Essentials of Data Science with R Software - 2
Sampling Theory and Linear Regression Analysis
Prof. Shalabh
Department of Mathematics and Statistics
Indian Institute of Technology, Kanpur

Lecture - 02
Installation and Working with R

Hello, welcome to the course Essentials of Data Science with R Software 2, where we are going to talk about the topics of Sampling Theory and Regression Analysis. And in this module of the course, we are going to talk about some basic commands to the R software.

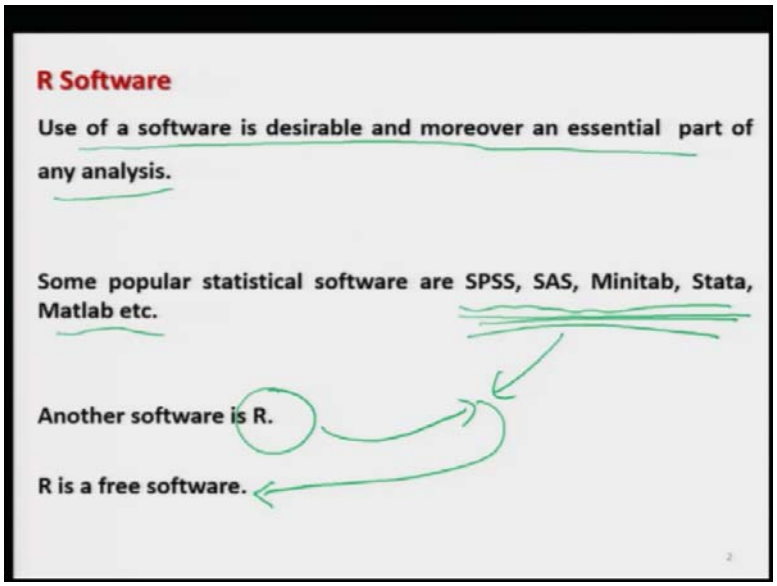
In the last lecture, I had tried to give you some overview that how statistics is related to the data science. So, we had discussed different aspects, statistics, mathematics, computer science, software, etc. So, definitely when we are trying to talk about data science, we need a software also.

So, in this course, I am going to use the software whose name is R, capital R. Means I am sure that you have heard about this software. Now, in the next couple of lectures, I will try to give you a concise brief information on the tools which I am going to use in R. And I would like to make it very clear in the beginning itself that it is not possible for me to teach you complete R software in this lecture.

I am assuming that you have a fair background in the R software, and you know how to operate it, you know about the basic commands. Now, in case if you do not know, then one of the suggestion I can give you is that I have another course on this R software called as Introduction to R software. And it is video files are uploaded on the websites of NPTEL. And possibly you can download them from there and can have a look or you can follow from any other sources any other book, any other video, any other notes etc. whatever you want.

So, now I will try to start, in this lecture I will try to give you a brief background about R and that the choice of commands which I have chosen in my lecture they are essentially related to the topics which I am going to cover in this course. So, I will try my best to make sure that whatever tools in R, I am going to use further those tools are explained here in the next couple of lectures, right.

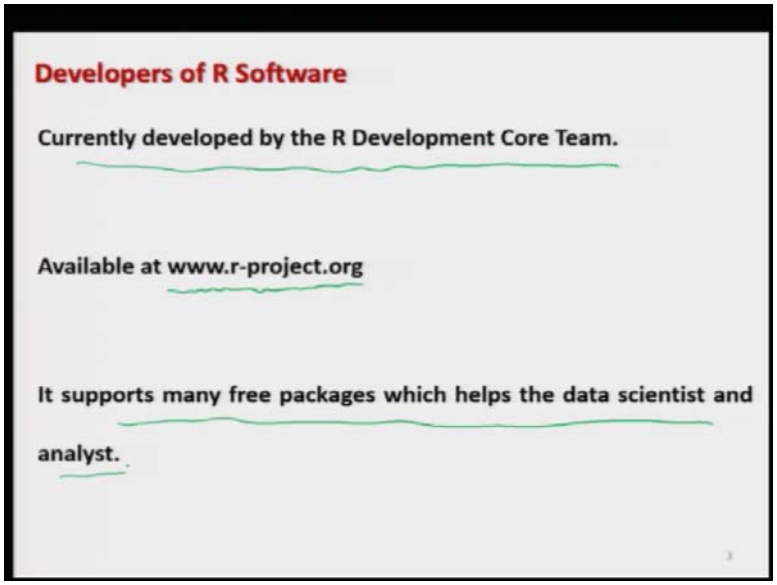
(Refer Slide Time: 03:18)



So, the first question comes here; what is R software and why it is needed. So, we all know that this use of software is desirable and now moreover it has become an essential part of any statistical analysis, any data analysis, without this you just cannot do it. And when I come to software, then with then there are some software which are dealing with statistics they have a emphasis on statistics. And such a popular software are like SPSS, SAS, Minitab, Stata, Matlab, they were popular, they were they are in the market for a very very long time. And similar to this, there is another software which is called here as a R. One of the biggest difference between these software like SPSS, SAS, Minitab, Stata etc. and R is that R is a free software, whereas, SPSS, SAS, Minitab, Stata, Matlab etc. they are the paid software, right.

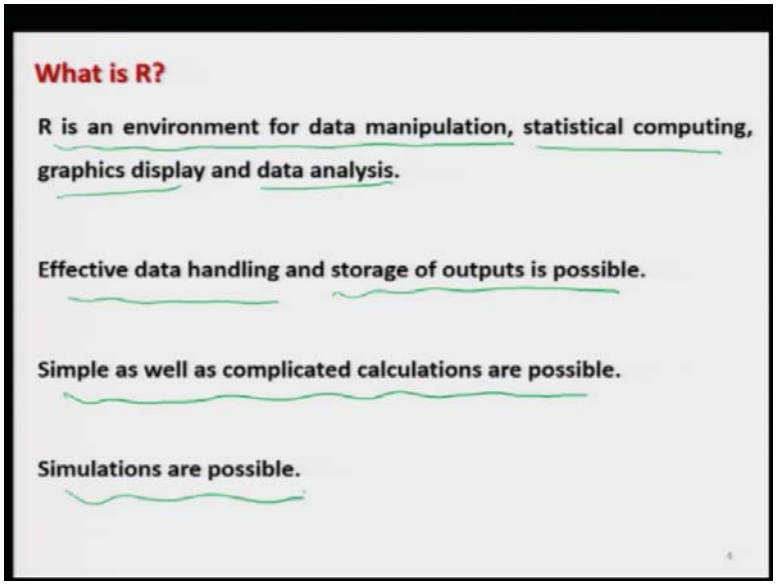
Remember one thing I am not criticizing any of the software. I am not saying that what is good or what is bad, but I am simply trying to choose the software R for my analysis. And one of the biggest advantage in my opinion is that this is a free software. You do not have to pay for it.

(Refer Slide Time: 04:43)



And this R software is currently developed by a team which is called as R development core team. And this software is freely available to download at www.r-project.org. And this software supports many free packages which help the data scientists and analysis in different types of statistical analysis.

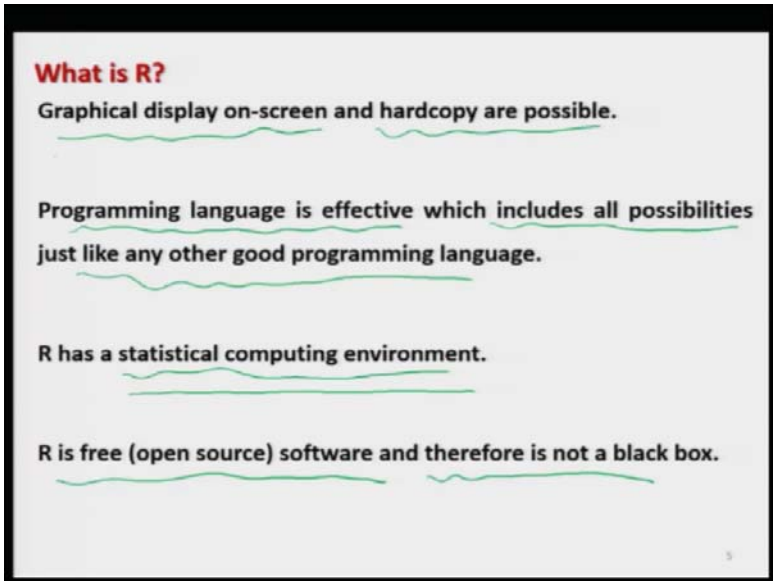
(Refer Slide Time: 05:15)



And just like any other programming language, programming software R is also similar to them and R is an environment for data manipulations, statistical computing, graphic display, data analysis simulation etc.; all those things whatever we can do in a normal software they are also possible in this software.

The effective data handling, storage of output is possible just like other software. Simple as well as complicated computation calculations are also possible in R software. Simulations are also possible in R software. So, in a nutshell that means, I can say that whatever type of computation simulation etc. whatever are possible in a usual software, they are also available in R software.

(Refer Slide Time: 05:57)



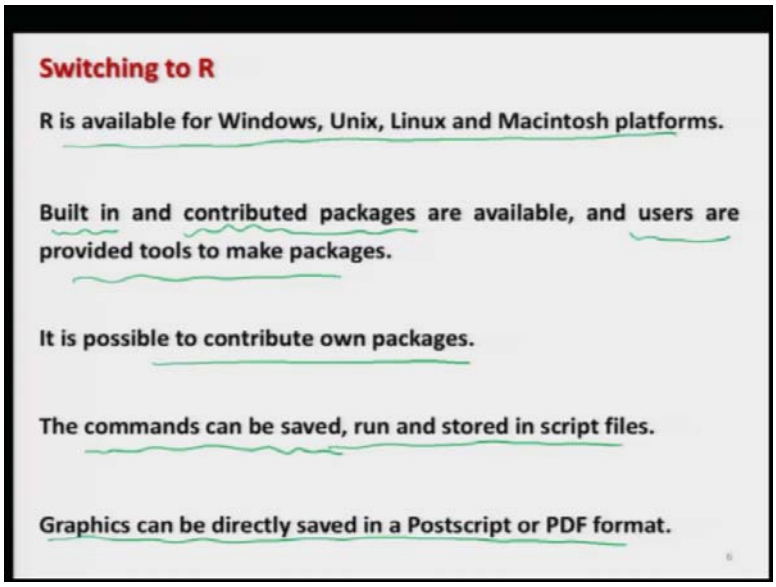
And when I come to the graphic the graphical display on a screen as well as the hard copies are also possible, the graphics can be saved in different formats like a pdf, post script, etc. And then R also has a programming language, right. And this programming language is quite effective and this includes all sorts of possibility just like any other good programming language and which also supports the statistical computing environment.

What you really mean by statistical computing environment? For example, if I want to do certain computation for the linear regression modelling and time series modelling, then there are special package which can be downloaded and those packages can be feeded with the required input and they will result in the statistical computing.

And since R is a free software, that is an open source software, so that is why it is not a black box. Black box means sometime it happens that when we try to do some analysis in any software many times we have no idea what the software is doing, but we get only the outcome.

So, R has an advantage that it is an open source software. So, one can really look what sort of tools have been used in the programming, and how this outcome is coming and how and what is the methodology or the programming language or the rules which have been used in getting that outcome.

(Refer Slide Time: 07:36)



Another advantage of R is that is that is R is available for Windows, Unix, Linux and Macintosh platforms, right. And there are two types of packages in R, one are built-in packages and one are the contributed packages.

And in case if somebody wants, someone can also create their own packages. It is also possible to contribute own package. What does this really actually mean these contributed packages or built in packages? You see in statistics, we are going to do different types of computation, different types of statistical manipulations.

Now, in case if a software is created in which all those possibilities are combined, then one outcome will be that the out that the software will become too heavy, the size of the software will become too large. Secondly, means everybody is not really going to use all the packages, means everybody is not really going to use time series package, everybody is not going to use the regression analysis, everybody is not really going to use the design of experiment.

So, what R development core team thought? They thought that ok, there are certain aspects which are commonly useful, they should be embedded in the base package. So, they created a

small package which is called as R base package and the commonly used commands are available in the base package.

Now, there will be some specialized tools required. For example, if somebody wants to use the time series, somebody wants to use the design of experiments. So, for that what they have done? That they have created the special packages, a special software I can say which are the integral part of R software and those parts of the software, they are uploaded on the websites of R project.

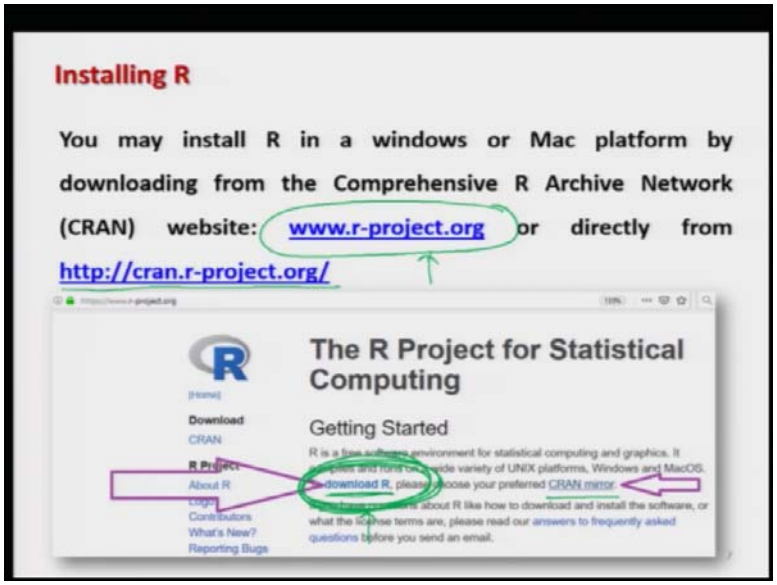
And whenever you need it, they can be downloaded, they can be installed within the R software, they can be used. So, now once we try to think about these type of packages. So, these packages are created either by the R development core team or they can be created by those people who are trying to develop a tool.

For example, if I do some research and if I develop another tool, then I have an option that I can create a package, I can do the programming of the methodology which I have developed and then I can contribute this package in the R software. After verifying, this software will be uploaded on the website of R software and anybody can download it. So, this is what we mean by contributed packages.

So, different people who are working in a statistics, those who are trying to develop new tools they try to develop those packages and they try to contribute in the R software. So, both the things are possible that one can download the packages which are developed by the R development core team or developed by individuals, ok.

Now, after this that R will also have some commands those commands can be saved, they can be run, they can be stored in a script file and just like any other software a group of a commands can be written in the form of a program or they are called as a script file in R or they can be or the individual commands can also be executed, right. And as I said the graphics can be directly saved also in the pdf format postscript format, jpeg format and various types of other things.

(Refer Slide Time: 12:24)

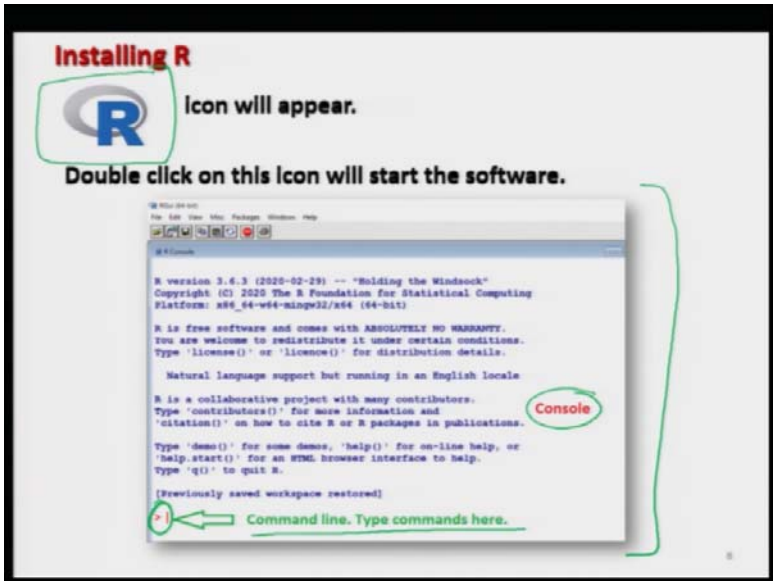


Now, in case we want to install the R on our computer. So, there are, there can be several options whether we have a window system, Mac platform or Linux platform or Unix platform, whatever it is we simply have to go to the website www.r-project.org and or directly if you go to this website this cran.r-project.org.

And actually this project can be this site can be can be exit correctly from the website www.r-project.org. And once you go there to this website you will get here in the information like here download R or you can choose here this CRAN mirror, actually CRAN mirror is the comprehensive R archive network.

So, different people have or say different places, different institutions have uploaded the R software and its package on their servers. So, in order to distribute the load of distribution of the R software and as well as R packages, these different sites have been created which are holding the R software as well as the packages and one can use any of them. So, you come here you try to click on this link over here download R, you will get a software, you can save it on your hard disk.

(Refer Slide Time: 14:17)



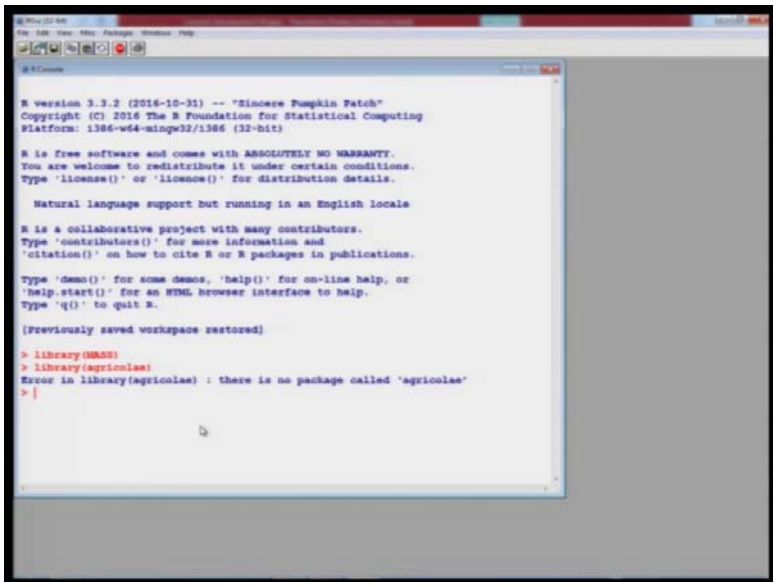
And then you simply have to just double click on the software, and then it will ask you several types of information and you simply have to just give the information, and then the software will be installed on your computer and finally, you finally you will see a icon like this one here R.

And if you double click on this icon, you will the software will begin and you will get here a window like this thing, right. So, I can show you here this window on my computer and actually first I can show you here, that if you come on this here desktop you can see here, this is the here R icon which I am moving here.

(Refer Slide Time: 15:04)

And if I double click it here I will get here this type of window.

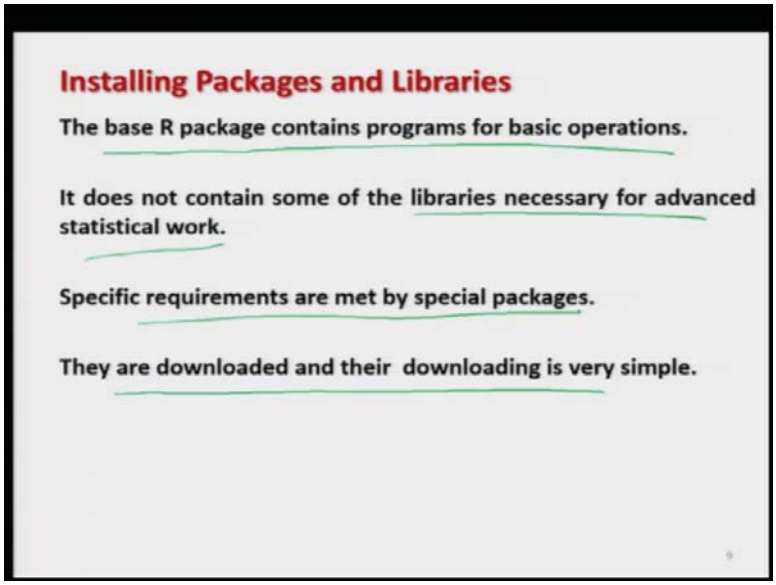
(Refer Slide Time: 15:16)



And you can see here, here it is written R console. There are different types of options here file, edit, view, miscellaneous, packages, windows, help, etc. they are just like any other software. So, I have taken a screenshot here for your convenience, so that I can explain it here.

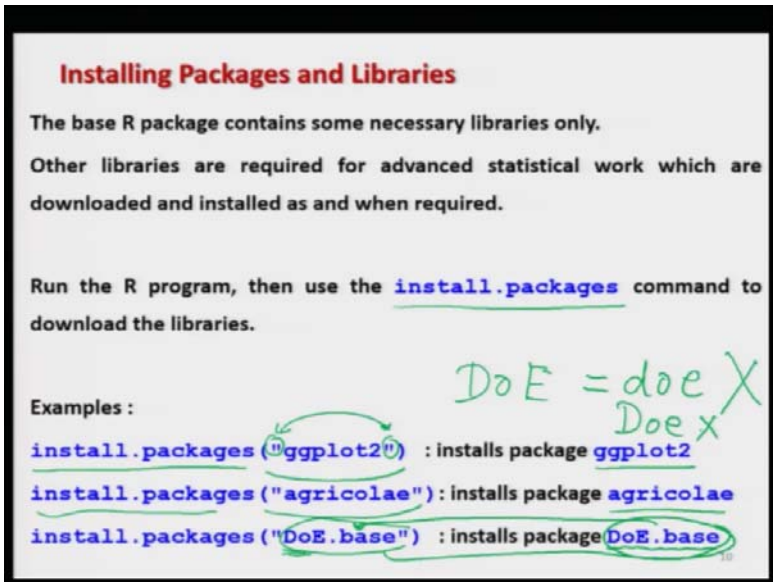
So, you can see here that in that we are getting a window like this one, this is called actually here a console. And here you can see here that there is a symbol here just like greater than, this is called as command line and we try to type our commands our instruction at this place, right. Sometimes people try to use this technical language like console, command line and people really get badly confused what are those thing.

(Refer Slide Time: 16:12)



So, I just want to convey that these are very very simple thing; you simply have to look into the nomenclature. And as I said, the package which you will install this is actually the base package of R and it contains the programs only for the basic operation. It does not contain any special libraries etc. for advanced statistical work. So, for any specific job, specific requirement you need a specific package, they can be downloaded from this site and this downloading is also extremely simple.

(Refer Slide Time: 16:52)



I will try to show you here. And, but before going further let me try to explain you in a more simple way. So, you have here a base package R. Have you heard the word library, in your college, in your university, in your school? What is library? Library will usually have a

collection of books or different types of things nowadays like as a CD, DVD or some electronic books, files, notes, etc.

And what do you do? Suppose you want to study a topic. You will not have that book in your home. So, you go to the library, you get it issued, the book is issued for some time, you bring that book to your home or to your office and you study from the book. And once you are done, once you have learned the subject, once you have finished the job, then you go back to the library and you return it.

And if you do not do it, then what will happen? Means if you start buying all the books, your bookshelf at home or in your office that will be continuously getting filled up and there will be too many thing in your office which are no more required, ok.

So, same thing is being followed in the R package also, that there are there are libraries for doing specific job. For example, if you go to the physical library, there will be books on physics, books on chemistry, books on mathematics, books on statistics, books on computer science. You go to the shelf; you try to choose the book which you want, get it issued.

Similarly, in the case of R software also there is a library, there are different types of packages, we try to go to the library and we try to download those packages whatever we need. Once we download it, we install it. Now, whenever I need it, I need to first call the software, means if all the packages are always uploaded, the R software will become too heavy. The speed will become less.

So, whenever I want to use a software, suppose today I decide I have to do a time series modelling. So, I will download the software and I have to install it, and then I will load the software by specific command. I will finish my job and I will return the software to the sorry; I will return the package to the software. So, that my basics, base software is not too heavy to be executed. So, this is how we do in R, right.

So, and remember one thing whenever you are trying to install a package on a computer that package will be installed only on that computer, means you cannot say that if you have got 2 laptops or 2 computers in your office and if you install the package in one computer and if you try to perform your analysis in another computer, that will be automatically installed in that computer, no. This is this will remain only to that laptop or that computer where you have installed, ok.

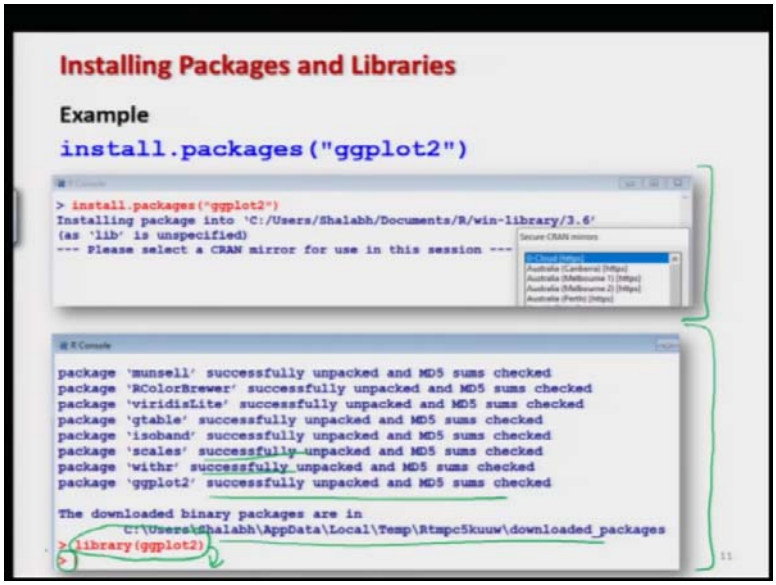
So, now, so in case if I want to install a package then I have a command here install dot packages, i n s t a l l dot packages. And when I use this command install dot packages and within the parenthesis and inside this double quotes, I have to write down the name of the package which I want to download.

Suppose, I want to download a package whose name is ggplot2, right. This is the package which is used for creating different types of nice graphics in R, so I have to simply give this name inside this double quotes over here ggplot2 and you will see that the software will be installed.

Similarly, if you want to install another package whose name is agricolae that is a g r i c o l a e, then you have to give the name inside the bracket within the double quotes, and you have to give the command install dot packages. And then the package will start getting installed on your computer. And similarly, if you want to install another package for design of experiment called as D o E dot b a s e, then you have to give the exact name to be here.

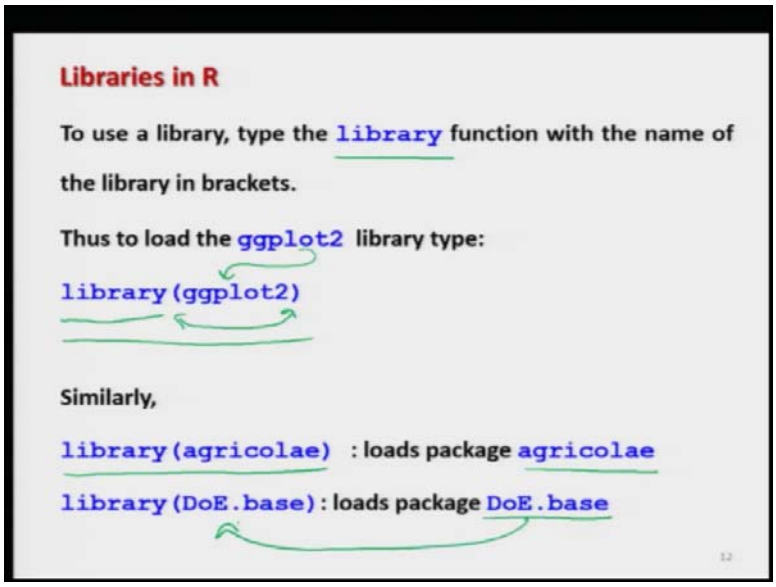
Remember one thing if the name has capital D, you have to give, you have to give here the capital D. This name and the name of the project, they should be exactly the same, small letter means small letter, capital letters means capital letters. You cannot say that that D o E is this same as d o e, no, this is wrong, right. Or even if you write down here D o e, this is also wrong. So, this has to be exactly the same.

(Refer Slide Time: 22:48)



So, once you execute this thing, you will get these types of screen. I have given you here the screenshot. These types of screens will come and then finally, you will see that you will get here message or something like here that means everything is successful and this package has been installed in this directory of your computer, right.

(Refer Slide Time: 23:24)



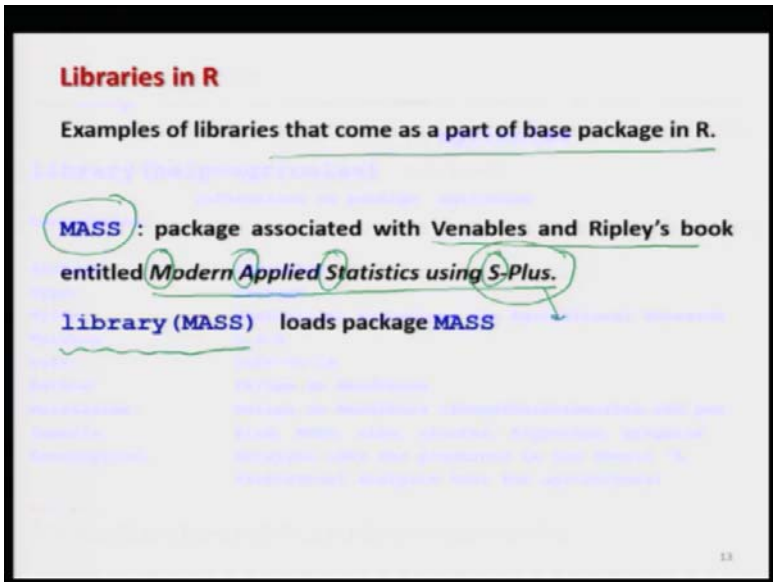
So, this type of and then after that now this package has been installed on your computer, but it is not, but you have to call it to use it, right. So, when you want to call it, means it is something like this there are three people in your office and if you want to take the help of one person you have to call the person by its name. So, that that job is been created by the command library l i b r a r y.

And this is a function which has to be written in this format that you write the function library and inside the brackets, you have to write down the name of the package which you want to load, right. For example, if you want to load the package ggplot2. So, you have to write down here library inside the bracket, you have to inside the parenthesis you have to write ggplot2. For example, you can see here I have written here in this earlier screenshot.

And once you enter here after typing library ggplot2 you will get here a simply command line here. And suppose if this package is not installed then you will get some error that the package is not installed or something like this then you have to solve it accordingly.

Similarly, if you want to load the package agricolae, then you have to write down library inside the parenthesis you have to write a g r i c o l a e, the name of the exact name of the package. And similarly if you want to load the package for design of experiment DoE dot base, then you have to write down the exact same name here in the function library DoE dot base, right.

(Refer Slide Time: 25:06)



Now, once I take the example of these libraries. So, as I said there are two types of libraries one library is which come as an integral part of the R software and another type of library which you have to have to download depending on your need. So, there are some libraries which come as a part of the base package in the R.

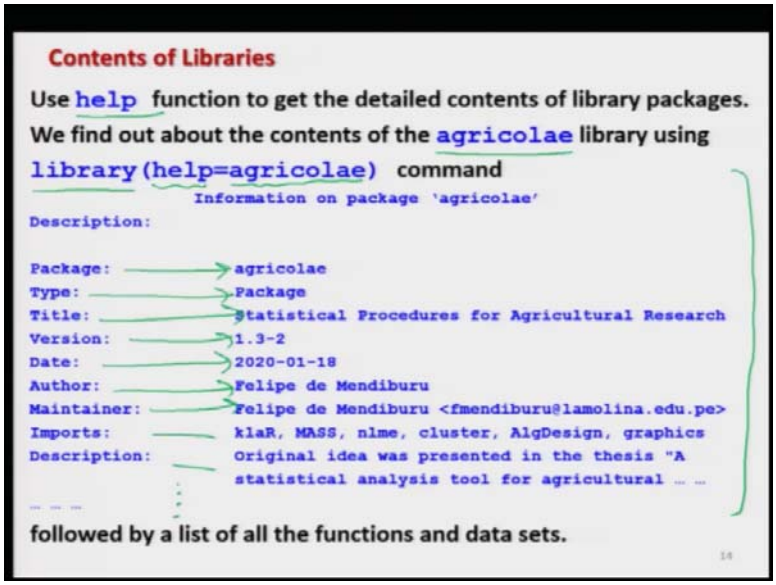
For example, there is one popular library which is here MASS, capital M, capital A, capital S, capital S, which is actually here a package which is associated with the book Modern Applied Statistics using S-plus, so M A S and here S.

This was the book which was written by two professors Professor Venables and Professor Ripley, and this was for say another software what was called as S-plus and actually this just for your information that is R was developed on the same lines as the S-plus. And I remember that in 90s, this S-plus package was extremely popular, there was the R packages was not popular at that time, right.

So, at that time this book was written and if you want to load this package for using it, so you have to use the same command library. Inside the parentheses you have to write the name of

the package MASS, all in capital letters. So, you have to write it exactly in the capital letter and it will load the package and then it is ready to use.

(Refer Slide Time: 26:50)

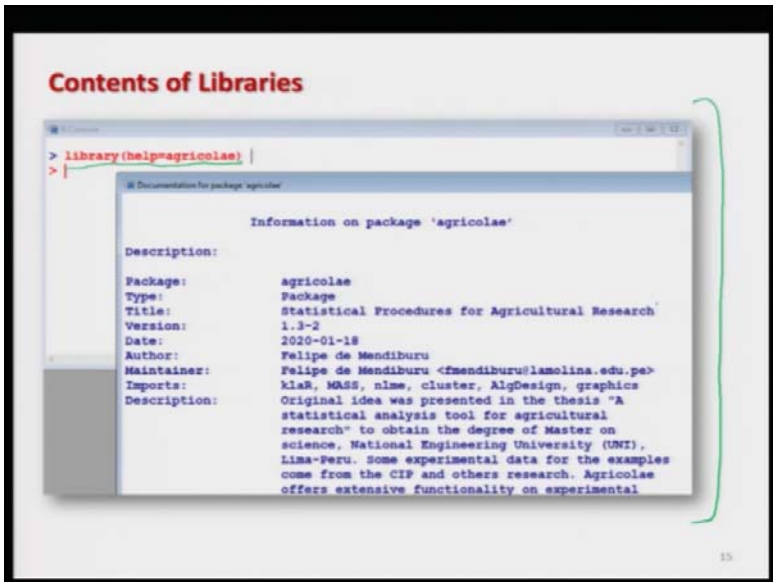


Now, in case if you want to see the contents of the libraries, then there is a help function and which will help you in getting the contents of the library packages. For example, if you want to know what is there inside in the package agricolae, a g r i c o l a e, then you have to use the command like library, inside the parenthesis you have to write help equal to the name of the package is agricolae, right. And then you can just copy and paste this command or you can type this command correctly on your R console and you will get here this type of information.

So, you can see here this is giving you information like what is here the name of the package, what is the type, what is the title; the title is the statistical procedure for agricultural research, what is the its version, when it was created, who is the author, who is maintaining it, and you can contact him and then there are different types of information which are helpful for you.

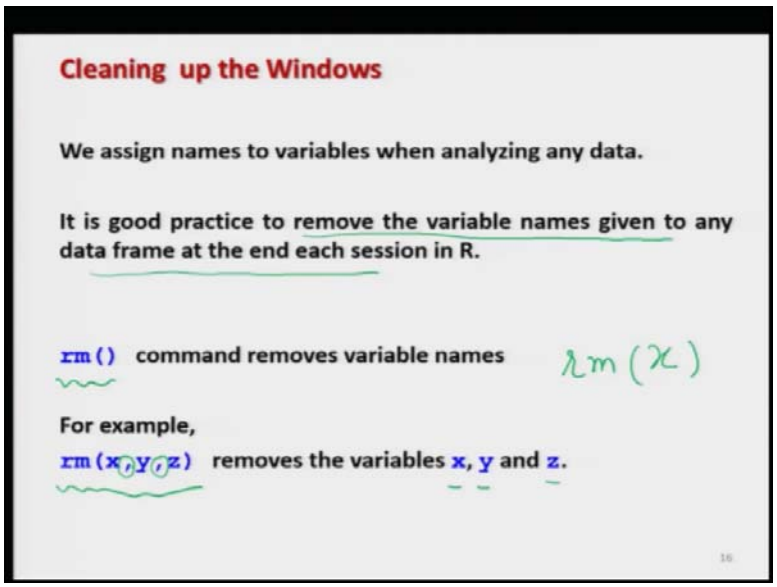
So, that is what I said that it is not really a black box, whatever the information you want that is available to you. And after this there are some other information, I am not presenting it, but then you can see here.

(Refer Slide Time: 28:15)



It is a screenshot of the same thing if you type a library inside parenthesis help equal to agricolae, you will get here this type of information, right.

(Refer Slide Time: 28:31)



Now, once you are working in the analysis you will try to use different types of variables for analysing a data, and it is always a good practice that you remove the variable names or the data frame after you have completed your discussion. Means I can give you a simple example which is one of the most common mistakes, one which is one of the most common mistake we make in real life when we are dealing with the data analysis.

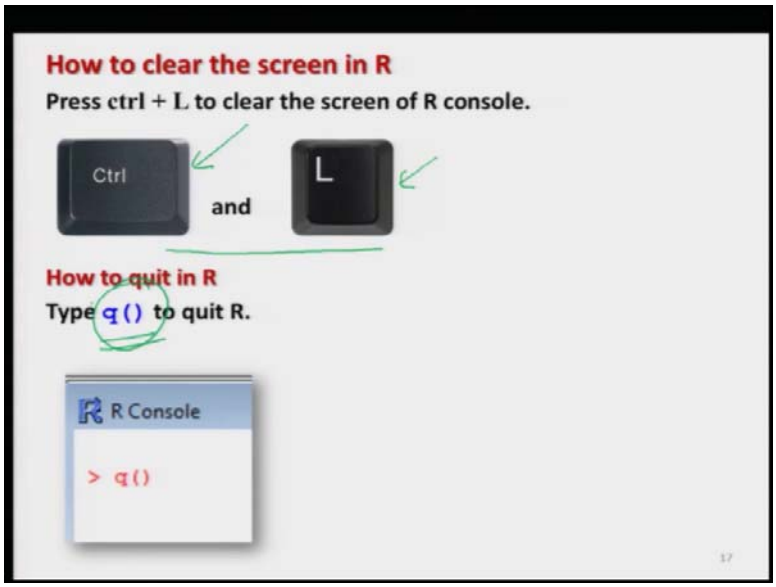
Suppose there is some data and if I use here use the variable name as age to denote the values of age, and I do my analysis. And suppose I finish my analysis, and after a couple of days somebody else comes to that computer or even I am using the same computer and there comes another data set and which also has age of some people.

Now, suppose if I forget that I have used the earlier variable the age or if I simply go to the computer and type age it will show me yes there is some data, and possibly I may not really be careful in verifying whether this is the same data on age which I had entered last week or this is the new data set. So, it is possible that the earlier data set will be used today. So, your statistical outcome will not be really be correct.

So, just to be on the safe side, it is always a better option to clear your library, clear your names, remove all the earlier defined names before you start a new analysis. So, this job can be done by the command rm that is remove. And if you want to remove a variable, suppose if there is some variable here x.

So, you have to write down here rm inside the parenthesis you have to write x and if there are more than one variable suppose there are three variables small x, small y, and small z, then I have to use the command rm inside the parentheses. I have to give all the names separated by comma like x comma y comma z and it will remove all the variables, right.

(Refer Slide Time: 31:14)



And once you are working on the screen you can, so something will be there on the screen and suppose if you want to clear the screen, right, so then you have to use here the two keys together control plus L, something like control key will be denoted as ctrl on your keyboard and this capital L alphabet key you have to press it together.

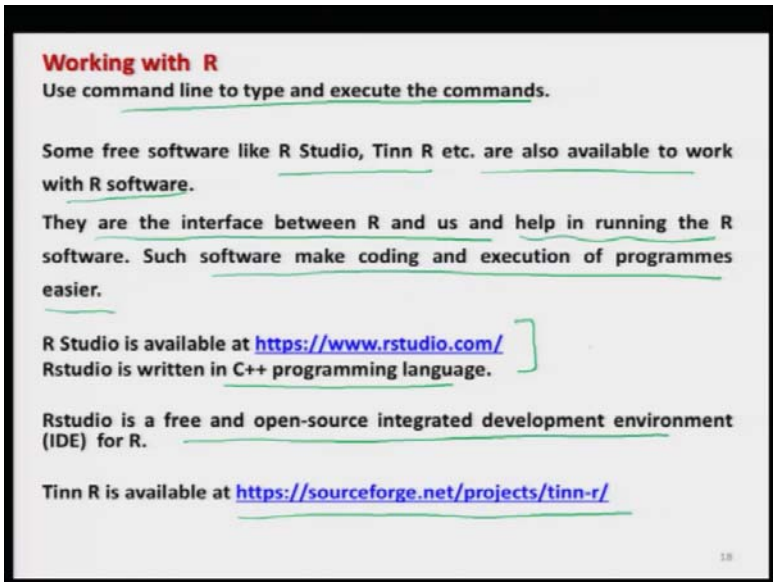
And in case if you want to quit from R you have to simply type the command here q and the parenthesis. For example, I can come to the R software and I can show you that how the things are happening. For example, if you see here, now if I want to load here a library I can say here library and I can see here MASS, right. You can see here I get here this thing. Now, the library is installed here.

Now, on the other hand if I try to install here another library, let me show you. Suppose I try to see here library agricolae, right. Now, you observe what is really happening. It is saying that there is an error in the library, there is no package called agricolae that is correct, up to now you have not installed this package.

So, if the package is not installed on your computer and if you try to call it, it will simply say I do not have. It is just like if you go to a restaurant and if you ask any dish and the person says no, we are sorry, we do not have. You can get a food in the restaurant only if that is available there, right.

Now, in case if I want to clear the screen here whatever is written here you can see here, now I am going to press on my keyboard control and capital L key together, control L and you can see here the screen is clear. And now I can rewrite the things over here. So, let me come back to our slides and proceed. And, means I am not using here the q function, but you can try with the q function, otherwise I will have to restart the entire program. So, anyway you can use here the q and within the parenthesis to quit R.

(Refer Slide Time: 33:54)



And when we are working on the R, then first option is this I can type all my commands and everything on the command line and they will be getting executed from there itself. There are some free software like as R Studio, Tinn R etc. which are also available to work with the R software.

Actually, these software are the interface between R and us and sometime they try to give you more information in a compact way. And they also help in the execution and the running of R software, R commands. And in case if we try to use such software, so our life becomes easier and more comfortable, like when we try to code a program or when we try to execute it.

So, this R studio software, this is also a free software which is written in the C ++ programming language and this is freely available at this website www r studio dot com. Well, they have different types of version, but there is also a free version available which is sufficient for us. And this is also a free and open source integrated development environment for R.

And similarly, there is another software here Tinn R t i double n and this software can be downloaded from this website sourceforce dot net and under that projects under tinn R, right.

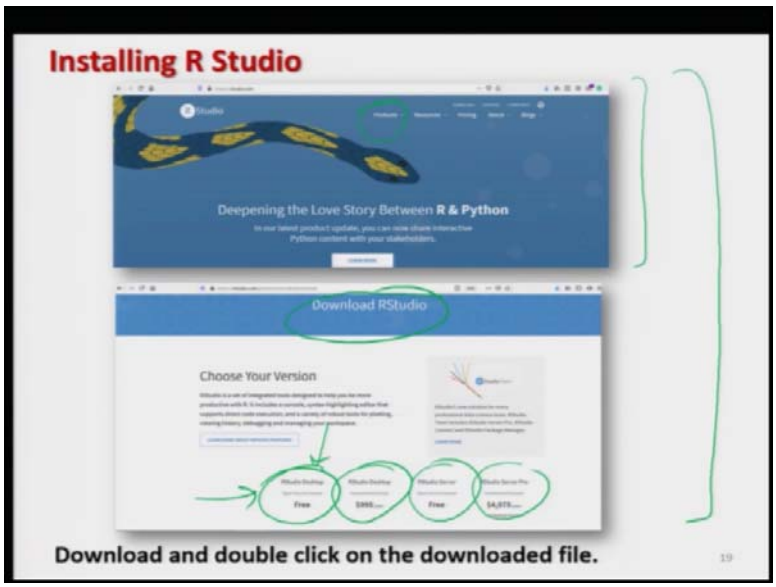
But anyway, I am not saying that you have to compulsorily use any of this software. This is only my advice. And even I am not making any comment whether R studio is better than R whether this Tinn R or vice versa. I am leaving all these decisions only to you, you have to

choose where you want to work correctly on the R console or through R studio or through Tinn R.

Well, during my course I will be working directly on the R console, right. So, now I will simply try to give you a very quick review of the R studio. Well, once again I will say that I am not doing any canvassing for R studio, you can use any of the software. But this is just for your help.

And my objective of showing is this I just want to show you that in case if you try to use these types of interface, many of the information you can obtain more compactly which is easy to handle rather easy to understand. You can view lots of information together in a single shot.

(Refer Slide Time: 37:08)

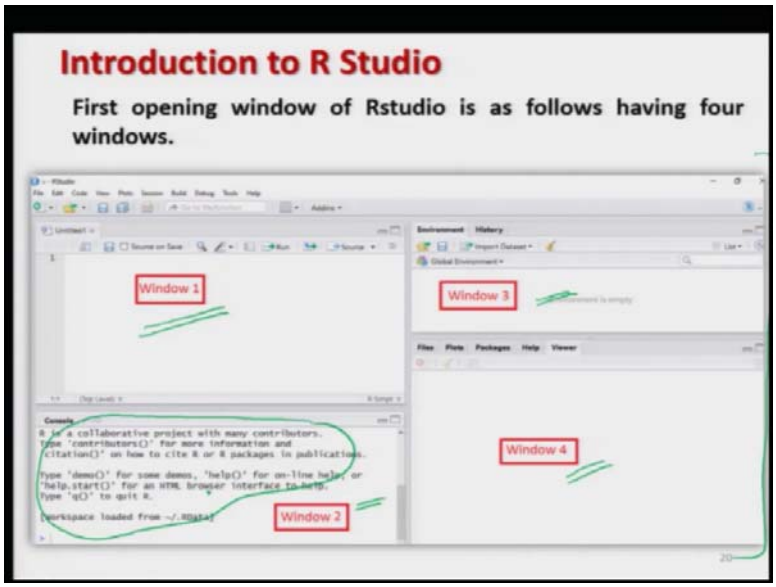


So, if you try to go to the website of R studio, you will get here this type of say page. For example, I have just taken the screenshots. So, right now you can see here this is the latest screen say screen of R studio.

Now, if you come on the see here because your option here something like your product, and then if you come to a location where it will ask you whether you want to download the R studio or not, and you can see here this is here a free version, there is a paid version and so on, but anyway I am talking of this thing.

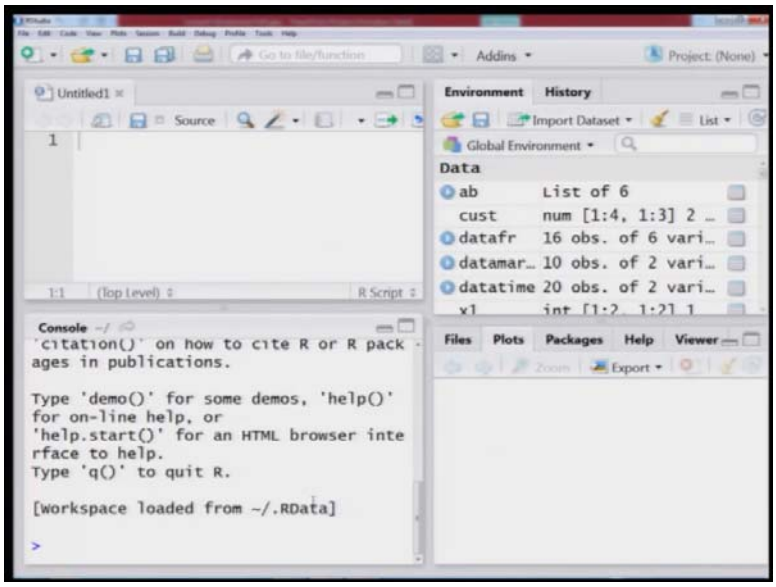
So, you can simply double click on this button R studio desktop which is for free, and then the file will be available on your desktop you simply have to install it just by making click click click.

(Refer Slide Time: 37:57)



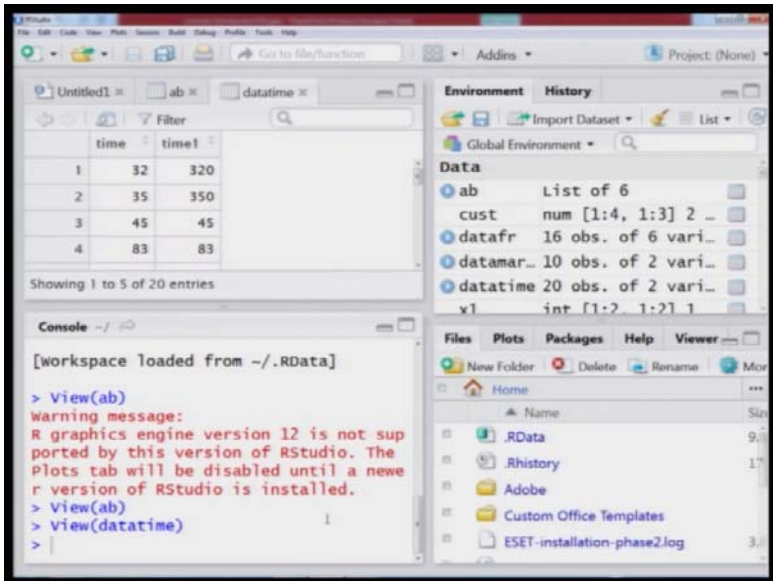
And once you start the R studio from your from the R studio icon you will get here this type of say screen, right. Firstly, let me show you here how it looks like. I have this R studio installed on my computer and I already have started it.

(Refer Slide Time: 38:16)



So, you can see here that these are the here the four screen. This is the first screen where my this cursor is blinking, then if you see here which I am highlighting here.

(Refer Slide Time: 38:36)



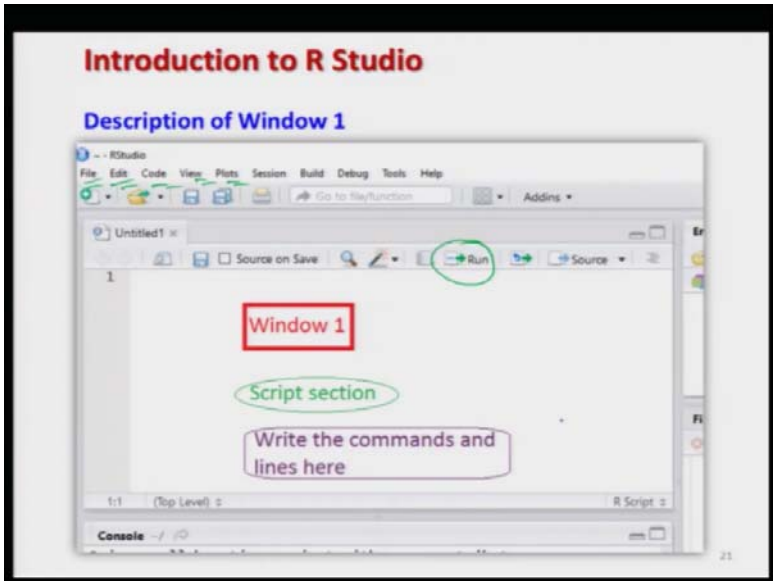
This is second screen and this is here means I can see here. This is my third screen here where my cursor is control is there you can see here and this is here the fourth screen where I it can files, plots, packages, help, viewer, etc., etc., there are different types of thing, right.

I have just clicked on different things at random. I do not want to show you anything, but I will let me come back to our here this package. And means I will just give you quickly what is really happening in these four windows.

So, you can. So, you have seen here there is a window here, which I am calling here as a window 1, and then I have the window in the bottom which I am calling as window 2, then the window on the right hand side in the top this is called as window 3 and the last window that is in the bottom of the right hand side, this is called as here window4.

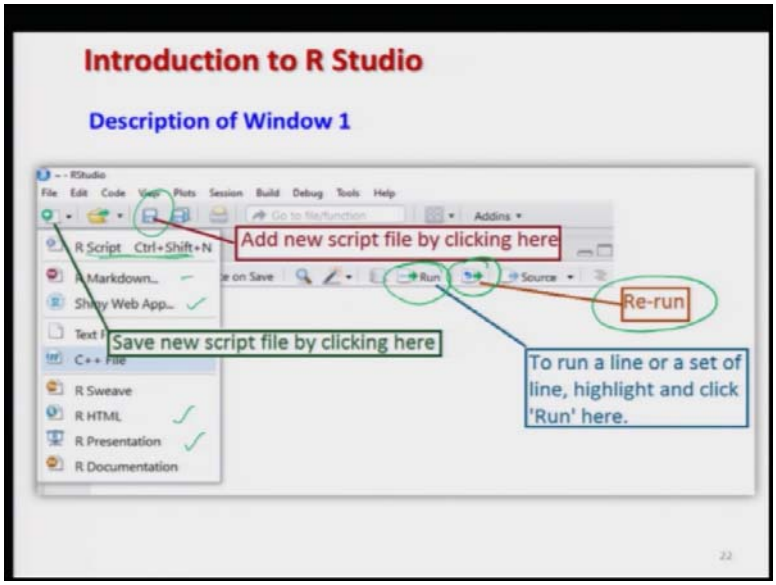
So, now I will try to give you here one by one quick review of these windows what can be done. But do you see here something how does this look like? Can you identify it what is here? Remember, I had just shown you the R console. Can you compare this R console over here? So, you can see here that this is going to give you here the same thing what you saw on the R console, ok. Anyway, let us come to our window 1.

(Refer Slide Time: 40:03)



So, this window 1 is used for writing the program you can type your all your commands over here and just by highlighting the command and then pressing here say run, you can execute the program, and you can see here there are different types of options, which are very familiar to all these all of you and this is file, edit, etc., etc. So, you can play with it.

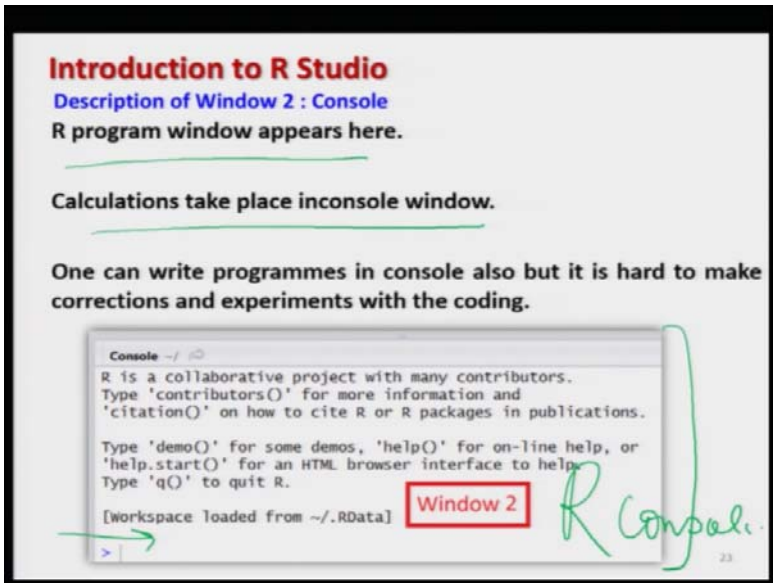
(Refer Slide Time: 40:39)



So, this, but this window 1 is essentially used to write down here the commands. And similarly once you come here for more detail, means I have I have just taken a screenshot you can see here you can open different types of new file for the R script or markdown, shiny web apps and R HTML, R presentation and you can save here, right.

This is for the execution, if you want to re-execute, re-run the lines this is the command here. There are many things, many options which I have not discussing here. But I will request you to just play with it, right.

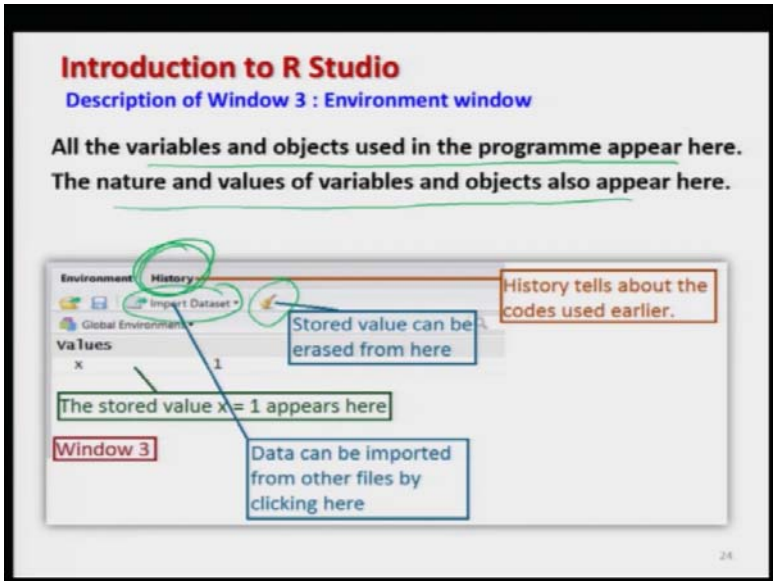
(Refer Slide Time: 41:16)



And then if I come to the window number 2, which I if you see here it was here like this. This is simply the R programming window; this is the same as your R console. Whatever the calculations you are going to take, they are happening here in here only. And one can write their programs in the console directly here also.

If you want to execute you can write it here also, but you can write them on the window number 1 and you can execute it and your outcome will be shown here. So, this is something just like your R console. This is exactly the your here the R console only. So, whatever is happening in the R console you can view it here.

(Refer Slide Time: 42:01)

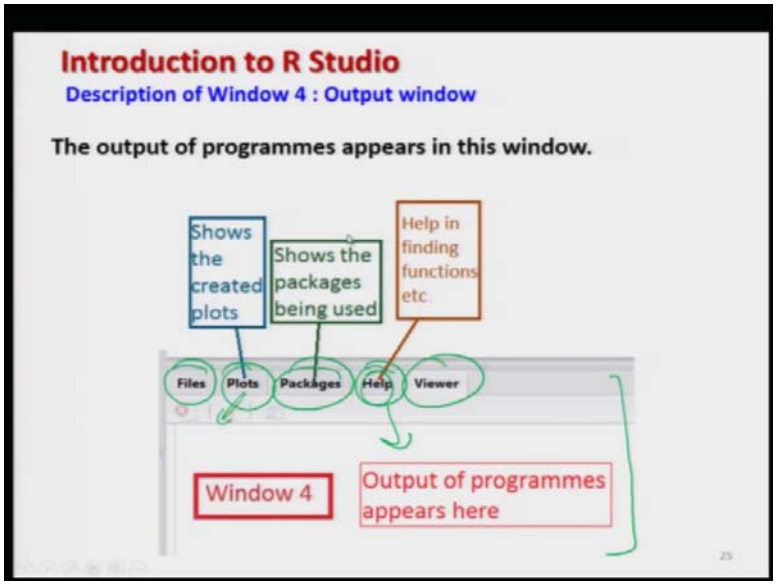


Now, in case if you come to the window number 3. So, this will give you information on different types of variable, different types of environment, and their associated information.

All the variables and the object which are used in the corresponding program they appear here, their nature, their values of variable objects also appear here. And yeah if you have a data set in suppose excel or another software you can use here the command here like as here into same port data set correctly.

And you can erase, you can clean the variables data set etc. from here and this is here a command here history. History will tell you what happened in the past, what you have done in the past on this computer and so on. So, this is what you will get from window number 3.

(Refer Slide Time: 42:54)



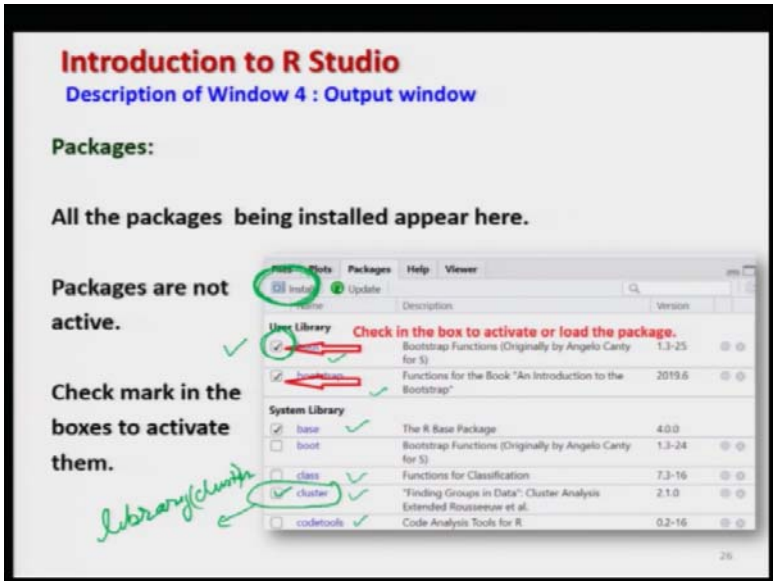
And similarly in the window number 4, there you can see here there are different types of options here file, plots, packages, help, viewer, that is a very helpful window here. Means if you want to install a package here in the R studio you do not have to type the commands like install dot packages.

But you simply have to go here packages just click here, type the name of the package and then it will come in the drop down menu and then you have to select from here and you have to simply press on the click button and the software will get automatically installed, right.

Similarly, if you have a graphics, if you have created any graphic or a plot that can be visualized from here. And yeah means that can be saved, that can be exported, various types of things can be done over here. So, basically this is window where you will get the output and as well as help, right.

Means help in installing the package, help up out the packages, help about the command, help about of the function, different types of things are there in the help menu. So, again I will say that I am not going into detail, but I will request you to please go through with this software try to play with it and then see.

(Refer Slide Time: 44:12)



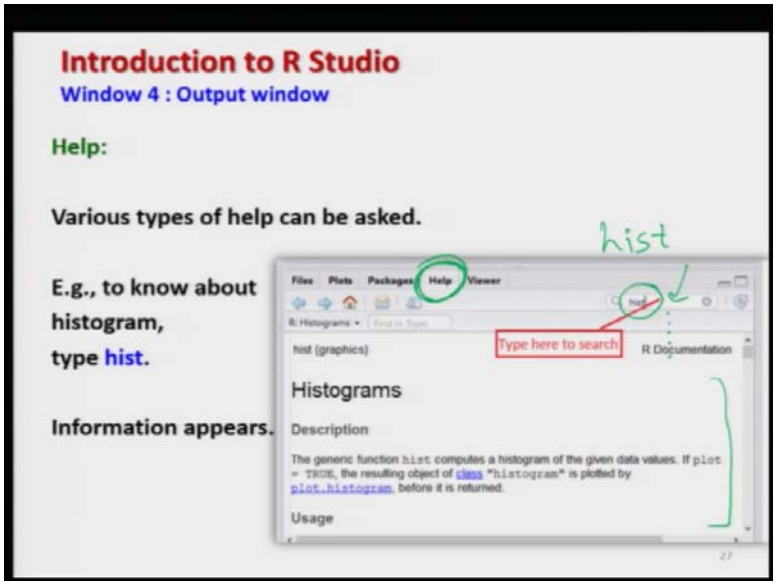
One important thing I would try to show you that for example, you have used here the command install dot packages and library. I will explain you. So, if you want to use them in R studio that is pretty simple. If you want to install any software you just go to this window this button here, install, and then you type the name of the package and it will show you the option and you simply try to choose it. You simply say click, and it will get installed in the due course of actions.

And now after the installation, for example, you can see here there are different names here boot, bootstrap, base, class, cluster, code tools, these are the different types of libraries which I have downloaded on my laptop, on my computer for my use. And suppose I want to use here a library on say boot and bootstrap. So I simply have to go to here this icon and I have to make it here a check mark.

For example, you can see here this command here if you try to see here cluster, cluster is available on my computer, I some time back possibly, I had installed it on my computer, but at this moment I cannot use it. If I want to use it I have to make it here something like check.

And once I do this here check this check is equivalent to your command say library, library and inside the name say here cluster, right. So, that is the advantage you can see here. So, the chances of making a typing mistake in writing the command library or install dot packages they are reduced, right. So, this is what we try to do.

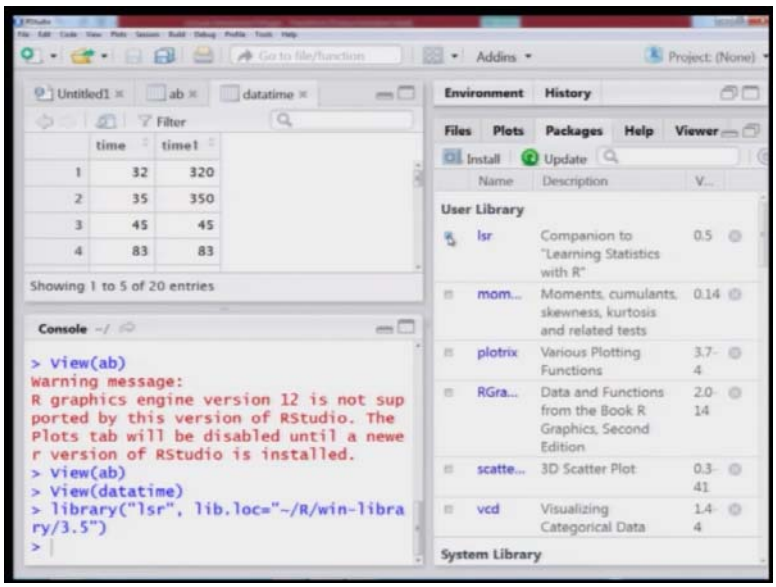
(Refer Slide Time: 46:06)



And yeah and if you want to take any help, just go to go to here help, and try to type here few letters and then you will get here some options here and then you try to choose here. And you can see here, this will for example, I am typing here h i s t, mean suppose I want to have some information about the histograms and I do not know about it, so I have some idea I am typing here histogram.

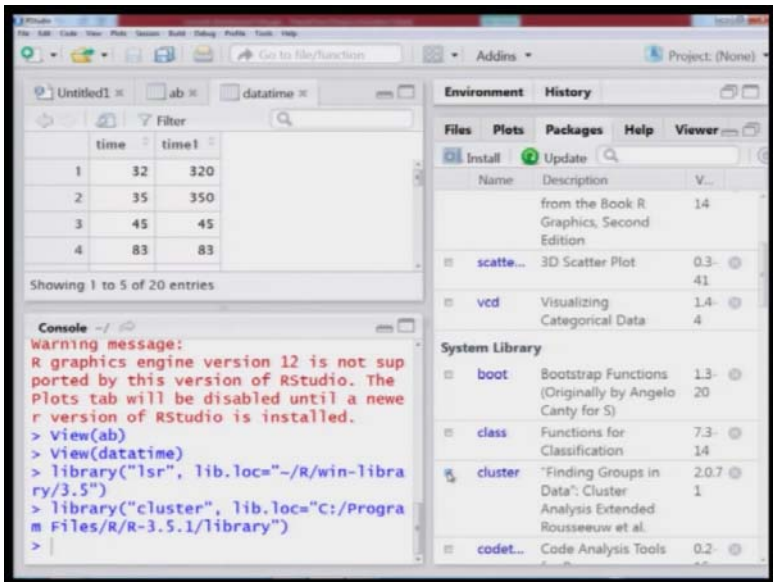
But as soon as I start typing h i s t, I will get here something, and yeah, and if I enter I will get here entire information on histograms. That is the one of the most simple way to get the help or I can suggest here and that is the advantage of a using the this package.

(Refer Slide Time: 47:05)



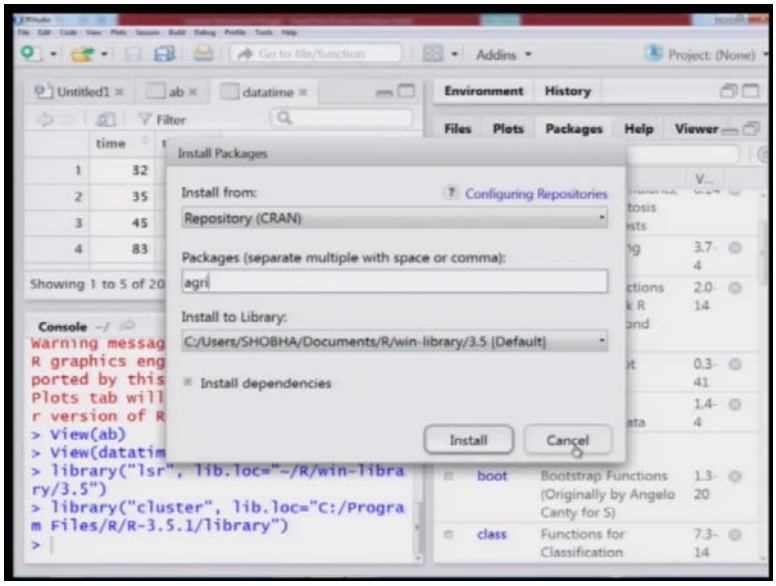
So, and the now in case if you try to come back again to your R studio and if you try to look here; for example, if I want to have here packages, suppose there is a packages here lsr and I want to use it, I simply have to click on this button and you can see here on the R console that it is attached here.

(Refer Slide Time: 47:21)



Or means if you want to say here, another is, suppose this is the higher cluster. Suppose, I want to use here the cluster, suppose and you can see here that the library cluster has been executed.

(Refer Slide Time: 47:35)



And similarly, if I want to install here some package here, suppose if I want to install here the agricolae, you can see here, so agricolae. So, you have to type here this name and then you have to just click, click here on the install and then so on; and then you have to just click on the further commands and it will be installed, right, ok.

So, now I will stop in this lecture. I have given you a very quick review that how you are going to start with the R software. And in the next couple of lectures, I will try to take say smaller topics. I have chosen some topics which I will try to explain you and related commands which I am going to use further in my lectures.

So, because, I am doing it because many times when students starts for a course, suppose if I say simply yes you should have a prerequisite of R software, students are getting confused that how much I should know to do this course. R is vast. So, that is why I am comprehending all these knowledge in this lecture and the next couple of lectures, so that you know that how much knowledge in R software is required to understand the course material which will be delivered in the next couple of weeks.

So, at this point of view only I am trying to take these topics. But my request to you is that unless and until you practice these commands, these topics, you will not be comfortable in executing them. So, you try to practice them, execute them. And I will see you in the next lecture; till then, good bye.