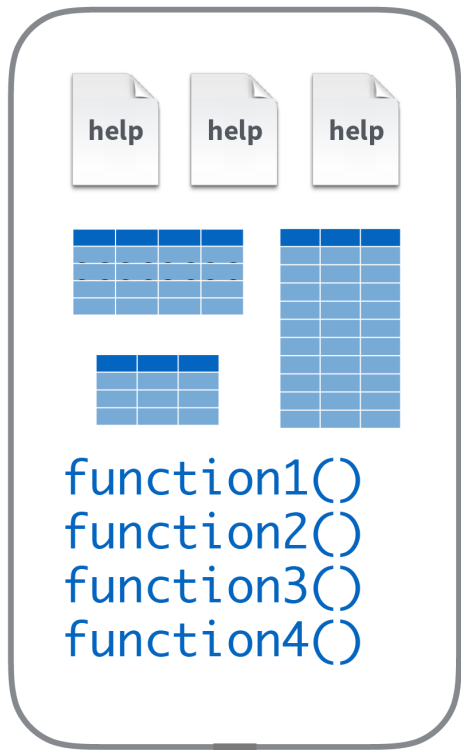


# Data basics

# Packages



help help help

function1()  
function2()  
function3()  
function4()

This diagram shows a package structure. At the top, there are three document icons labeled 'help'. Below them are two data file icons: a 3x4 grid and a 6x4 grid. At the bottom, there is a list of four functions: function1(), function2(), function3(), and function4().

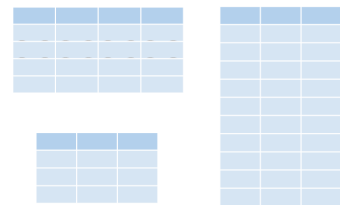


**Base R**



help help help

Three document icons labeled 'help'.



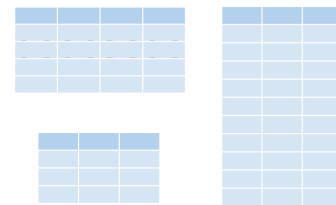
Two data file icons: a 3x4 grid and a 6x4 grid.

function5()  
function6()  
function7()  
function8()



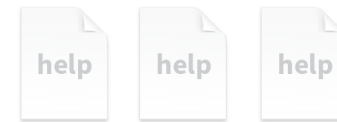
help help help

Three document icons labeled 'help'.



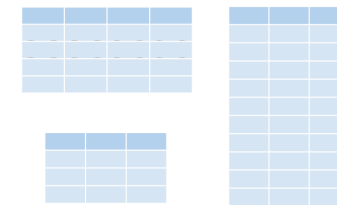
Two data file icons: a 3x4 grid and a 6x4 grid.

function9()  
functionA()  
functionB()  
functionC()



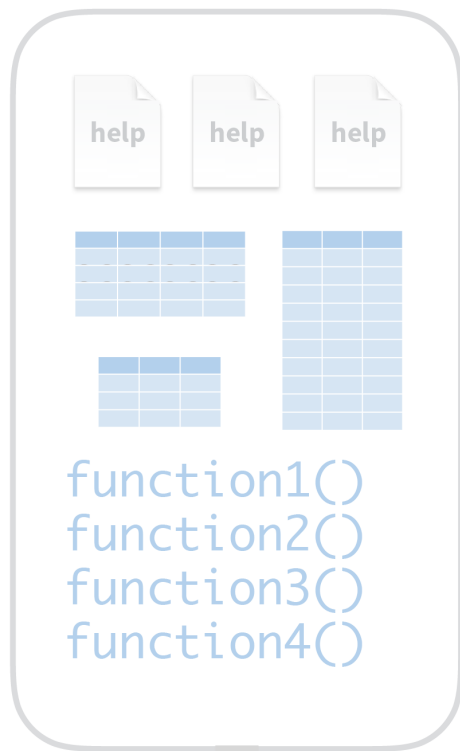
help help help

Three document icons labeled 'help'.

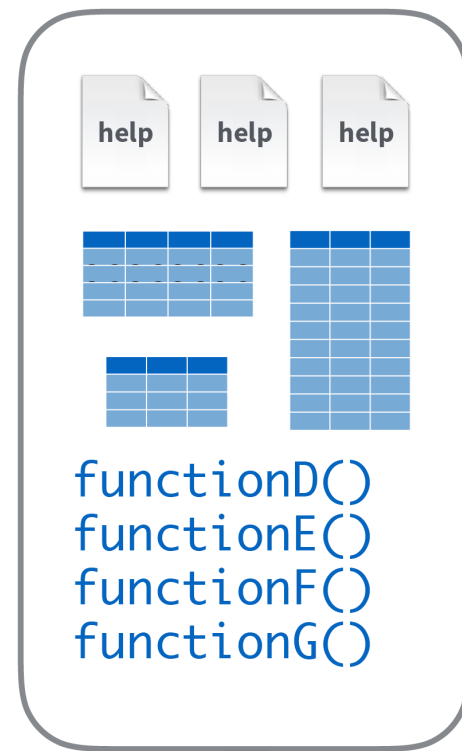
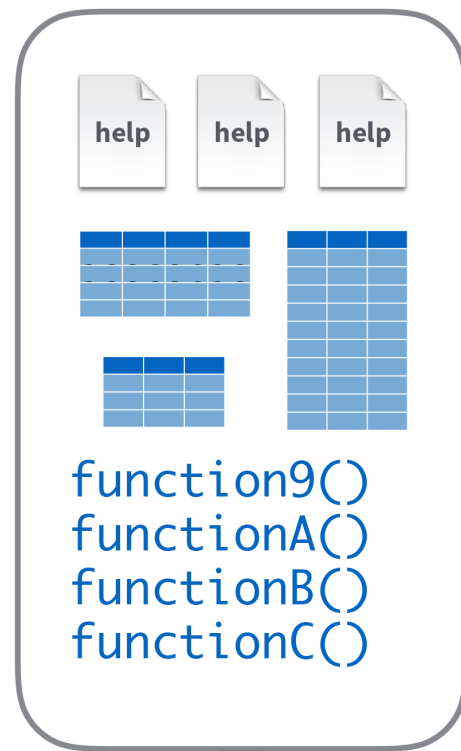
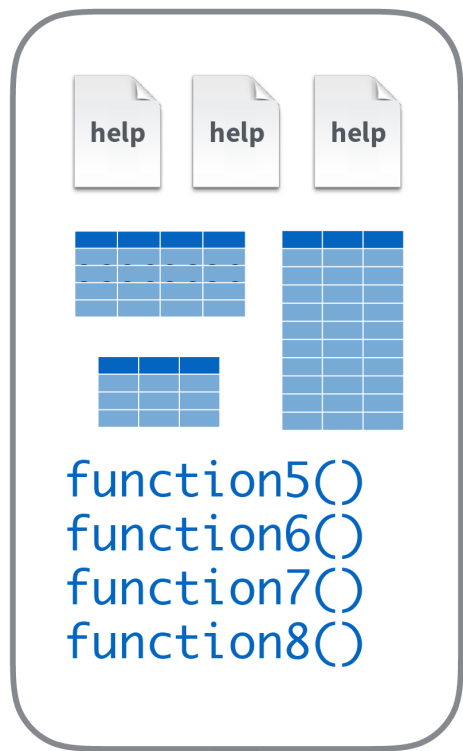


Two data file icons: a 3x4 grid and a 6x4 grid.

functionD()  
functionE()  
functionF()  
functionG()



Base R



R Packages

# Using packages

```
install.packages("name")
```

Downloads files  
to your computer

Do this once per computer

```
library(name)
```

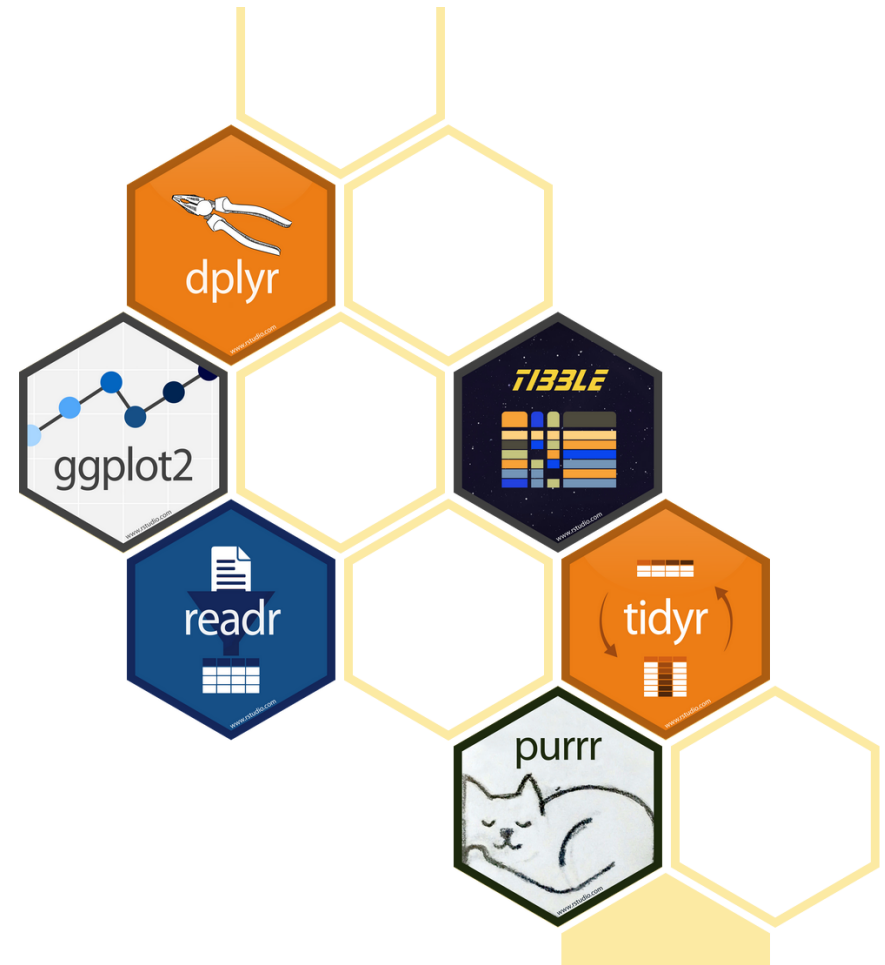
Loads the package

Do this once per R session

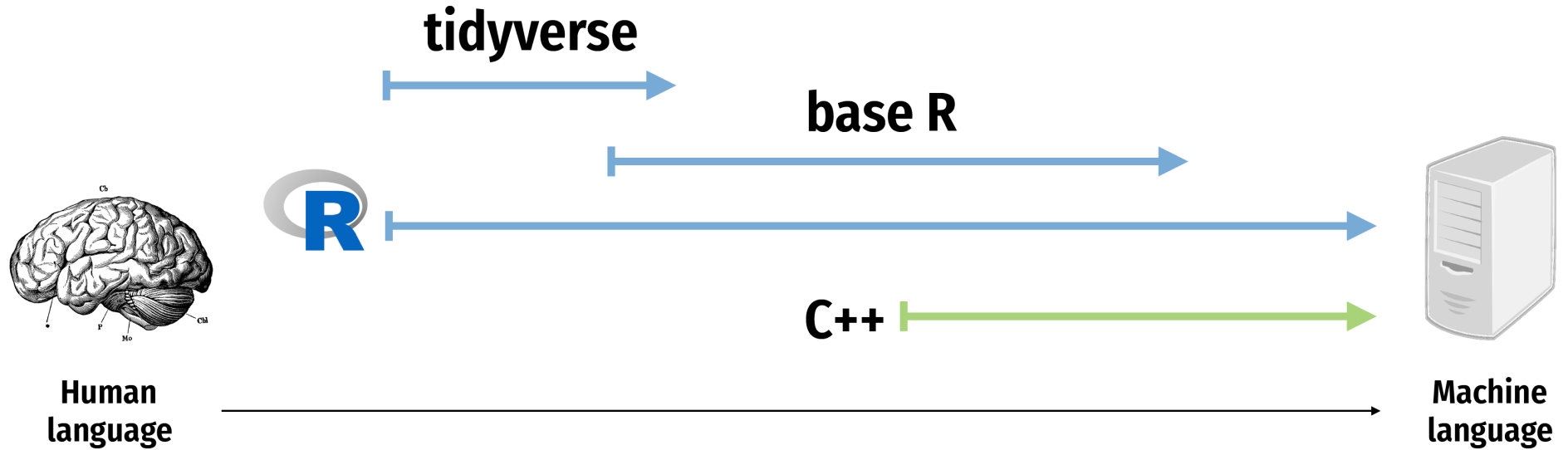
# The tidyverse

"The tidyverse is an opinionated collection of R packages designed for data science. All packages share an underlying design philosophy, grammar, and data structures."

... the tidyverse makes data science faster, easier and more fun...



# The tidyverse



# The tidyverse package

```
library(tidyverse)
```

The tidyverse package is a shortcut for installing and loading all the key tidyverse packages



```
install.packages("tidyverse")
```

```
install.packages("ggplot2")  
install.packages("dplyr")  
install.packages("tidyr")  
install.packages("readr")  
install.packages("purrr")  
install.packages("tibble")  
install.packages("stringr")  
install.packages("forcats")  
install.packages("lubridate")  
install.packages("hms")  
install.packages("feather")  
install.packages("haven")  
install.packages("httr")  
install.packages("jsonlite")  
install.packages("readxl")  
install.packages("rvest")  
install.packages("xml2")  
install.packages("modelr")  
install.packages("broom")
```

```
library(tidyverse)
```

```
library(ggplot2)  
library(dplyr)  
library(tidyr)  
library(readr)  
library(purrr)  
library(tibble)  
library(stringr)  
library(forcats)  
library(lubridate)
```

# Rectangular data

# Data frames and tibbles

Data frames are the most common kind of data objects; used for rectangular data (like spreadsheets)

Data frames: R's native data object

Tibbles (tbl): a fancier enhanced kind of data frame

(You really won't notice a difference today)

# Vectors and lists

# Vectors

Vectors are a list of values of the same type  
(all text, or all numbers, etc.)

Make them with `c()`:

```
c(1, 4, 2, 5, 7)
```

You'll usually want to assign them to something:

```
neat_numbers <- c(1, 4, 2, 5, 7)
```

# Basic data types

---

<b>Integer</b>	Whole numbers	<code>c(1, 2, 3, 4)</code>
<b>Double</b>	Numbers	<code>c(1, 2.4, 3.14, 4)</code>
<b>Character</b>	Text	<code>c("1", "blue", "fun", "monster")</code>
<b>Logical</b>	True or false	<code>c(TRUE, FALSE, TRUE, FALSE)</code>
<b>Factor</b>	Category	<code>c("Strongly disagree", "Agree", "Neutral")</code>

---

# Importing data

# Packages for importing data



Work with plain text data

```
my_data <-  
read_csv("file.csv")
```



Work with Excel files

```
my_data <-  
read_excel("file.xlsx")
```



Work with Stata, SPSS, and SAS data

```
my_data <-  
read_stata("file.dta")
```



# Other types of data

<b>Package</b>	<b>Type of data</b>
jsonlite	JSON data
xml2	XML data
httr2	Web APIs
rvest	Web pages (web scraping)
DBI	Databases (SQL)

# Next up

**Data visualization**