

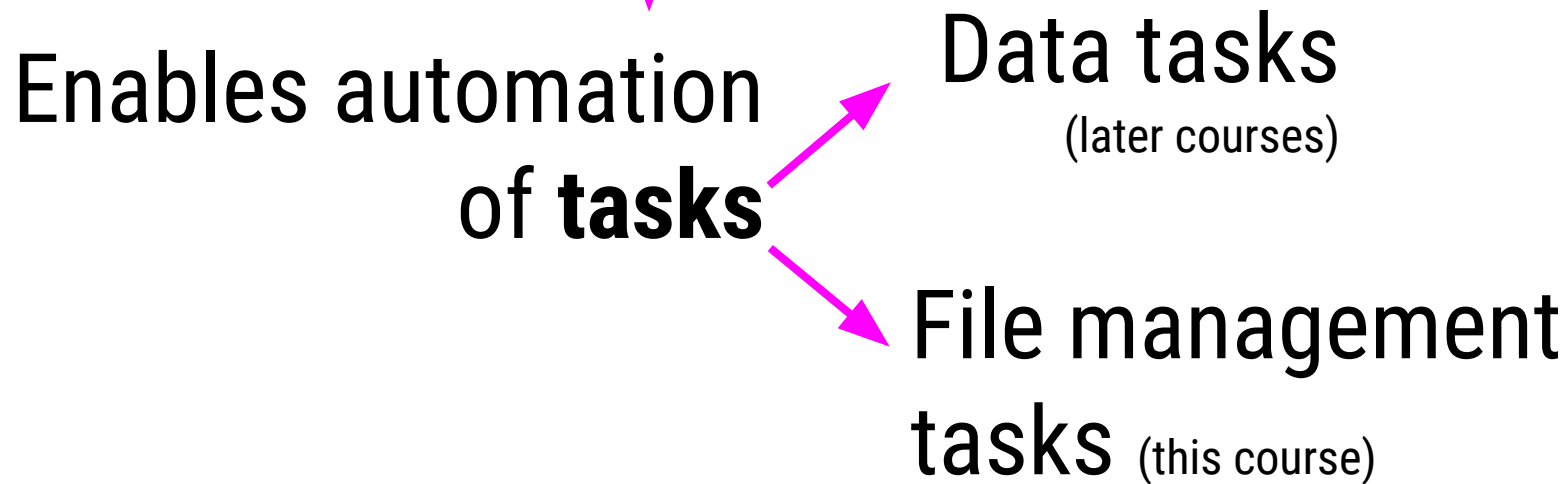
Managing Files in R



Organizing Data Science Projects

What is R?

R is a **programming language**.



Why manage files in R?

R is a **programming language**.



Complex automation of tasks

Complete documentation of work



File Edit Code View Plots Session Build Debug Profile Tools Help

Go to file/function

Addins

R 3.4.4

Console Terminal

/cloud/project/

R version 3.4.4 (2018-03-15) -- "Someone to Lean On"
Copyright (C) 2018 The R Foundation for Statistical Computing
Platform: x86_64-pc-linux-gnu (64-bit)

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.

> |

Environment History Connections

Import Dataset

Global Environment

Environment is empty

Files Plots Packages Help Viewer

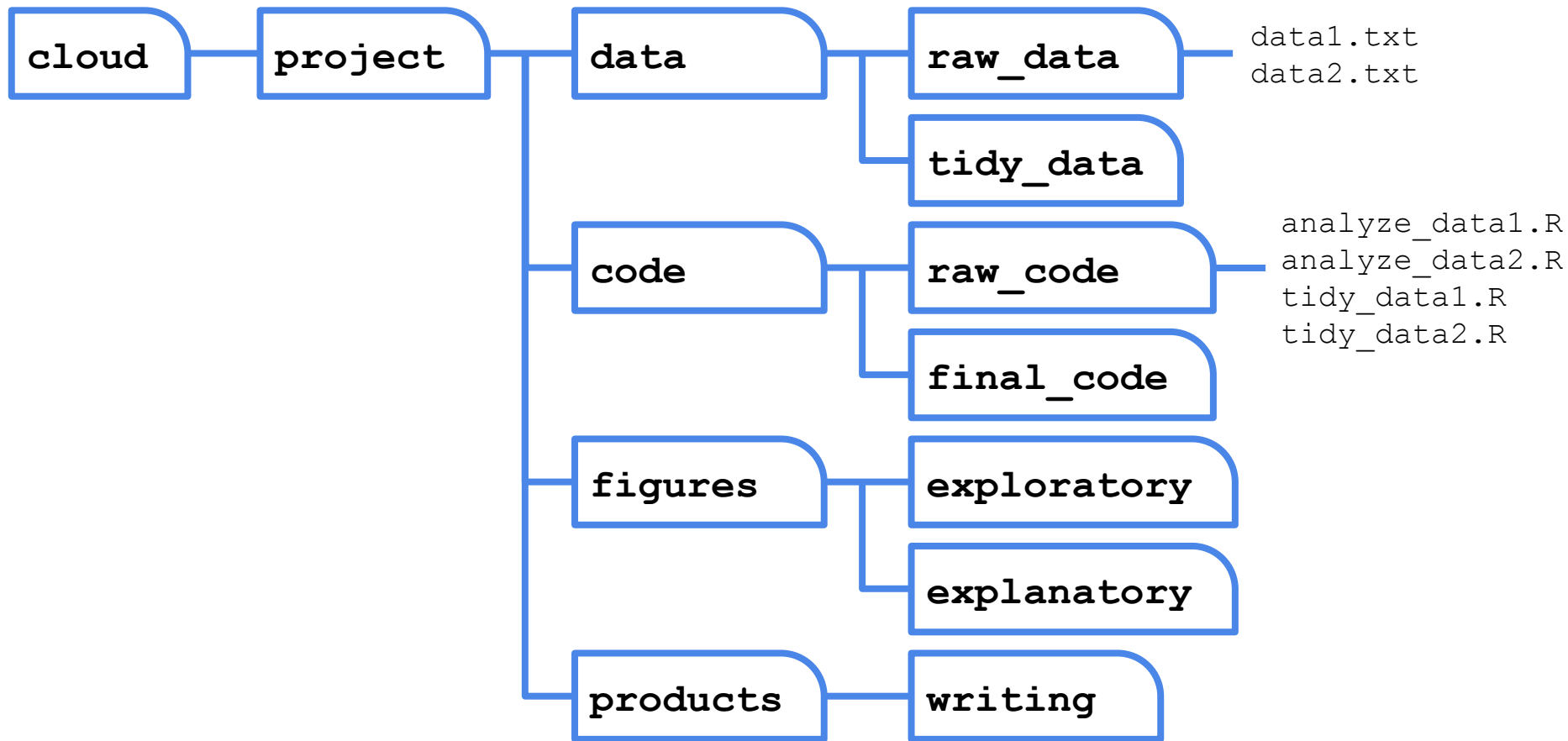
New Folder Upload Delete Rename More

/ > cloud > project

	Name	Size	Modified
	..		
<input type="checkbox"/>	.Rhistry	4 B	May 16, 2018, 10:43 PM
<input type="checkbox"/>	code		
<input type="checkbox"/>	data		
<input type="checkbox"/>	figures		
<input type="checkbox"/>	products		
<input type="checkbox"/>	project.Rproj	205 B	May 16, 2018, 10:43 PM

R prompt





Important Functions



Get current working directory `getwd()`

Terminal equivalent: `pwd`

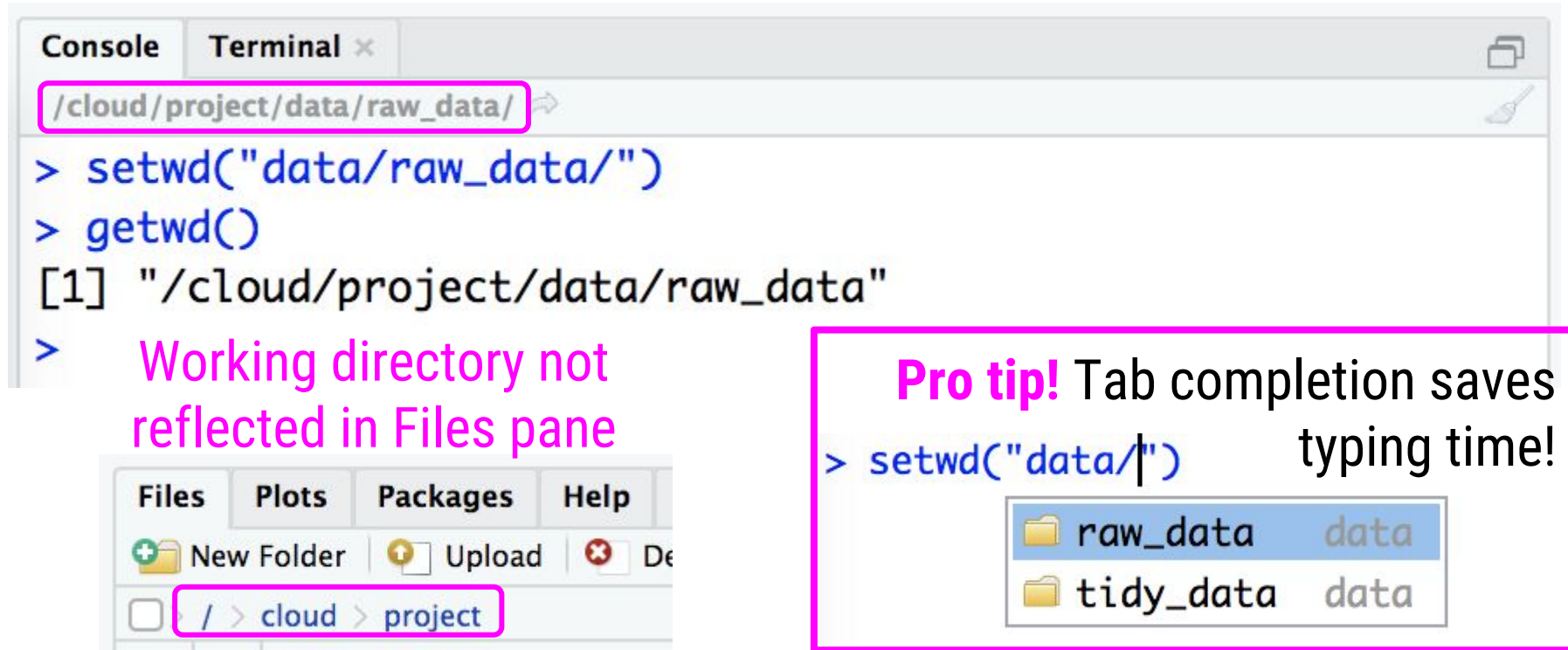


The screenshot shows a RStudio console window with two tabs: 'Console' and 'Terminal'. The 'Terminal' tab is active, showing the current directory as '/cloud/project/'. Below the terminal path, the R console shows the command `> getwd()` being executed, followed by the output `[1] "/cloud/project"`. The prompt `> |` is visible at the bottom of the console.

```
Console Terminal x
/cloud/project/
> getwd()
[1] "/cloud/project"
> |
```

Change current working directory `setwd()`

Terminal equivalent: `cd`



Console **Terminal** x

```
/cloud/project/data/raw_data/
```

```
> setwd("data/raw_data/")  
> getwd()  
[1] "/cloud/project/data/raw_data"  
>
```

Files **Plots** **Packages** **Help**

New Folder Upload De

/ > cloud > project

Pro tip! Tab completion saves typing time!

```
> setwd("data/|")
```

raw_data	data
tidy_data	data

List directory contents `list.files()`

Terminal equivalent: `ls`

```
Console Terminal x
/cloud/project/data/raw_data/

> list.files()
[1] "data1.txt" "data2.txt"
> list.files(".")
[1] "data1.txt" "data2.txt"
> list.files("../..code/raw_code/")
[1] "analyze_data1.R" "analyze_data2.R"
[3] "tidy_data1.R"    "tidy_data2.R"
> list.files("../..code/raw_code/", full.names = TRUE)
[1] "../..code/raw_code//analyze_data1.R"
[2] "../..code/raw_code//analyze_data2.R"
[3] "../..code/raw_code//tidy_data1.R"
[4] "../..code/raw_code//tidy_data2.R"
>
```

Full relative paths

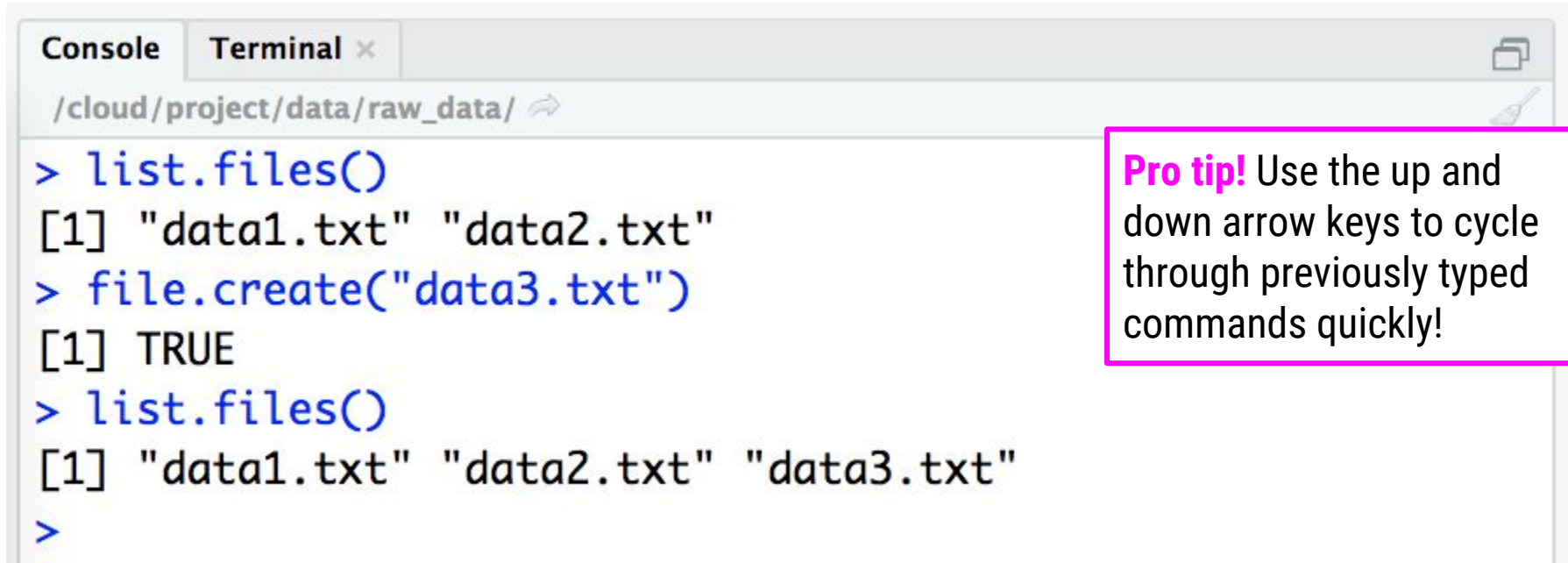
No double slashes

```
> list.files("../..code/raw_code", full.names = TRUE)
[1] "../..code/raw_code/analyze_data1.R"
[2] "../..code/raw_code/analyze_data2.R"
[3] "../..code/raw_code/tidy_data1.R"
[4] "../..code/raw_code/tidy_data2.R"
```



Create a file `file.create()`

Terminal equivalent: `touch`



The screenshot shows a terminal window with two tabs: 'Console' and 'Terminal'. The 'Terminal' tab is active, and the current directory is `/cloud/project/data/raw_data/`. The terminal shows the following commands and output:

```
> list.files()
[1] "data1.txt" "data2.txt"
> file.create("data3.txt")
[1] TRUE
> list.files()
[1] "data1.txt" "data2.txt" "data3.txt"
>
```

A pink-bordered box on the right side of the terminal window contains the following text:

Pro tip! Use the up and down arrow keys to cycle through previously typed commands quickly!

Moving and renaming a file `file.rename()`

Terminal equivalent: `mv`

```
Console Terminal x
/cloud/project/data/raw_data/ ↗
> list.files()
[1] "data1.txt" "data2.txt" "data3.txt"
> file.rename("data3.txt", "../data3.txt")
[1] TRUE
> list.files()
[1] "data1.txt" "data2.txt"
> list.files("../")
[1] "data3.txt" "raw_data"  "tidy_data"
> file.rename("../data3.txt", "data3_renamed.txt")
[1] TRUE
> list.files("../")
[1] "raw_data" "tidy_data"
> list.files()
[1] "data1.txt"      "data2.txt"
[3] "data3_renamed.txt"
>
```

Example 1: Move a file by keeping the same name

Example 2: Move and rename simultaneously



Copy a file `file.copy()`

Terminal equivalent: `cp`

```
Console Terminal x
/cloud/project/data/raw_data/
> list.files()
[1] "data1.txt"          "data2.txt"
[3] "data3_renamed.txt"
> file.copy("data3_renamed.txt", "data3.txt")
[1] TRUE
> list.files()
[1] "data1.txt"          "data2.txt"
[3] "data3_renamed.txt"  "data3.txt"
> file.copy("data3_renamed.txt", "data3.txt")
[1] FALSE
> file.copy("data3_renamed.txt", "data3.txt", overwrite = TRUE)
[1] TRUE
> list.files()
[1] "data1.txt"          "data2.txt"
[3] "data3_renamed.txt"  "data3.txt"
> |
```

Example 1: Copy a file and rename

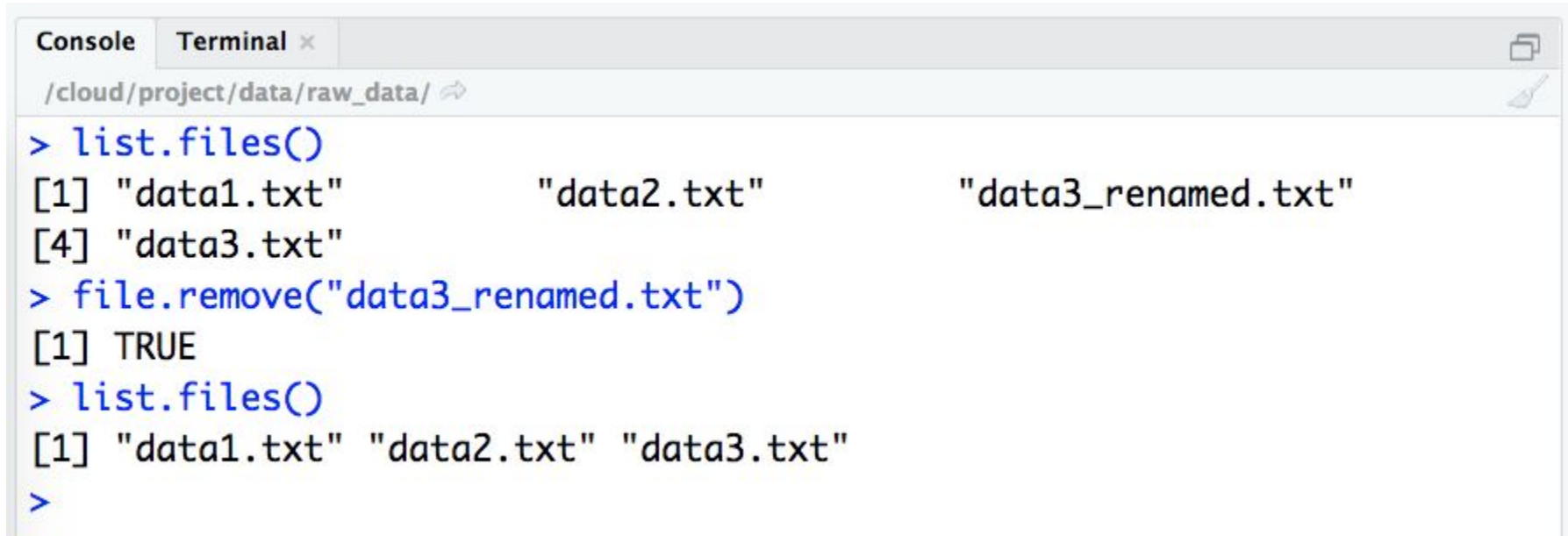
Example 2: Copying fails because `data3.txt` already exists

Example 3: Explicitly overwrite



Remove a file `file.remove()`

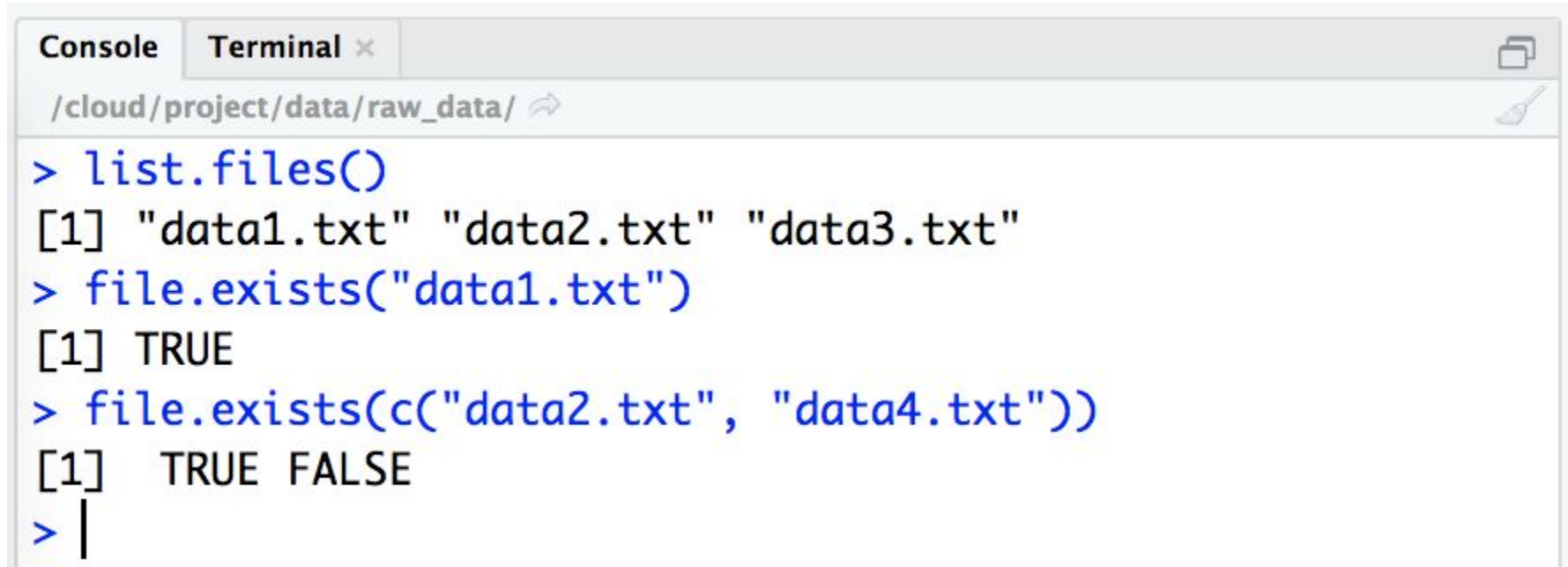
Terminal equivalent: `rm`

A screenshot of a RStudio Terminal window. The window has a title bar with 'Console' and 'Terminal x' tabs. The current directory is '/cloud/project/data/raw_data/'. The terminal shows the following commands and output:

```
> list.files()
[1] "data1.txt"          "data2.txt"          "data3_renamed.txt"
[4] "data3.txt"
> file.remove("data3_renamed.txt")
[1] TRUE
> list.files()
[1] "data1.txt" "data2.txt" "data3.txt"
>
```

Does a file exist? `file.exists()`

Terminal equivalent: `ls` + verification



The screenshot shows a RStudio Terminal window with the following content:

```
Console | Terminal x | /cloud/project/data/raw_data/
> list.files()
[1] "data1.txt" "data2.txt" "data3.txt"
> file.exists("data1.txt")
[1] TRUE
> file.exists(c("data2.txt", "data4.txt"))
[1] TRUE FALSE
> |
```

Summary



Terminal	R
<code>pwd</code>	<code>getwd()</code>
<code>cd</code>	<code>setwd()</code>
<code>ls</code>	<code>list.files()</code>
<code>touch</code>	<code>file.create()</code>
<code>mv</code>	<code>file.rename()</code>
<code>cp</code>	<code>file.copy()</code>
<code>rm</code>	<code>file.remove()</code>
<code>ls</code> + visual verification	<code>file.exists()</code>