



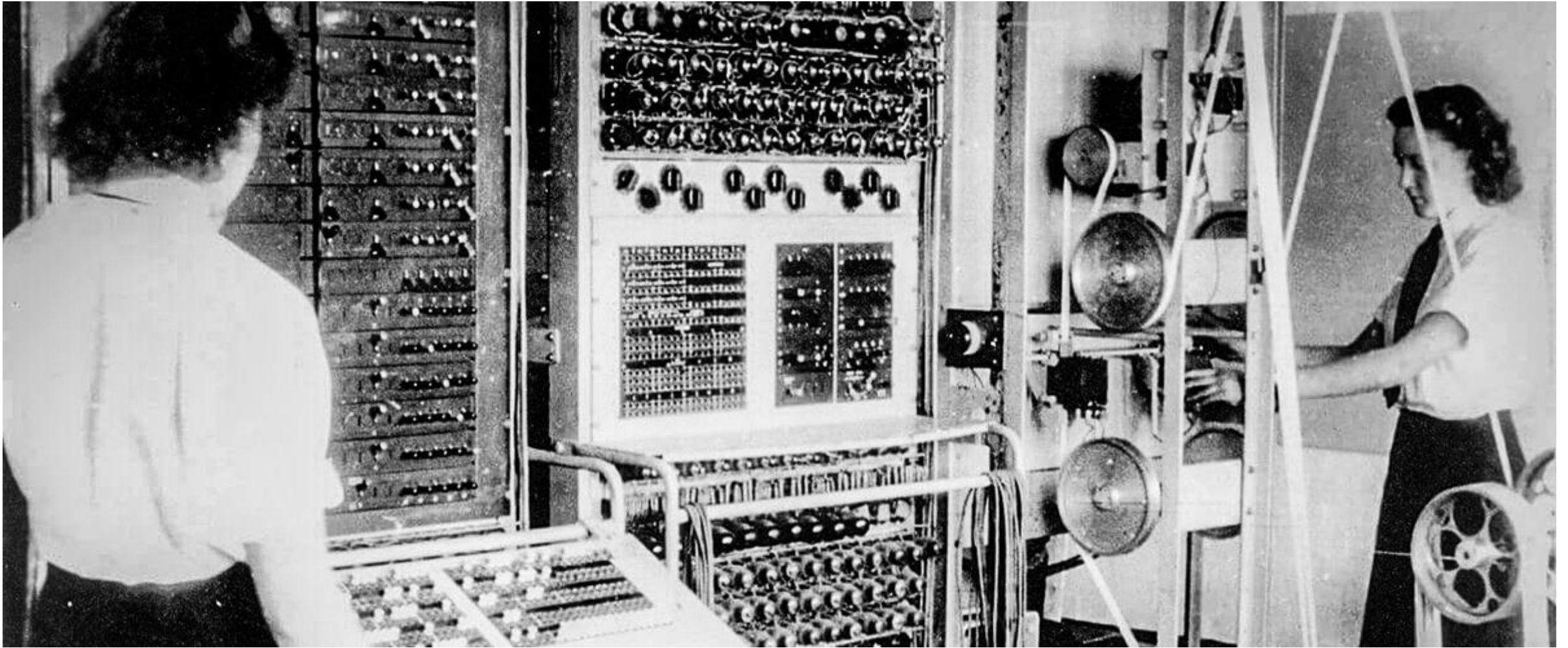
Introduction to R programming

NEUROSYSM930

Dimitrios Kleftogiannis

dimitrios.kleftogiannis@uib.no

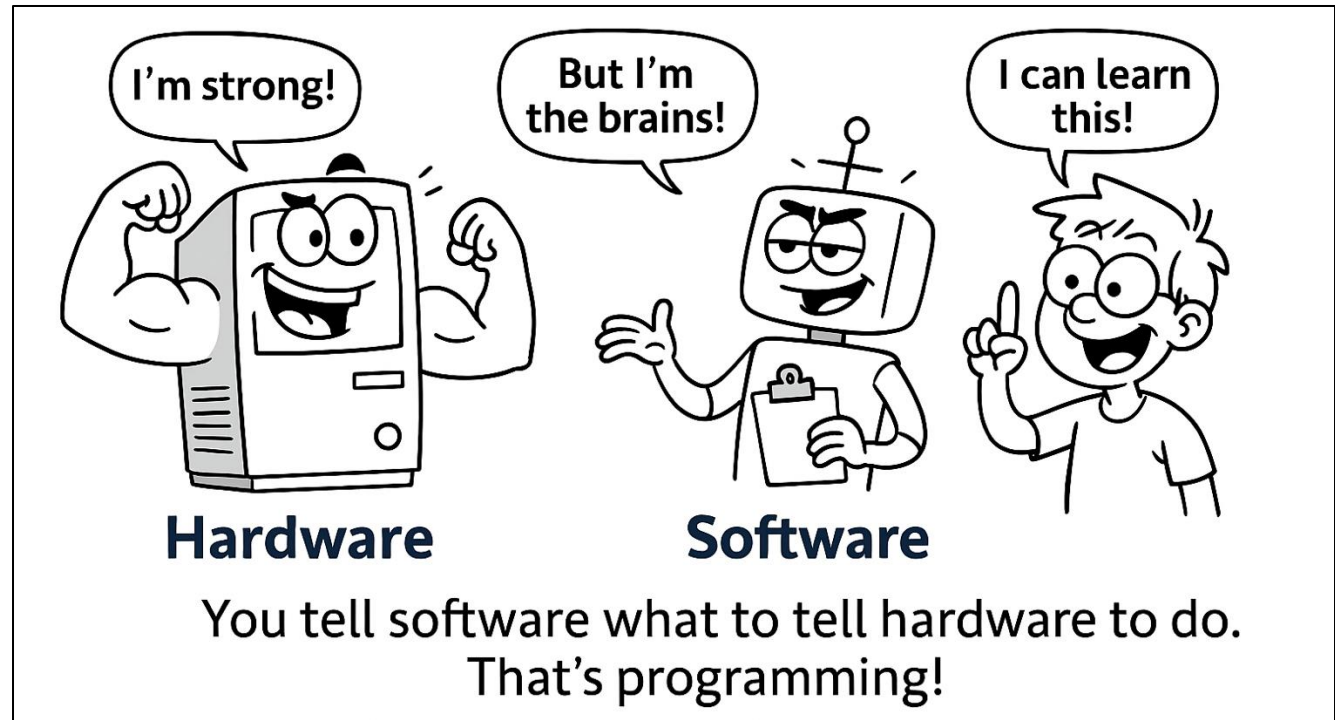
Computers and Programming



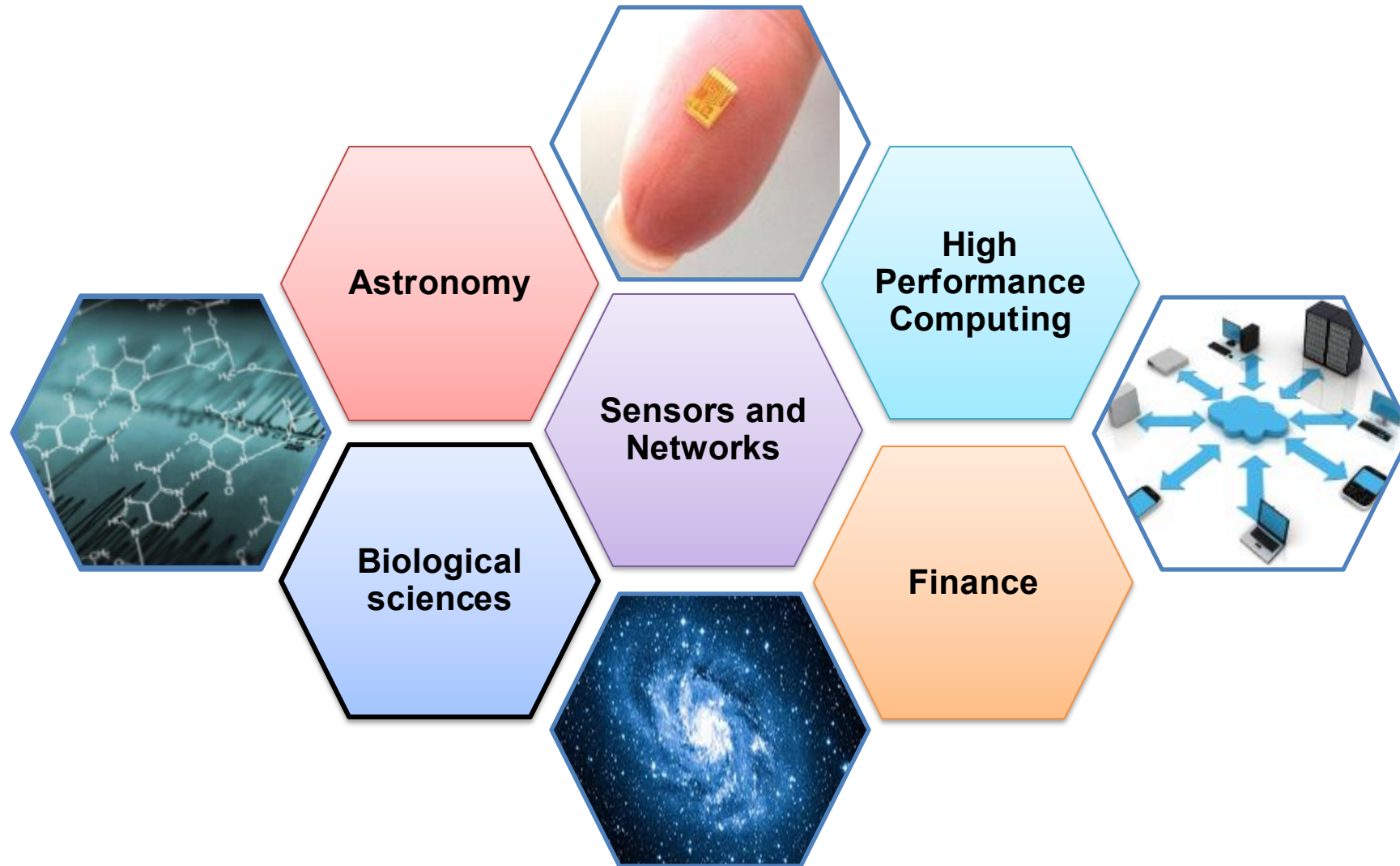
(c) Image taken from the www.
All rights reserved.

Computers and Programming

- ❖ A computer is just a machine (**hardware**) for executing programs (**software**)
- ❖ The process of creating a program is called **programming**, and it is the focus of this course
- ❖ **Anyone can learn programming**



Computers and programming are part of our life



Why to learn programming in clinical research ?



Perform patient cohort/
data exploration



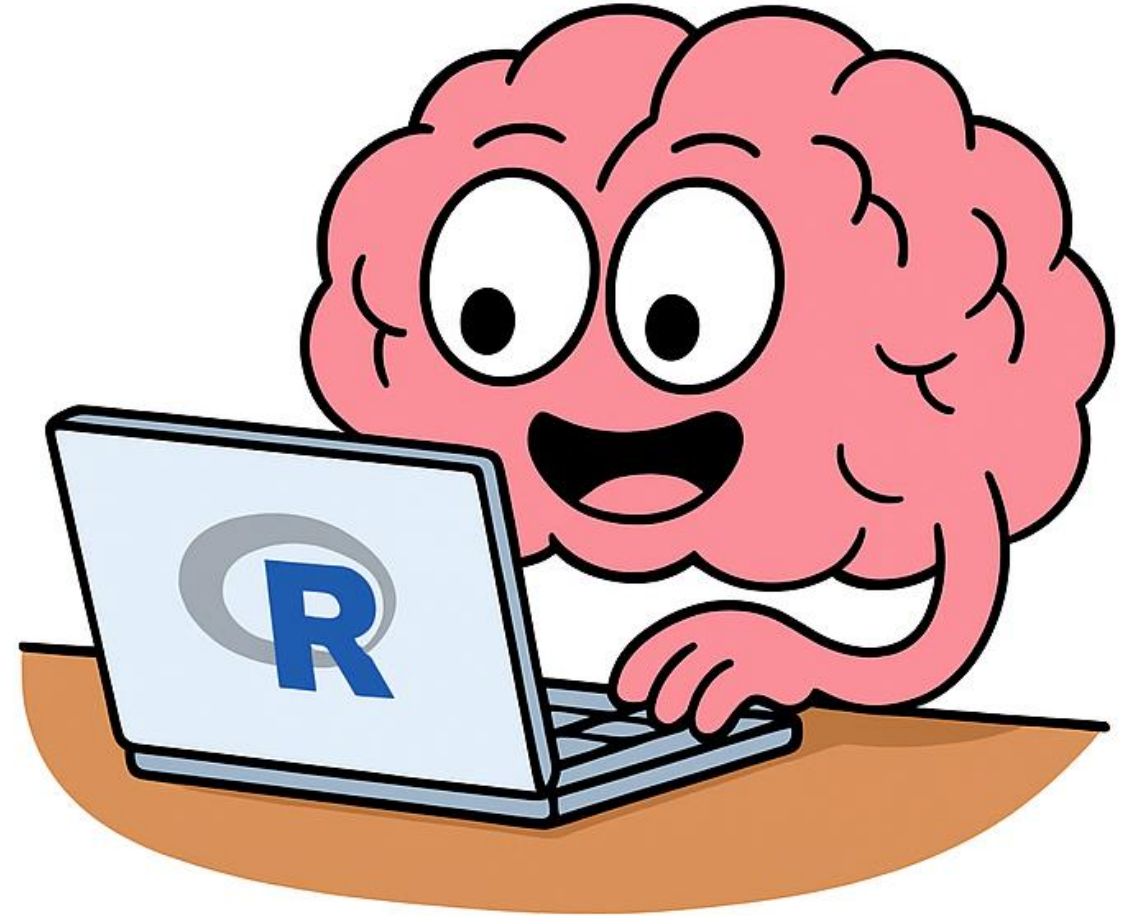
Compute p-values and
other statistics



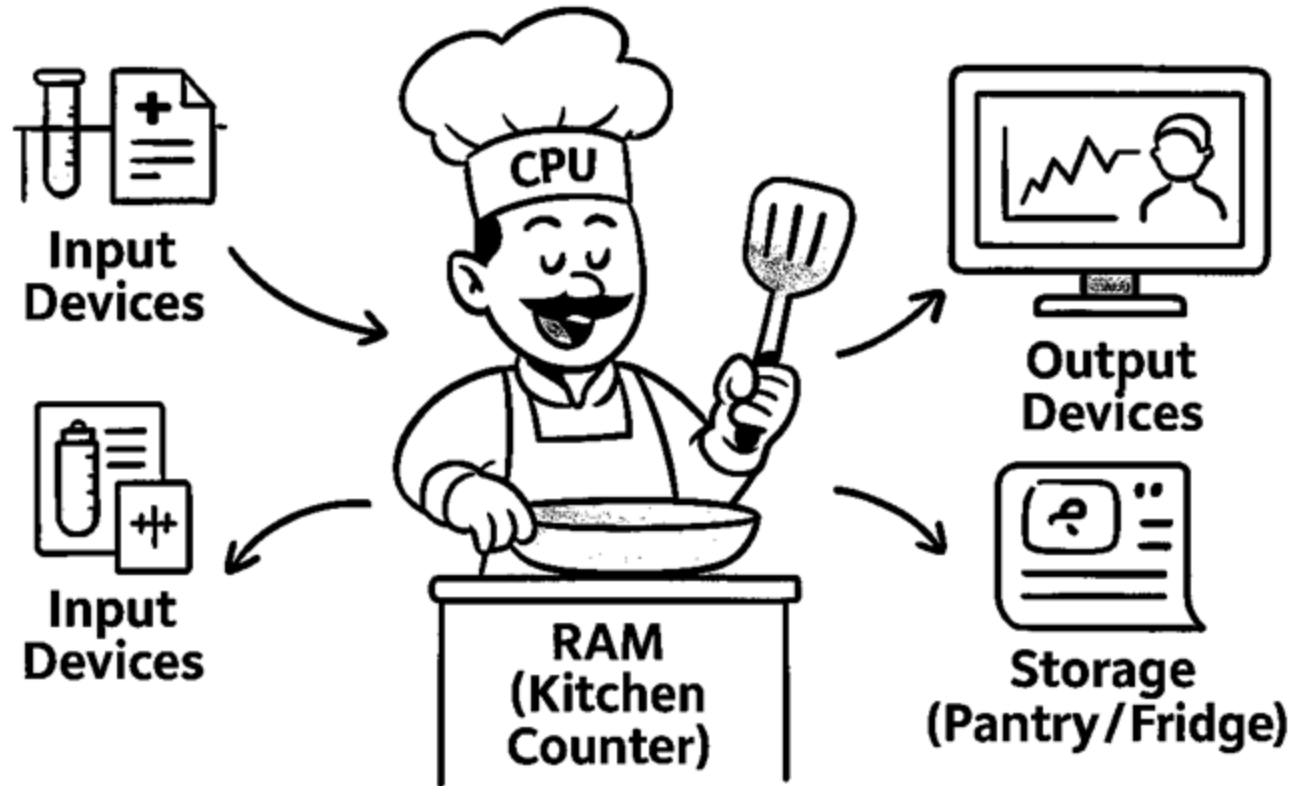
High-throughput
data analysis



Reproducible research



Computers explained simply



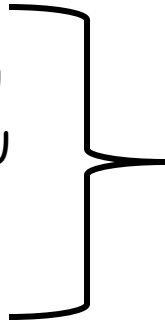
Hardware is the kitchen. Software is the recipe.
Programming = writing the recipe

Programming languages

- The hardware understands a **very low-level language** known as machine language which is translated to binary....

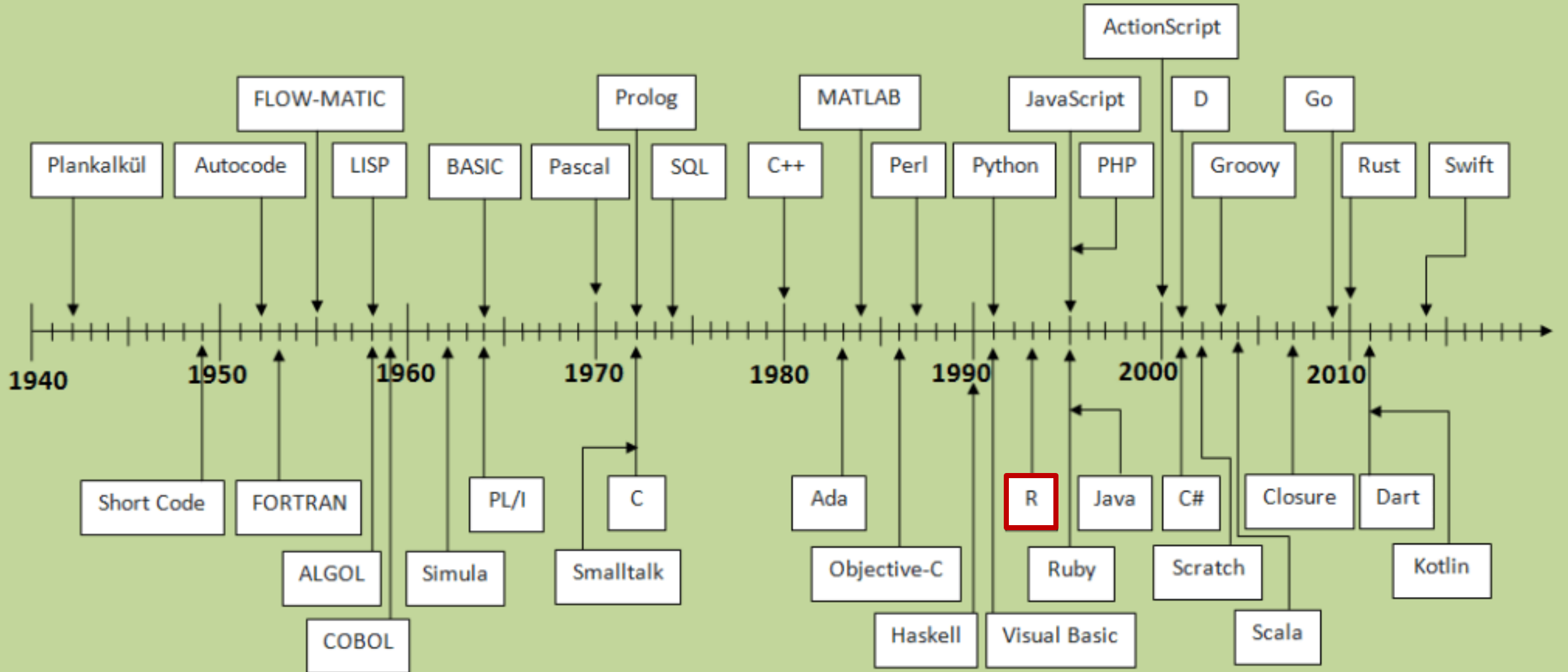
For example:

1. Load the number from memory location 2001 into the CPU
2. Load the number from memory location 2002 into the CPU
3. Add the two numbers in the CPU
4. Store the result into location 2003


$$A + B = C$$

- We need to use a **higher-level language and convert it** into a machine language that the hardware can execute...
 - **Compiled languages** → use a complex program called compiler
 - **Interpreted languages** → executes the code on the fly

Timeline of Programming Languages



(c) Image taken from <https://javaconceptoftheday.com/history-of-programming-languages/>

All rights reserved.



The brief history of R



“R is a programming language for statistical computing and data visualization. It has been adopted in the fields of data mining, bioinformatics, and data analysis” (from wiki)

- R was developed in 1993 by **R**oss Ihaka and **R**obert Gentleman at the University of Auckland
- It was based on the S statistical programming language
- R software is **open-source and free** software licensed by the GNU Project
- In 1998, the Comprehensive R Archive Network **CRAN** was established
 - ✓ Now CRAN has more than 22,000 packages

Explore the R architecture

R is the engine



RStudio is the dashboard



Use R from the terminal

```
(base) c02z46udlvdc:NEUROSYSMED930_R_workshop kleftogi$ R

R version 4.4.1 (2024-06-14) -- "Race for Your Life"
Copyright (C) 2024 The R Foundation for Statistical Computing
Platform: x86_64-apple-darwin20

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

> |
```

```
R Console

R version 4.4.1 (2024-06-14) -- "Race for Your Life"
Copyright (C) 2024 The R Foundation for Statistical Computing
Platform: x86_64-apple-darwin20

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

Natural language support but running in an English locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

During startup - Warning messages:
1: Setting LC_CTYPE failed, using "C"
2: Setting LC_COLLATE failed, using "C"
3: Setting LC_TIME failed, using "C"
4: Setting LC_MESSAGES failed, using "C"
5: Setting LC_MONETARY failed, using "C"
[R.app GUI 1.80 (8416) x86_64-apple-darwin20]

WARNING: You're using a non-UTF8 locale, therefore only ASCII characters will work.
Please read R for Mac OS X FAQ (see Help) section 9 and adjust your system preferences accordingly.
[Workspace restored from /Users/kleftogi/.RData]
[History restored from /Users/kleftogi/.Rapp.history]

2024-11-18 10:20:30.006 R[48455:10555424] +[IMKClient subclass]: chose IMKClient_Modern
2024-11-18 10:20:30.823 R[48455:10555424] +[IMKInputSession subclass]: chose IMKInputSession_Modern
> |
```

Use the R console

The R studio environment

The screenshot shows the R Studio interface with several key components highlighted by red boxes and arrows:

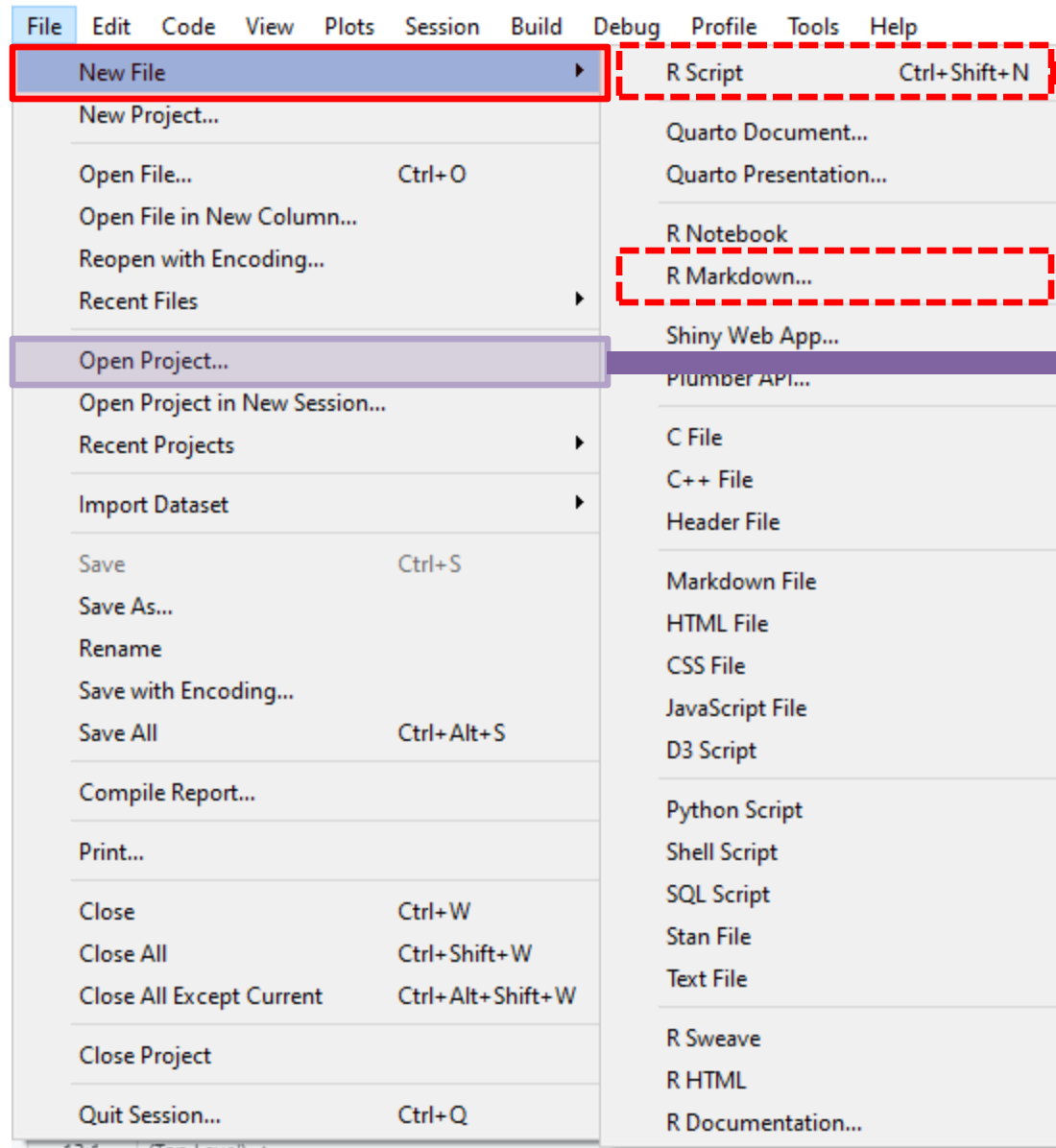
- Run Button:** A red box labeled "To run code, click on 'Run'" points to the "Run" button in the top toolbar.
- Code Editor:** A red box labeled "It is just a text editor. Write code here." points to the main editor area where code is written.
- Environment Panel:** A red box labeled "Object will occur here:" with a bulleted list of "Data" and "Functions, etc." points to the Environment panel on the right.
- Console:** A red box labeled "R console Code will be executed here." points to the Console panel at the bottom left.
- Bottom Panel:** A red box with a bulleted list of "File explorer (working directory)", "Plots", "Packages to be downloaded", and "Help" points to the bottom panel, which includes tabs for Files, Plots, Packages, Help, Viewer, and Presentation.

The R Studio interface includes a menu bar (File, Edit, Code, View, Plots, Session, Build, Debug, Profile, Tools, Help), a toolbar with icons for file operations and running code, and a status bar at the bottom showing the current file and session information.

What shall I do now ?

- As any other programming language R has its own syntax and semantics
- No buttons and clicking to perform computations...
- **But you need to learn how to program and you must invest time on that !**
Because...
 - ✓ You can do basically everything you need in your daily data wrangling life with only one software solution...
 - ✓ Statistics and machine learning
 - ✓ Small and big data exploration
 - ✓ Great visualization capabilities
 - ✓ Develop web-apps even databases and more complex workflows
- R packages are essentially free R programs that provide extra functionalities
 - ✓ Need to be downloaded and installed

Different ways to create code



New text editor (to write code)

Another type of text editor great for data presentation

Includes several files and generates a nice working space that:

- ✓ You can copy the folder anywhere
- ✓ You can share the whole folder with colleagues

R Scripts vs. R Markdowns

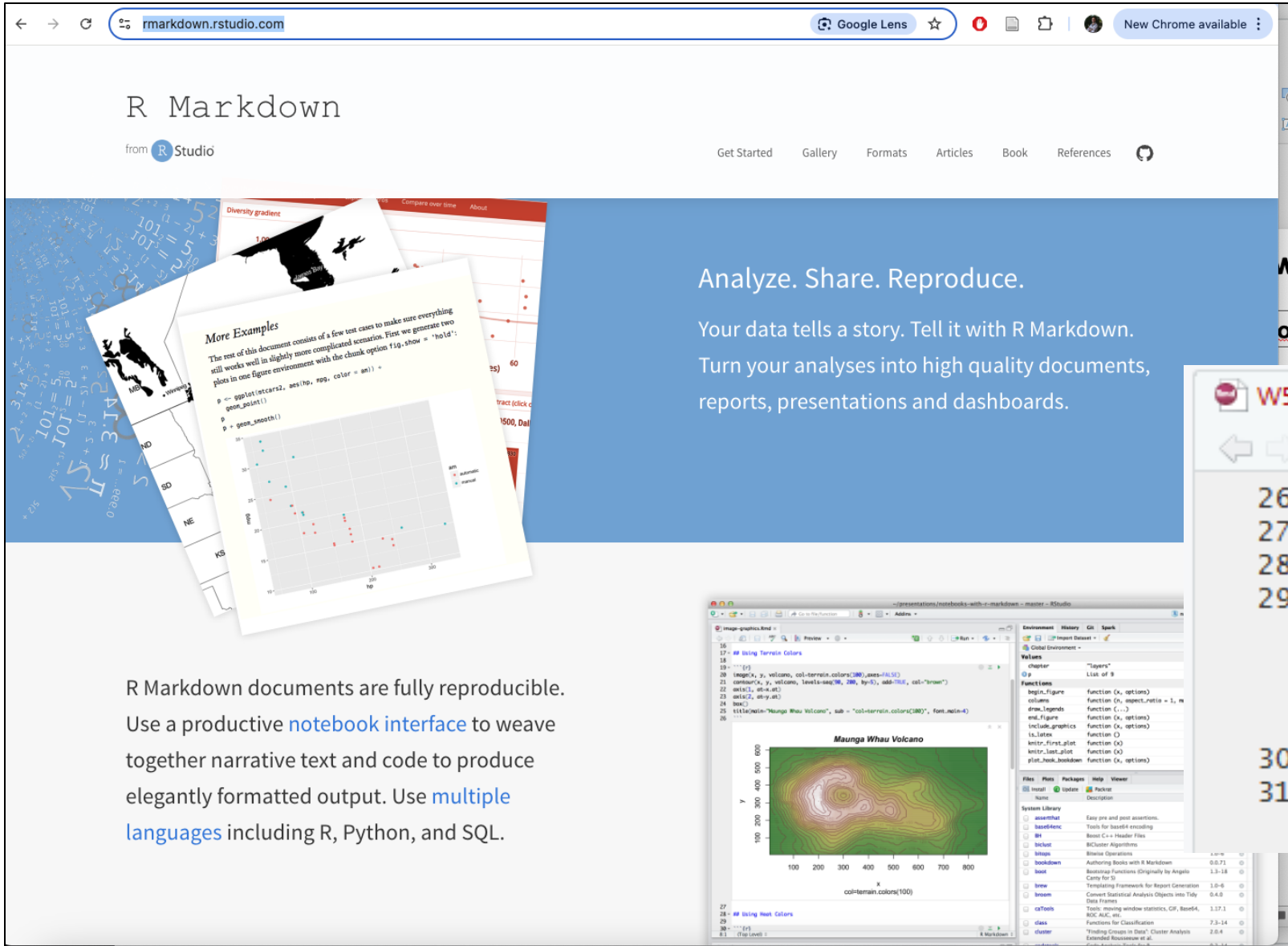
```
1 # this is my first R script
2 # do some things
3 x = 34
4 y = 16
5 z = x + y # addition
6 w = y/x   # division
7 # display the results
8 x
9 y
10 z
11 w
12 # change x
13 x = "some text"
14 # display the results
15 x
16 y
17 z
18 w
19
```

1:1 (Top Level) R Script

Console ~/Documents/R/MATH-3200/

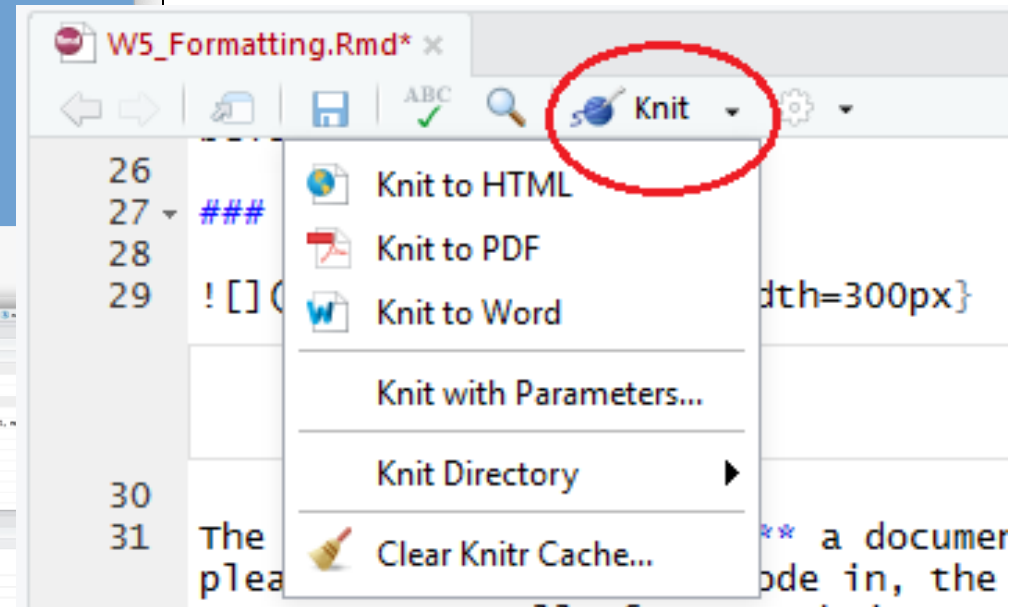
```
> z = x + y
~ w = y/x
```

R Scripts vs. R Markdowns



The top part of the image shows the rmarkdown.rstudio.com website. The header includes the title "R Markdown" and the "from R Studio" logo. Navigation links for "Get Started", "Gallery", "Formats", "Articles", "Book", and "References" are present. The main content area features the text "Analyze. Share. Reproduce." and "Your data tells a story. Tell it with R Markdown. Turn your analyses into high quality documents, reports, presentations and dashboards." Below this, there is a collage of R Markdown documents, including one titled "More Examples" which shows a scatter plot and R code. The bottom part of the image shows a screenshot of the RStudio interface with an R Markdown document open. The document contains R code for plotting a map of Maunga Whau Volcano. The RStudio interface also shows the "Environment" and "Files" panels.

R Markdown documents are fully reproducible. Use a productive [notebook interface](#) to weave together narrative text and code to produce elegantly formatted output. Use [multiple languages](#) including R, Python, and SQL.



Basic R concepts and terminology

- **Console** → where you enter in commands
- **Run** → the act of telling R to perform a computation by giving it commands in the console
- **Objects** → where values are saved in R, check the **Environment panel**
- **Data types** → integers, doubles/numerics, logicals, and characters.

Errors, warnings and messages

```
> dat <- read.csv("data.csv")
```

```
Error in file
```

```
In addition
```

```
In file(fil
```

```
cannot op
```

```
> |
```

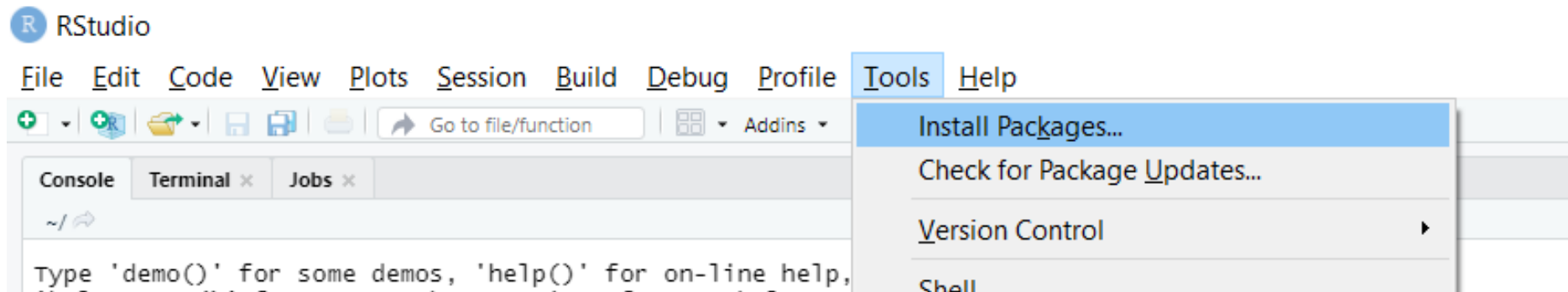
```
1:1 (Top Level) R Script
Console Terminal x Jobs x
~/
> library("dplyr")
Attaching p
The followi
between
The followi
filter,
The followi
interse
Warning message:
package 'dplyr' was built under R version 3.6.2
```



oup"

Packages

- Different ways to install packages exist



Install Bioconductor Packages

To install core packages, type the following in an R command window:

```
if (!require("BiocManager", quietly = TRUE))
  install.packages("BiocManager")
BiocManager::install(version = "3.20")
```

To install core packages, type the following in an R command window:

```
BiocManager::install(c("GenomicFeatures", "AnnotationDbi"))
```

The `install()` function (in the `BiocManager` namespace) has arguments that change its default behavior type `BiocManager::install` for further help.

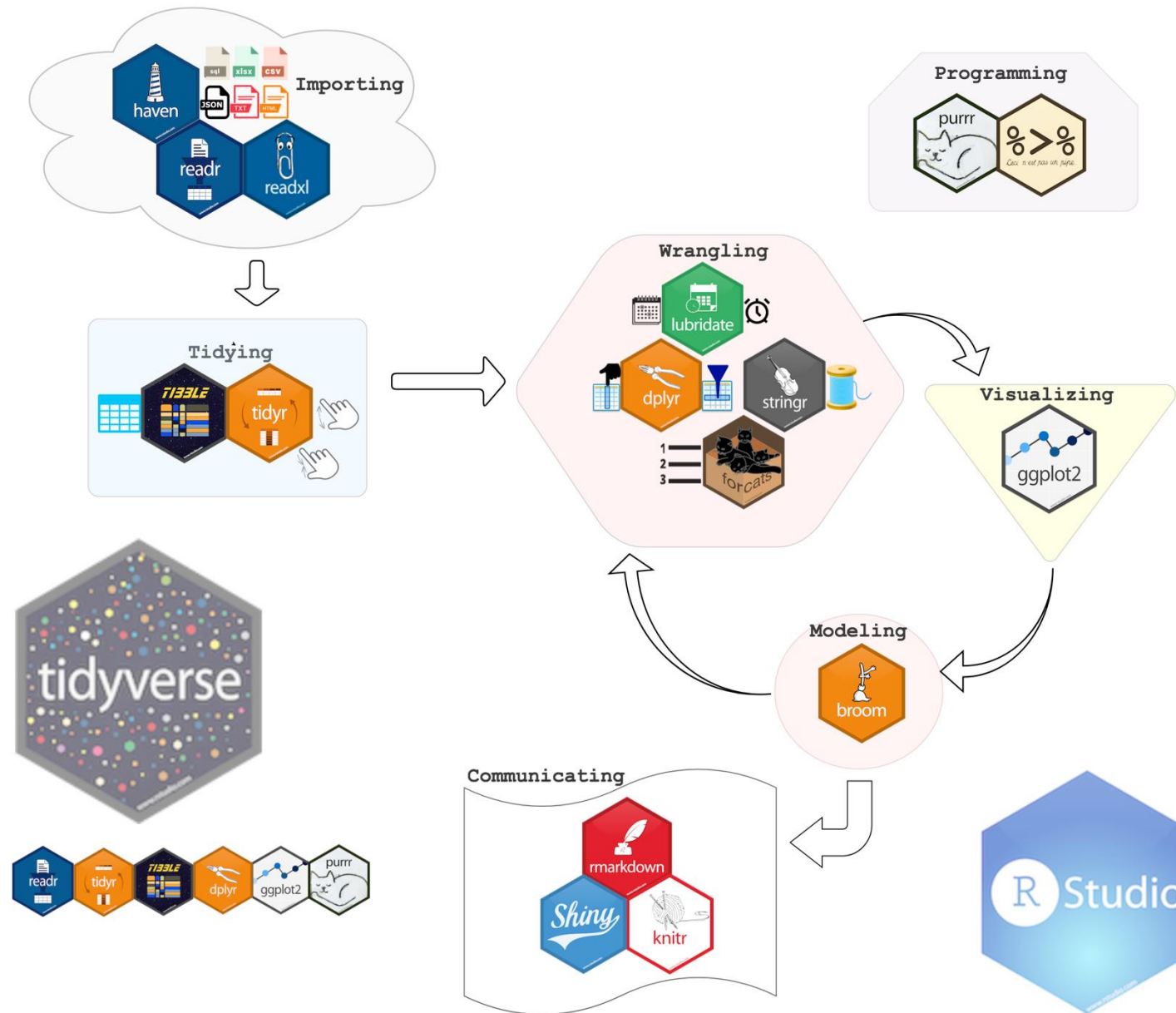
Packages

- Never forget to load the installed packages....

A screenshot of an R console window. The window has a title bar with a green 'R' icon. Below the title bar, there are three circular icons: a copy icon, a paste icon, and a moon icon. The main area of the console is a dark gray bar containing the text `library("ggplot2")` in a light blue font.

And never forget that R is distributed with fifteen "base packages": *base, compiler, datasets, grDevices, graphics, grid, methods, parallel, splines, stats, stats4, tcltk, tools, translations, and utils*

By combining R packages, we create workflows





(c) Image from the www
All rights reserved.

Take home message

Nobody expects you to become an advanced programmer just because you attended this course.

You need to invest time on programming.

So practice, practice and practice...

During the practical session, we will just illustrate few out of many things you can do with R



Thanks for your attention !

Q&A

dimitrios.kleftogiannis@uib.no

<https://www.uib.no/en/persons/Dimitrios.Kleftogiannis>

