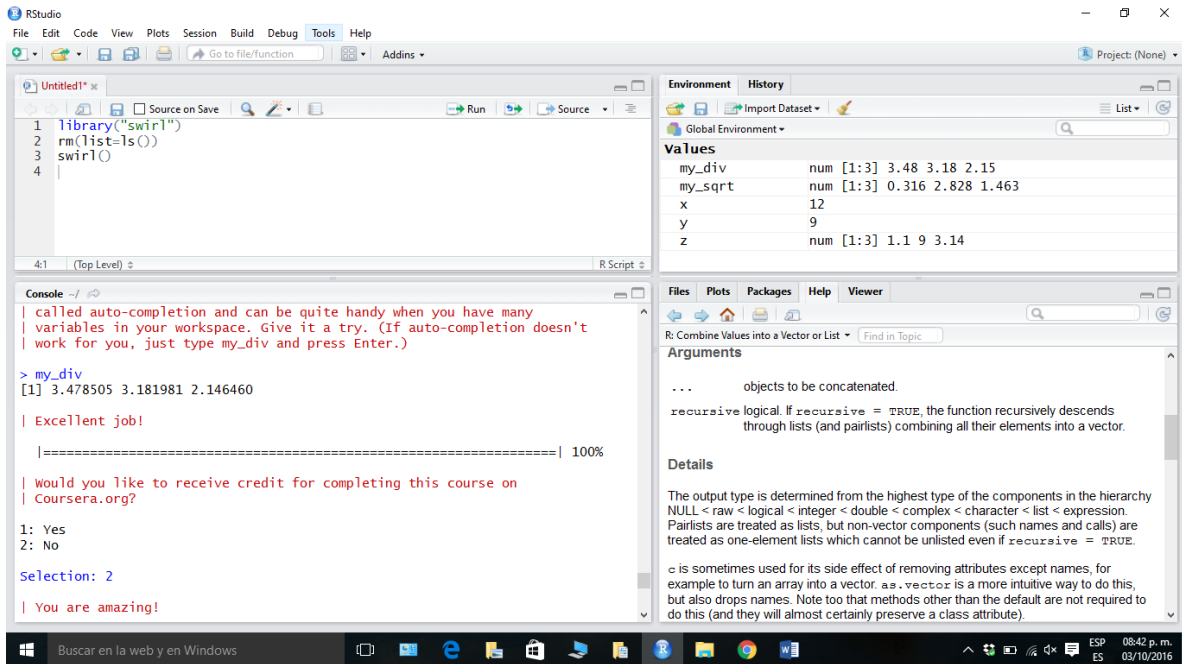


Lección 1



RStudio interface showing the first part of Lesson 1. The console displays a message about auto-completion and a poll question about receiving credit for completing the course.

```
library("swirl")
rm(list=ls())
swirl()

> my_div
[1] 3.478505 3.181981 2.146460

| Excellent job!

|=====| 100%

| Would you like to receive credit for completing this course on
| Coursera.org?

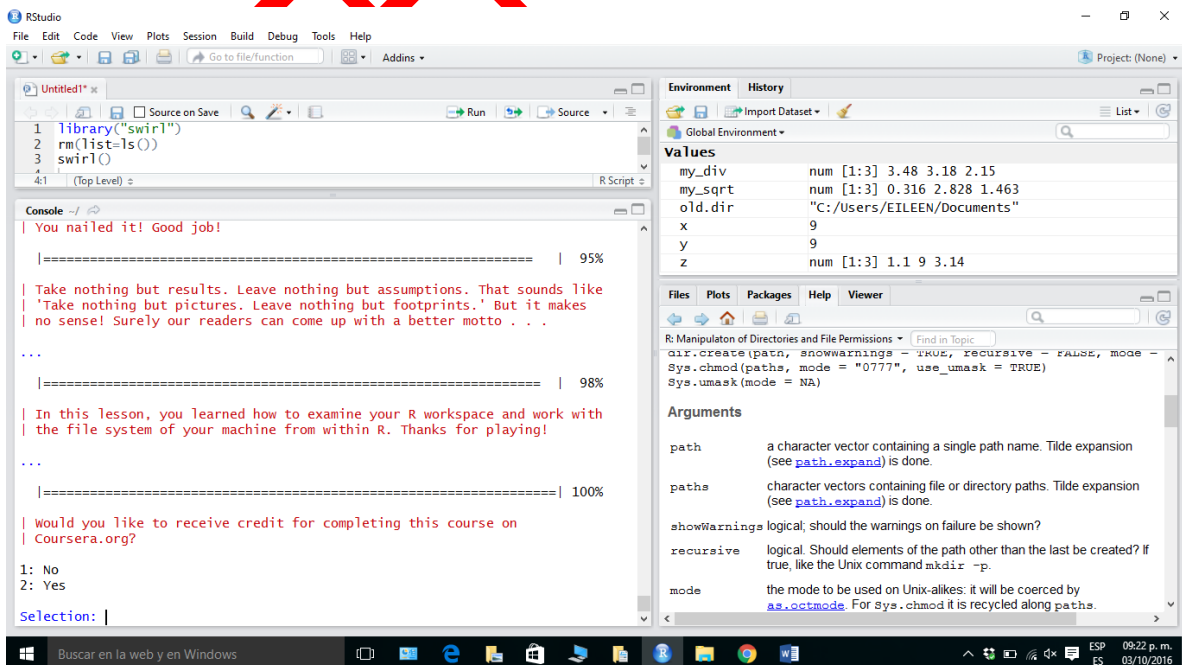
1: Yes
2: No

Selection: 2

| You are amazing!
```

Environment: my_div (num [1:3] 3.48 3.18 2.15), my_sqrt (num [1:3] 0.316 2.828 1.463), x (12), y (9), z (num [1:3] 1.1 9 3.14).

SWIRL



RStudio interface showing the second part of Lesson 1. The console displays a message about taking nothing but results and a poll question about receiving credit for completing the course.

```
library("swirl")
rm(list=ls())
swirl()

| You nailed it! Good job!

|=====| 95%

| Take nothing but results. Leave nothing but assumptions. That sounds like
| 'Take nothing but pictures. Leave nothing but footprints.' But it makes
| no sense! Surely our readers can come up with a better motto . . .

...

|=====| 98%

| In this lesson, you learned how to examine your R workspace and work with
| the file system of your machine from within R. Thanks for playing!

...

|=====| 100%

| Would you like to receive credit for completing this course on
| Coursera.org?

1: No
2: Yes

Selection: |
```

Environment: my_div (num [1:3] 3.48 3.18 2.15), my_sqrt (num [1:3] 0.316 2.828 1.463), old_dir ("C:/Users/EILEEN/Documents"), x (9), y (9), z (num [1:3] 1.1 9 3.14).

Files: R: Manipulation of Directories and File Permissions. Arguments: path (a character vector containing a single path name), paths (character vectors containing file or directory paths), showWarnings (logical), recursive (logical), mode (the mode to be used on Unix-alikes).

Lección 2



The screenshot shows the RStudio IDE with the following content:

Script Editor:

```
library("swirl")
rm(list=ls())
swirl()

# Run button
```

Console:

```
...
|=====| 95%

| Also worth noting is that the numeric vector 1:4 gets 'coerced' into a
| character vector by the paste() function.

...
|=====| 97%

| We'll discuss coercion in another lesson, but all it really means is that
| the numbers 1, 2, 3, and 4 in the output above are no longer numbers to
| R, but rather characters "1", "2", "3", and "4".

...
|=====| 100%

| Would you like to receive credit for completing this course on
| Coursera.org?

1: Yes
2: No

Selection:
```

Environment Pane:

Variable	Class	Value
my_char	chr	[1:3] "My" "name" "is"
my_div	num	[1:3] 3.48 3.18 2.15
my_name	chr	[1:4] "My" "name" "is" "Eileen"
my_seq	num	[1:30] 5 5.17 5.34 5.52 5.69 ...
my_sqrt	num	[1:3] 0.316 2.828 1.463
num_vect	num	[1:4] 0.5 55 -10 6

Lección 3

The RStudio interface displays a script with the following code:

```
1 library("swirl")
2 rm(list=ls())
3 swirl()
4 ?sample
5
```

The console shows the output of the script, including a progress bar at 100% and a question about receiving credit for completing the course on Coursera.org.

The Environment pane shows the following objects:

Object	Class	Attributes
my_data	num	[1:100] -0.166 -1.629 NA NA NA ...
my_div	num	[1:3] 3.48 3.18 2.15
my_na	logi	[1:100] FALSE FALSE TRUE TRUE TRUE FAL...
my_name	chr	[1:4] "My" "name" "is" "Eileen"
my_seq	num	[1:30] 5 5.17 5.34 5.52 5.69 ...
my_sqrt	num	[1:3] 0.316 2.828 1.463
num_vect	num	[1:4] 0.5 55 -10 6

The Help pane shows the documentation for the `is.finite` and `is.infinite` functions.

The RStudio interface displays a script with the following code:

```
1 library("swirl")
2 rm(list=ls())
3 swirl()
4 ?sample
5
```

The console shows the output of the script, including a progress bar at 97% and a question about receiving credit for completing the course on Coursera.org.

The Environment pane shows the following objects:

Object	Class	Attributes
my_seq	num	[1:30] 5 5.17 5.34 5.52 5.69 ...
my_sqrt	num	[1:3] 0.316 2.828 1.463
num_vect	num	[1:4] 0.5 55 -10 6
old_dir	chr	"c:/Users/EILEEN/Documents"
tf	logi	[1:4] TRUE FALSE TRUE FALSE
vect	Named num	[1:3] 11 2 NA
vect2	Named num	[1:3] 11 2 NA

The Help pane shows the documentation for the `is.finite` and `is.infinite` functions.

Lección 4

RStudio interface showing the end of Lesson 4. The console displays a progress bar at 97%, followed by a 100% completion message and a selection prompt. The Environment pane shows variables my_data, my_matrix, my_matrix2, and my_vector. The R Documentation pane shows the 'Matrices' section.

```
| All that practice is paying off!
|=====| 97%

| In this lesson, you learned the basics of working with two
| very important and common data structures -- matrices and data
| frames. There's much more to learn and we'll be covering more
| advanced topics, particularly with respect to data frames, in
| future lessons.
...
|=====| 100%

| Would you like to receive credit for completing this course on
| Coursera.org?

1: Yes
2: No

Selection: 2

| That's a job well done!

| You've reached the end of this lesson! Returning to the main
| menu...

| Would you like to continue with one of these lessons?

1: R Programming Basic Building Blocks
2: No. Let me start something new.

Selection: |
```

Environment: my_data (4 obs. of 6 variables), my_matrix (int [1:4, 1:5]), my_matrix2 (int [1:4, 1:5]), my_vector (int [1:4, 1:5]).

Values: chr [1:6] "patient" "age" "weight" "bp" "rating" ...

R Documentation: matrix (base). Description: matrix creates a matrix from the given set of values. Usage: matrix(data = NA, nrow = 1, ncol = 1, byrow = FALSE, dimnames = NULL).

RStudio interface showing the start of Lesson 5. The console displays a progress bar at 98%, followed by a 100% completion message and a selection prompt. The Environment pane shows variables my_data, my_matrix, my_matrix2, and my_vector. The R Documentation pane shows the 'Matrices' section.

```
1 |

2: all(c(TRUE, FALSE, TRUE))
3: any(ints == 2.5)
4: any(ints == 10)

Selection: 4

| You are really on a roll!
|=====| 98%

| That's all for this introduction to logic in R. If you really want to
| see what you can do with logic, check out the control flow lesson!
...
|=====| 100%

| Would you like to receive credit for completing this course on
| coursera.org?

1: Yes
2: No

Selection: |
```

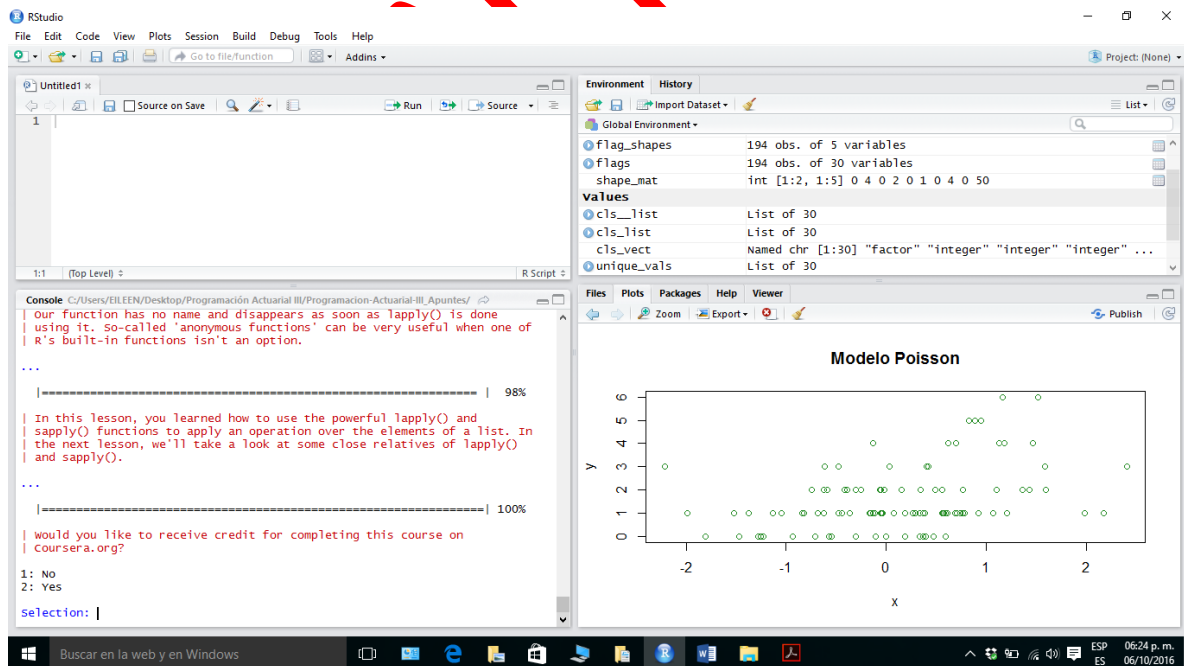
Environment: my_data (4 obs. of 6 variables), my_matrix (int [1:4, 1:5]), my_matrix2 (int [1:4, 1:5]), my_vector (int [1:4, 1:5]).

Values: chr [1:6] "patient" "age" "weight" "bp" "rating" ...

R Documentation: matrix (base). Description: matrix creates a matrix from the given set of values. Usage: matrix(data = NA, nrow = 1, ncol = 1, byrow = FALSE, dimnames = NULL).

Lección 5

✓



Lección 6

The screenshot shows the RStudio interface. The script editor on the left contains a single line of code: `1`. The console on the bottom left shows the output of a selection function, with a progress bar at 96% and 100%. The environment pane on the right shows the Global Environment with variables: `flag_shapes` (194 obs. of 5 variables), `flags` (194 obs. of 30 variables), `shape_mat` (int [1:2, 1:5] 0 4 0 2 0 1 0 4 0 50), `cls_list` (List of 30), `cls_vect` (Named chr [1:30] "factor" "integer" "integer" "integer" ...), and `unique_vals` (List of 30). The viewer pane on the right shows the documentation for `tapply` (base), titled "Apply a Function Over a Ragged Array".

```
1
```

Selection: 1

| All that practice is paying off!

|=====| 96%

In this lesson, you learned how to use `vapply()` as a safer alternative to `sapply()`, which is most helpful when writing your own functions. You also learned how to use `tapply()` to split your data into groups based on the value of some variable, then apply a function to each group. These functions will come in handy on your quest to become a better data analyst.

...

|=====| 100%

| would you like to receive credit for completing this course on coursera.org?

1: Yes
2: No

Selection:

The screenshot shows the RStudio interface. The script editor on the left contains a single line of code: `1`. The console on the bottom left shows the output of a selection function, with a progress bar at 96% and 100%. The environment pane on the right shows the Global Environment with variables: `flag_shapes` (194 obs. of 5 variables), `flags` (194 obs. of 30 variables), `plants` (5166 obs. of 10 variables), `shape_mat` (int [1:2, 1:5] 0 4 0 2 0 1 0 4 0 50), `cls_list` (List of 30), `cls_vect` (Named chr [1:30] "factor" "integer" "integer" "integer" ...), and `unique_vals` (List of 30). The viewer pane on the right shows the documentation for `tapply` (base), titled "Apply a Function Over a Ragged Array".

```
1
```

Selection: 1

| All that practice is paying off!

|=====| 96%

In this lesson, you learned how to get a feel for the structure and contents of a new dataset using a collection of simple and useful functions. Taking the time to do this upfront can save you time and frustration later on in your analysis.

...

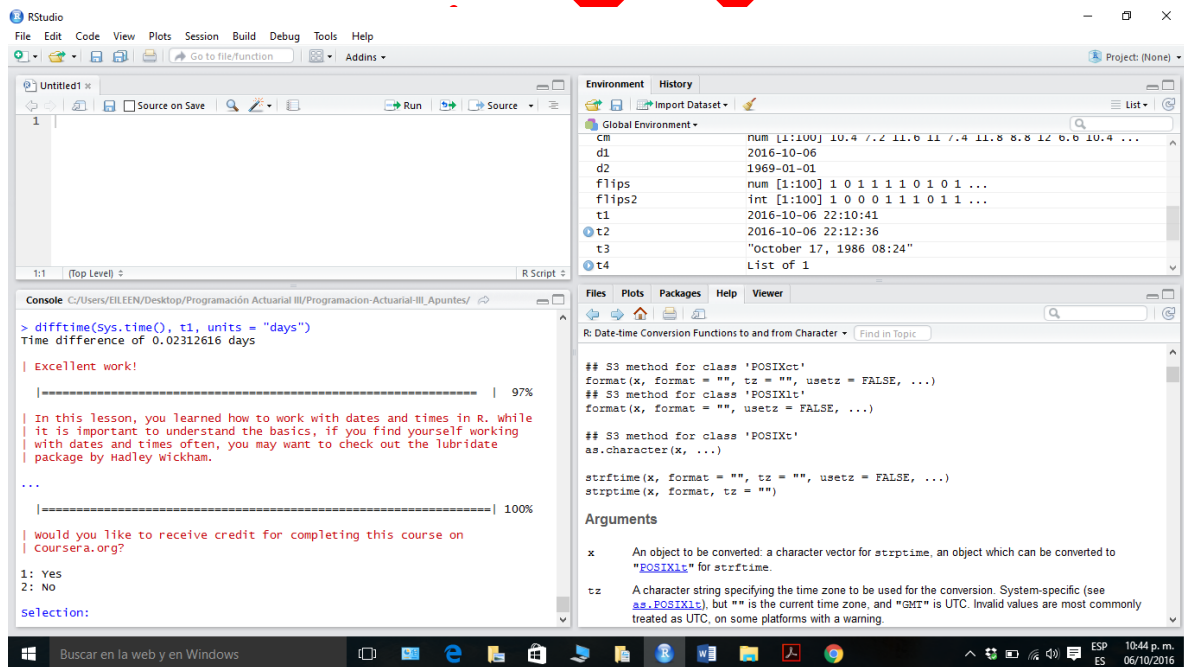
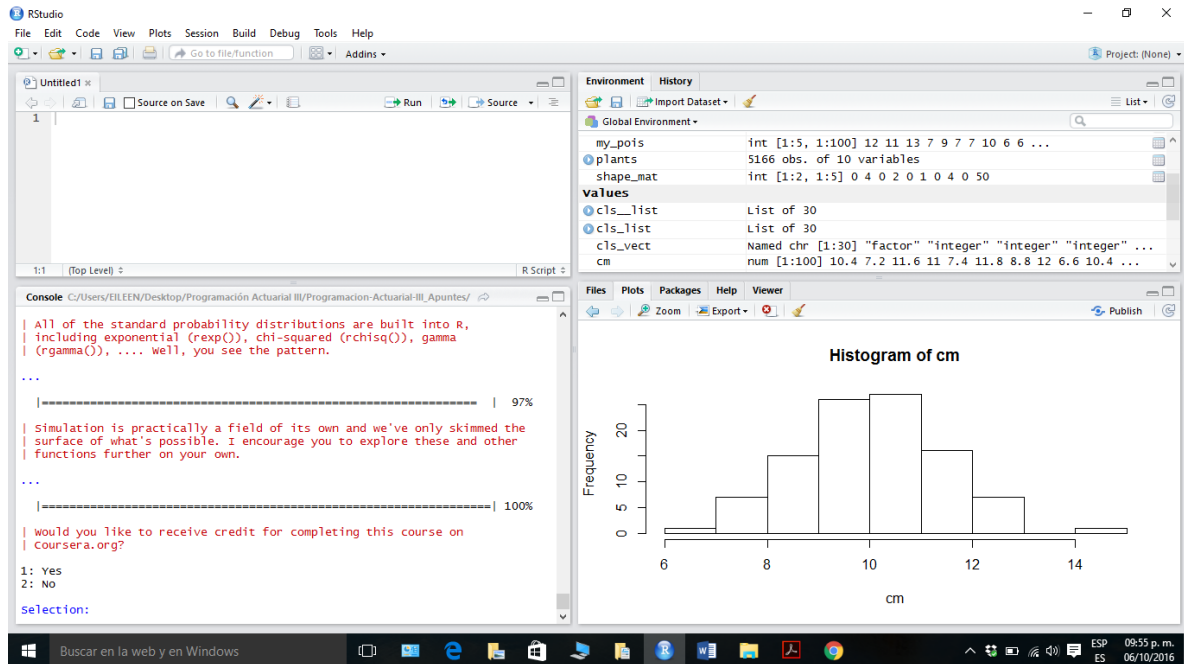
|=====| 100%

| would you like to receive credit for completing this course on coursera.org?

1: Yes
2: No

Selection: |

Lección 7



Lección 8

