**Rule Variable in R**

1. The variable name must start with letter and can contain number,letter,underscore('*') and period('.').*

**Example:** variableName1,new.variable,

* + Underscore('') at the beginning of the variable name are not allowed.

**Example:** '\_my\_var' is not a valid variable name.

* + Period('.') at the beginning of the variable name are allowed but should not be followed by a number. It is preferable in R to use '.' which helps to separate the different words for the identifier.

**Example:** '.myvar' is a valid variable name. However, '.1myvar' is not a valid variable name because the period followed by a number is not valid.

1. Reserved words or keywords are not allowed to be defined as a variable name.
2. Special characters such as '#', '&', etc., along with White space (tabs, space) are not allowed in a variable name.

## Variable Assignment

Variables in R can be assigned in one of three ways.

1. Assignment Operator: "=" used to assign the value.The following example contains 20 as value which is stored in the variable 'first.variable' **Example:**first.variable = 20**(In R Studio)**
2. '<-' Operator: The following example contains the New Program as the character which gets assigned to 'second.variable'.  
   **Example:**second.variable <- "New Program"
3. '->' Operator: The following example contains 565 as the integer which gets assigned to 'third.variable'.  
   **Example:**565 -> third.variable

## Variable Type

The type of variable in R can be determined by class(),typeof() and mode()

1. **'class()'-** will give the high-level type of an object, which is to say from the perspective of Object-Oriented Programming in R.Depending upon whether it is a vector or any other data structure, the return type will be different.
   * If the type of object is the vector, then one of the following data types is chosen.
     + Integer
     + Numeric
     + Character
     + Logical

For example, the integer type and numeric type will be returned below.  
my.var1 = 8L ( The L specifies that it is an integer and not numeric)  
**Returns:**'integer'  
my.var2 = 8.5  
**Returns:**'numeric'

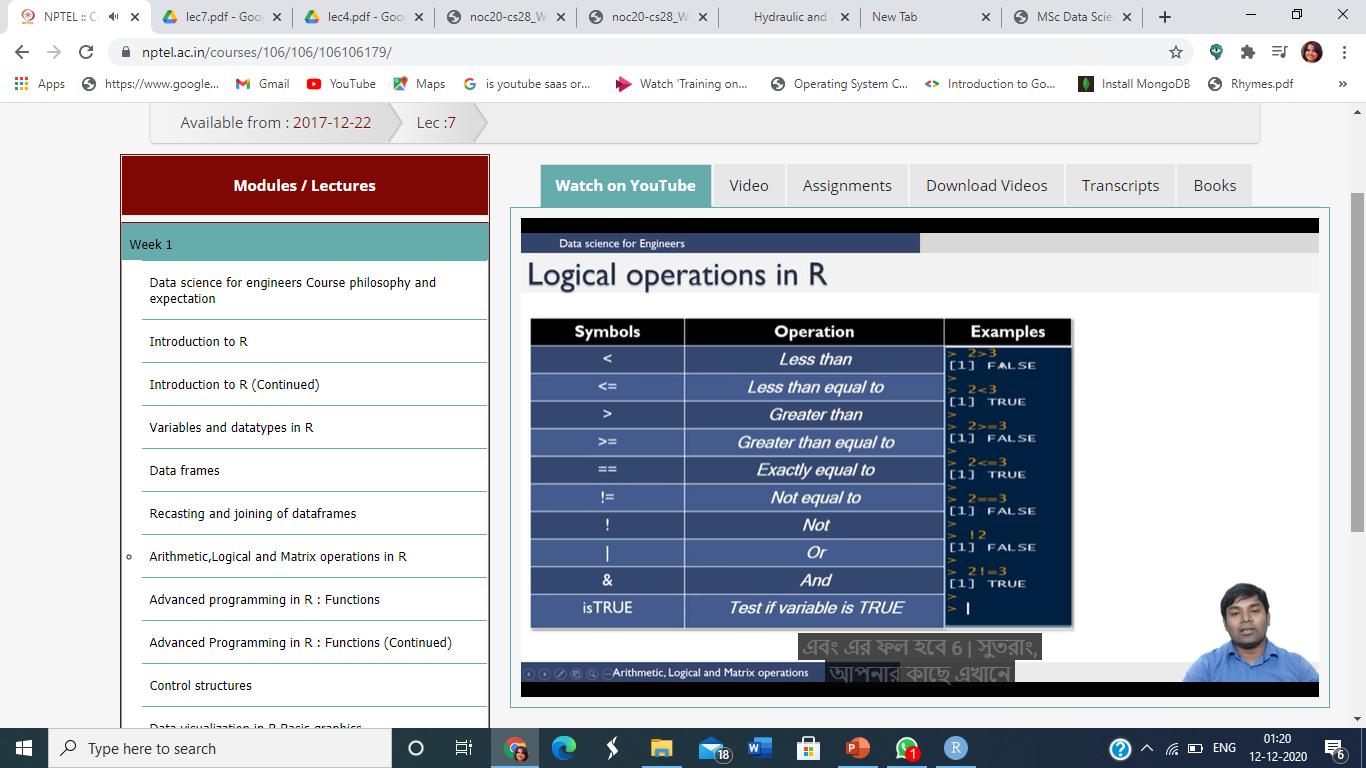
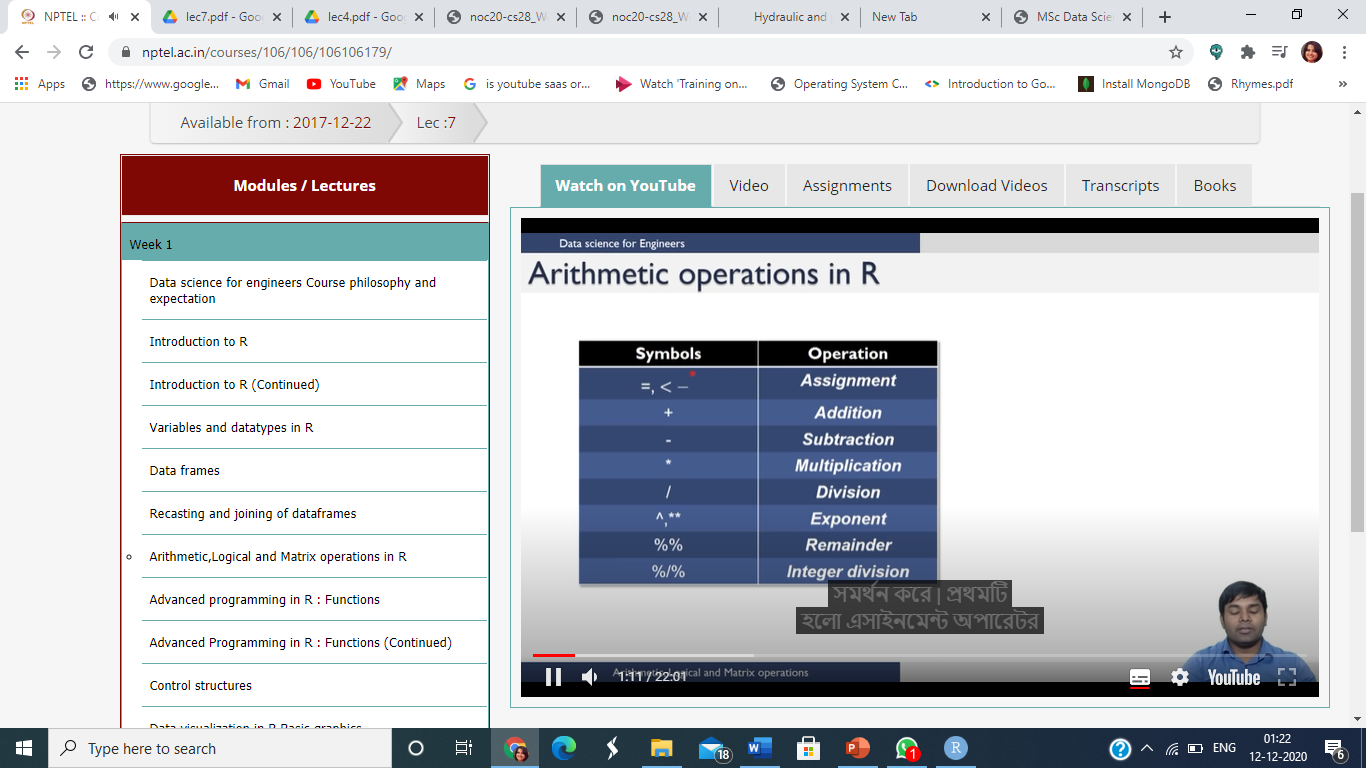
However, with the objects like matrix(),data.frame(),list() and array() the respective type is returned. For example, the list type will be returned as "list" below.  
  
my.var3 = list(6,9,5)  
**Returns:**'list'

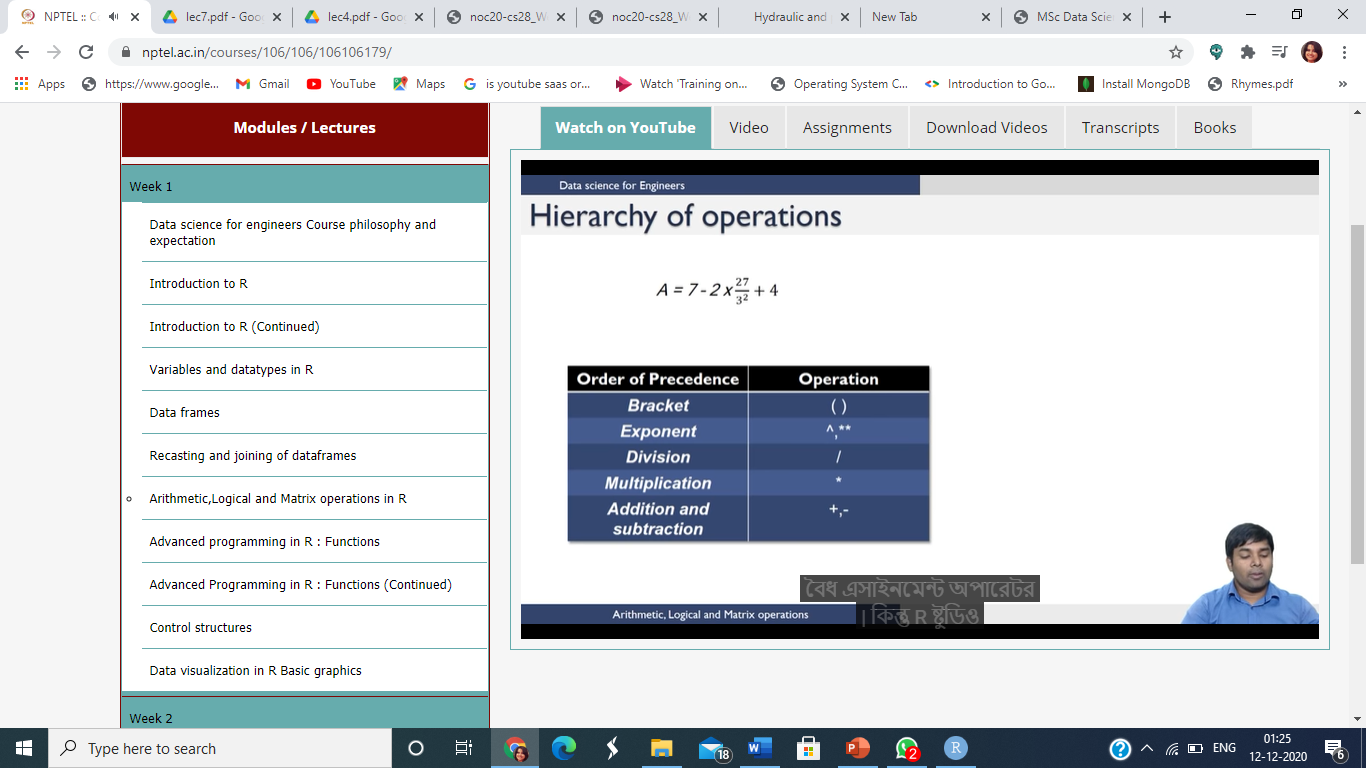
1. **typeof()-** The return values are low-level, which matches the internal type of object in R.It is mostly used by R programmer to determine the type of an object.  
   For Example, some of the internal types of objects in R are shown below:  
     
   typeof(8+6i)  
   **Returns:** complex  
     
   typeof(factor(c(4,5,6)))  
   **Returns:** integer

3.**mode()**- It also returns the values which are the same and closely related to 'typeof().'

## Removing Variable

The variable in R can easily be removed or deleted by using 'rm()'.

In R only <- is used as assignment operator but R Studio allows = and<- for assignment



**Data Read Excersies**

**Exercise 1**  
Read the file [Table0.txt](https://www.r-exercises.com/wp-content/uploads/2015/12/Table0.txt).  
a) Change the names of the columns to Name, Age, Height, Weight and Sex.  
b) Change the row names so that they are the same as Name, and remove the variable Name.

**Exercise 2**  
Read the file [Table1.txt](https://www.r-exercises.com/wp-content/uploads/2015/12/Table1.txt), how many rows and columns does it have?  
a) Re-read the file and make the variable Name be the row names and the headers were taken from file. Make sure you read the variable as characters and not as factors.

**Exercise 3**  
Read the file [Table2.txt](https://www.r-exercises.com/wp-content/uploads/2015/12/Table2.txt), watch out for the first line.

**Exercise 4**  
Read the file [Table3.txt](https://www.r-exercises.com/wp-content/uploads/2015/12/Table3.txt), watch out for the first line and the missing values.

**Exercise 5**  
Read the file [states1.csv](https://www.r-exercises.com/wp-content/uploads/2015/12/states1.csv), the names of the states should be the row names.

**Exercises on dump,dget and connection**

1. **Use of Dump() and Source()**

* Create a vector v with elements: “Weather”, “100”, and dataframe d with two elements: element a is a vector with two integers with value 10, element b is vector v.
* Use dump function to store v and d at file “dumpdata.R”.

**v=c(“Weather”,”100”)**

**d=data.frame( a=c(10,10), b=v)**

( syntax: dump(c(var1,var2,…),file=”filename”)

* Remove object v and d
* Use source() function to read object v and d back to memory and print it out

1. **Use of Dput() and Dget()**

* Create a dataframe Students with columns vector of RollNo,Name, Total
* Use dput to add it to a file “dputdata.R”
* Remove object Students
* Use dget() function to read the object

1. **cUse of Connection to read from text file**
2. Create a text file Employee with columns Empid, Empname,Designation and Salary and a connection to it using file() function
3. Open a file using open() function in read mode # open(con,”r”)
4. Read from connection using read.csv() # data1=read.csv(con)
5. Close the connection #close(con)

**Use of Connection to read from url**

1. Open a URL connection to this link: http://www.stats.ox.ac.uk/pub/datasets/csb/ch11b.dat
2. Read the webpage to data.
3. Check the first 50 rows of these data

**Use of Connection to read from gz file (zipped file)**

1. Open a connection to a gz file # con= gzfile(filename)
2. Readlines from it without extracting