
T. Zucarro    13     14     10          9     B
Volterra      12     15      5          8     B
Barrocci      14     15      6         10     C
```

So, first of all you have to upload here the package. So, I try to use here the library mass.

And then, I if you try to see here the painter here is like this, right. So, you can see here this is the name of the Da Udine and this is here the name of the variable composition and its value here is 10, which I just shown you, right.

(Refer Slide Time: 16:39)



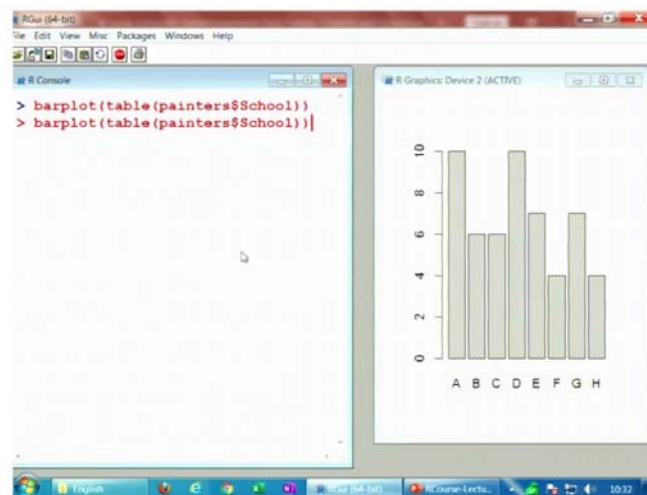
```
R Console
> str(painters)
'data.frame': 54 obs. of 5 variables:
 $ Composition: int 10 15 8 12 0 15 8 15 4 17 ...
 $ Drawing    : int 8 16 13 16 15 16 17 16 12 18 ...
 $ Colour     : int 16 4 16 9 8 4 4 7 10 12 ...
 $ Expression : int 3 14 7 8 0 14 8 6 4 18 ...
 $ School     : Factor w/ 8 levels "A","B","C","D",...: 1 1 1 1 1 1 1 1 1 1 $
> colnames(painters)
[1] "Composition" "Drawing" "Colour" "Expression"
[5] "School"
> painters$Composition
 [1] 10 15 8 12 0 15 8 15 4 17 10 13 10 15 13 12 14 16 10 13
 [21] 11 15 6 4 8 6 12 5 8 15 12 15 14 6 13 15 18 14 15 8
 [41] 9 4 8 11 10 13 15 18 15 15 10 16 15 15
> painters$School
 [1] A A A A A A A A A B B B B B B C C C C C D D D D D D D D
 [32] 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
> |
```

But, suppose if you do not know what is this data frame and if you want to know the structure of this painters. So, you can see here this will give you this is the data frame which has got 54 observations on 5 variables and so on and it has all such information, right.

And similarly, after that if you want to extract here a school. So, as I said the best approach is that you try to first see the column names, otherwise, we can always make a mistake, right. So, now, if you want to see the, what is the data on any variable what you have to do here, you simply have to write down the name of the this data frame which is here painter.

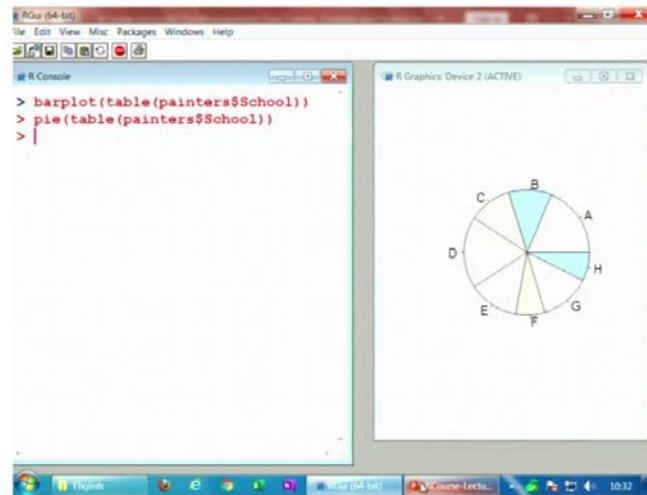
And then, you can write down here, say here composition if I try to see here, right. So, composition here you can see here like this, right. And, similarly if you try to see here for a school if you simply write down here painters and your school you get here like this, right.

(Refer Slide Time: 17:55)



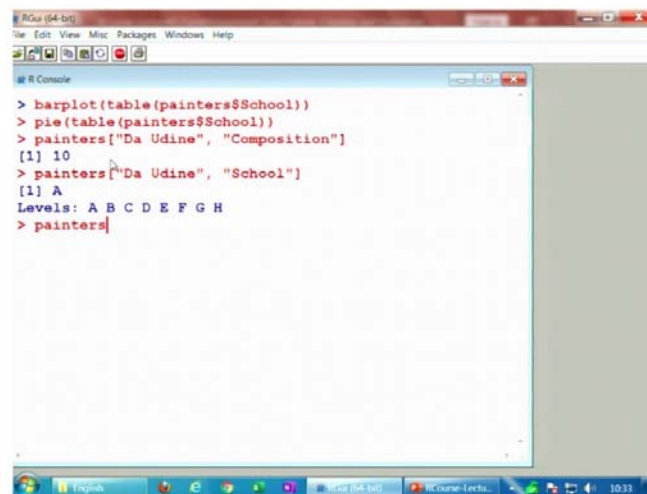
And similarly, if you want to create here, suppose here some diagrams on it. So, I can use here the see here this command here bar plot, on here schools. So, this will be table and then here this bar plot and if you try to see here you get here this type of curve, right. And similarly, if you try to use here the pie chart.

(Refer Slide Time: 18:17)



So, we can see here you will get here this type of graphic. So, you can see here that doing such operation is not very difficult. And similarly, if you want to obtain any particular observation from here also.

(Refer Slide Time: 18:35)



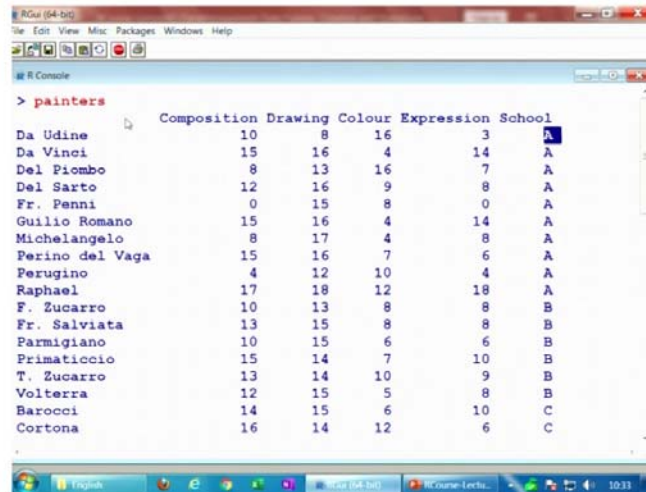
The screenshot shows an R console window with the following commands and output:

```
> barplot(table painters$School))
> pie(table painters$School))
> painters["Da Udine", "Composition"]
[1] 10
> painters["Da Udine", "School"]
[1] A
Levels: A B C D E F G H
> painters|
```

So, you can see here, suppose I want to have the data on the Da Udine and composition, I want to know the value it is here like this. Similarly, if you want to know what is the value of a school, you can see here this will come out to be here like this, right. And if

you want to see here this value, you can see here you can see here this is here the value of the school here in Da Udine and the composition value here is 2.

(Refer Slide Time: 19:13)

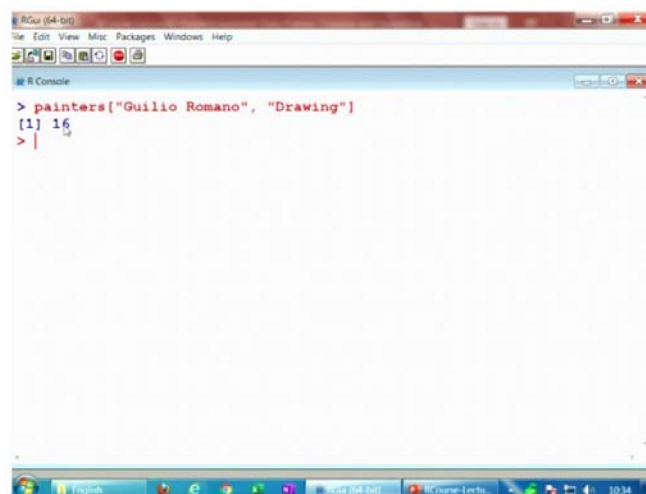


```
> painters
```

	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
Michelangelo	8	17	4	8	A
Perino del Vaga	15	16	7	6	A
Perugino	4	12	10	4	A
Raphael	17	18	12	18	A
F. Zucarro	10	13	8	8	B
Fr. Salviata	13	15	8	8	B
Parmigiano	10	15	6	6	B
Primaticcio	15	14	7	10	B
T. Zucarro	13	14	10	9	B
Volterra	12	15	5	8	B
Barocci	14	15	6	10	C
Cortona	16	14	12	6	C

Similarly, if you want to know this about this painter Guilio Romano and if you want to know the value of his drawing. So, this is here you can see here 16. So, I try to use here.

(Refer Slide Time: 19:35)

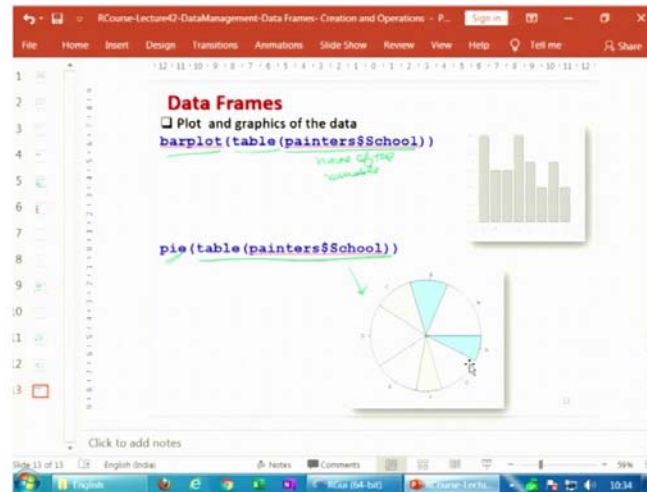


```
> painters["Guilio Romano", "Drawing"]  
[1] 16  
> |
```

And I try to write down here the name and then I try to write down here the drawing. So, you can see here this value comes out of here 16 as such what you have obtained there, right.

So, you can see here that obtaining this type of information is not very difficult.

(Refer Slide Time: 19:59)



And similarly, you can do different type of say this calculations on this extracted variables also, right. So, now, we come to an end to this lecture and you can see that this was also a short lecture and I have taken very small number of commands here; because my objective is that, I want to give you some time. So, that you can settle down these concepts in your mind.

And you can have some time to practice these commands. So, gradually I will try to build up more commands in the next lecture. So, I would request you that, why do not you take up any data set which you have created in any in the format of any spreadsheet or in your MS Excel and try to see how you can get the same information in the R software also, right. So, you try to practice and I will see you in the next lecture till then, goodbye.