This template demonstrates many of the bells and whistles of the `reprex::reprex_document()` output format. The YAML sets many options to non-default values, such as using `#;-)` as the comment in front of output.

Code style

Since `style` is `TRUE`, this difficult-to-read code (look at the `.Rmd` source file) will be restyled according to the Tidyverse style guide when it's rendered. Whitespace rationing is not in effect!

```
x=1;y=2;z=x+y;z
```

```
## [1] 3
```

Quiet tidyverse

The tidyverse meta-package is quite chatty at startup, which can be very useful in exploratory, interactive work. It is often less useful in a reprex, so by default, we suppress this.

However, when `tidyverse_quiet` is `FALSE`, the rendered result will include a tidyverse startup message about package versions and function masking.

```
library(tidyverse)
```

Chunks in languages other than R

Remember: knitr supports many other languages than R, so you can reprex bits of code in Python, Ruby, Julia, C++, SQL, and more. Note that, in many cases, this still requires that you have the relevant external interpreter installed.

Let's try Python!

```
x = 'hello, python world!'
print(x.split(' '))
```

```
## ['hello,', 'python', 'world!']
```

And bash!

```
echo "Hello Bash!";
pwd;
ls | head;
```

```
## Hello Bash!
## /Users/enricopirani/Statistica
## Starg.ipynb
## Statistic_without_math.Rmd
## Statistica.Rproj
## Statistica_Python.ipynb
## Statistica_Python.py
## Tufte.Rmd
## cripto_update.Rmd
## cripto_update.html
## cripto_update.log
## cripto_update.pdf
```

Write a function in C++, use Rcpp to wrap it and ...

```
#include <Rcpp.h>
using namespace Rcpp;
```

```
// [[Rcpp::export]]
NumericVector timesTwo(NumericVector x) {
  return x * 2;
}
```

then immediately call your C++ function from R!

```
timesTwo(1:4)
```

```
## [1] 2 4 6 8
```

Standard output and error

Some output that you see in an interactive session is not actually captured by rmarkdown, when that same code is executed in the context of an `.Rmd` document. When `std_out_err` is `TRUE`, `reprex::reprex_render()` uses a feature of `callr:r()` to capture such output and then injects it into the rendered result.

Look for this output in a special section of the rendered document (and notice that it does not appear right here).

```
system2("echo", args = "Output that would normally be lost")
```

Session info

Because `session_info` is `TRUE`, the rendered result includes session info, even though no such code is included here in the source document.

Standard output and standard error

`/Users/enricopirani/Statistica/tutorial_Rmd_std_out_err.txt`

Session info

```
sessioninfo::session_info()
```

```
## - Session info ---------------------------------------------------------------
##  setting  value
##  version  R version 4.2.3 (2023-03-15)
##  os       macOS Big Sur ... 10.16
##  system   x86_64, darwin17.0
##  ui       X11
##  language (EN)
##  collate  en_US.UTF-8
##  ctype    en_US.UTF-8
##  tz       Europe/Rome
##  date     2024-03-10
##  pandoc   3.1.11.1 @ /usr/local/bin/ (via rmarkdown)
##
## - Packages -------------------------------------------------------------------
##  package     * version date (UTC) lib source
##  cli           3.6.2   2023-12-11 [1] CRAN (R 4.2.0)
##  digest        0.6.33  2023-07-07 [1] CRAN (R 4.2.0)
##  evaluate      0.23    2023-11-01 [1] CRAN (R 4.2.0)
##  fastmap       1.1.1   2023-02-24 [1] CRAN (R 4.2.0)
##  htmltools     0.5.7   2023-11-03 [1] CRAN (R 4.2.0)
##  knitr         1.45    2023-10-30 [1] CRAN (R 4.2.0)
##  Rcpp          1.0.12  2024-01-09 [1] CRAN (R 4.2.3)
##  rlang         1.1.2   2023-11-04 [1] CRAN (R 4.2.0)
```

```
##  rmarkdown     2.25    2023-09-18 [1] CRAN (R 4.2.0)
##  rstudioapi    0.15.0  2023-07-07 [1] CRAN (R 4.2.0)
##  sessioninfo   1.2.2   2021-12-06 [1] CRAN (R 4.2.0)
##  xfun          0.41    2023-11-01 [1] CRAN (R 4.2.0)
##  yaml          2.3.8   2023-12-11 [1] CRAN (R 4.2.0)
##
##  [1] /Users/enricopirani/Library/R/x86_64/4.2/library
##  [2] /Library/Frameworks/R.framework/Versions/4.2/Resources/library
##
## ------------------------------------------------------------------------------
```