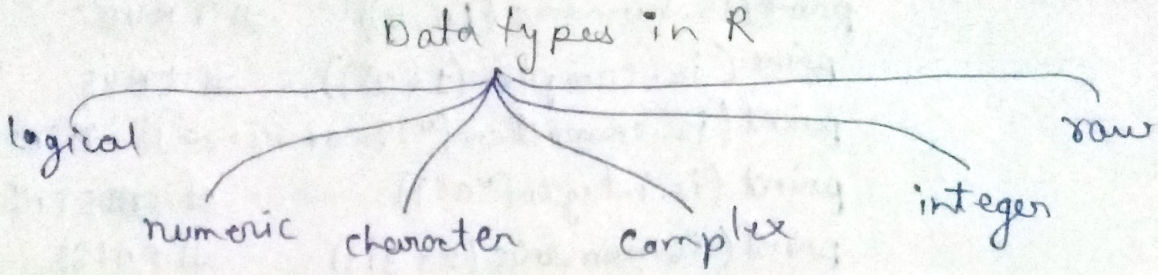
DATA TYPES



(1) Numeric data type.
example. - -12, 1324, 12.89, 12, -34.78
the type of numeric data type is double.

(2) Integer data type.
example- 34L, 344L, 225L
34l is gives an error.

(3) Complex data type:
example. 10-26i, 2+2i, -32+0i

(4) Logical data type:
example. - TRUE, FALSE
not use "0" and "1" in logical data type

(5) Character data type:
example - 'a', "hello", "FALSE", '234'

(6) raw data type:
it is hold the raw bit

* To print the class name and type of the variable use class( ) function and typeof( ) function.

Type verification:
To do that, you need to use the prefix "is." before the data type as a command.
Syntax: is.data_type(object)

example:

```
print(is.logical(TRUE))                    # TRUE
print(is.integer(3L))                      # TRUE
print(is.numeric(10.5))                    # TRUE
print(is.complex(1+2i))                    # TRUE
print(is.character("12-04-2020"))          # TRUE
print(is.integer("a"))                     # FALSE
print(is.numeric(2+3i))                    # FALSE
```

Coerce or convert the data type of an object to another:

- The process of altering the data type of an object to another type is referred to as coercion or data type conversion.

Syntax:

```
as.data_type(object)
```

- All the coercions are not possible and if attempted will be returning an "NA" value.

- examples:

```
print(as.numeric(TRUE))
print(as.complex(3L))
print(as.logical(10.5))
print(as.character(1+2i))
print(as.numeric("12-04-2020"))
```

output:

```
1
3+0i
TRUE
"1+2i"
NA
warning message:
In print(as.numeric("12-04-2020")) : NAs introduced by coercion
```