**Unit 3: Functional Programming**

R, at its heart, is a **functional** language. This means that it has certain technical properties, but more importantly that it lends itself to a style of problem solving centered on functions.

Firstly, functional languages have **first-class functions**, functions that behave like any other data structure.

Secondly, many functional languages require functions to be **pure**. A function is pure if it satisfies two properties:

* The output only depends on the inputs, i.e. if you call it again with the same inputs, you get the same outputs. This excludes functions like [runif()](https://rdrr.io/r/stats/Uniform.html), [read.csv()](https://rdrr.io/r/utils/read.table.html), or [Sys.time()](https://rdrr.io/r/base/Sys.time.html) that can return different values.
* The function has no side-effects, like changing the value of a global variable, writing to disk, or displaying to the screen. This excludes functions like [print()](https://rdrr.io/r/base/print.html), [write.csv()](https://rdrr.io/r/utils/write.table.html) and [<-](https://rdrr.io/r/base/assignOps.html).

Collectively, these types of function are called **higher-order functions** and they fill out a two-by-two table:

A grid of black and white text

Description automatically generated

**Chapter 9 Notes:**

A **functional** is a function that takes a function as an input and returns a vector as output.

The most fundamental functional is [purrr::map()](https://purrr.tidyverse.org/reference/map.html)53. It takes a vector and a function, calls the function once for each element of the vector, and returns the results in a list. In other words, [map(1:3, f)](https://purrr.tidyverse.org/reference/map.html) is equivalent to [list(f(1), f(2), f(3))](https://rdrr.io/r/base/list.html).

[map()](https://purrr.tidyverse.org/reference/map.html) returns a list, which makes it the most general of the map family because you can put anything in a list. But it is inconvenient to return a list when a simpler data structure would do, so there are four more specific variants: [map\_lgl()](https://purrr.tidyverse.org/reference/map.html), [map\_int()](https://purrr.tidyverse.org/reference/map.html), [map\_dbl()](https://purrr.tidyverse.org/reference/map.html), and [map\_chr()](https://purrr.tidyverse.org/reference/map.html). Each returns an atomic vector of the specified type.

9.2.2 Anonymous functions and shortcuts

Instead of using [map()](https://purrr.tidyverse.org/reference/map.html) with an existing function, you can create an inline anonymous function

Can generate random data with this function

9.2.3 Passing arguments with ...

It’s often convenient to pass along additional arguments to the function that you’re calling. For example, you might want to pass na.rm = TRUE along to [mean()](https://rdrr.io/r/base/mean.html). One way to do that is with an anonymous function: