Difference Between print() and cat()

In this lesson, we provide details about the differences between print() and cat().

We'll cover the following

- Difference between print() and cat()
  - Concatenating Strings Using cat()
  - Concatenating Strings Using print()

Difference between `print()` and `cat()`

In a previous lesson, we discussed that both `print()` and `cat()` can be used to display text on the screen. Let's have a look at the code again:

print() cat()

```
print("Educative")
```

The simple printing method in R is to use `print()`. As its name indicates, this method prints its arguments on the R console. However, `cat()` does the same thing but is valid only for atomic types (logical, integer, real, complex, character) and names, which will be covered in the later chapters. This means that we cannot call `cat()` on a **non-empty list** or any type of **object**. So, as its name implies, it converts arguments to characters and concatenates them.

An essential difference between `cat()` and `print()` is the **class** of the object they return. For now, you can assume that class means the type of object. This difference between the two methods can have practical consequences because there will be certain constraints as to what you can do with the **returned object**.

> `print()` returns a character vector. A vector is an **object** in R language. We will be discussing this in the coming chapters. `cat()` returns an **object** `NULL`.

As you can see, `cat()` prints its arguments without quotes. In essence, `cat()` displays its content on the screen or in a file.

Concatenating Strings Using `cat()`

The actual usefulness of `cat()` can be obtained when we have two or more strings that we want to concatenate.

```
# concatenate and print
cat("Hello World", "with R")
```

Using cat() to concatenate two strings.

Concatenating Strings Using `print()`

Since we say that `print()` and `cat()` have almost the same functionality, we can also concatenate multiple strings using `print()` with the help of `paste()`.

```
# concatenate using paste() and then print
print(paste("Hello World", "with R"))
```

Using paste() to concatenate two strings.

In the next lesson, we have a short exercise for you to test your skills in handling strings.