Making an R package

Hadley Wickham
Assistant Professor / Dobelman Family Junior Chair
Department of Statistics / Rice University

November 2010
What I’ve learned from developing 20+ R packages

https://github.com/hadley/devtools/wiki
1. Getting started
2. Development cycle
3. Documentation
4. What to learn next
Use a mac! (or linux)

Otherwise, start by downloading and installing
http://www.murdoch-sutherland.com/Rtools/
Getting started
1. Decide on a name (stringr)

2. Create `stringr/` and `stringr/R/` directories

3. Copy in your existing code

```
R
├── file1.r
├── file2.r
```
4. Add a description file
Package: stringr
Type: Package
Title: Make it easier to work with strings.
Version: 0.5
Author: Hadley Wickham <h.wickham@gmail.com>
Maintainer: Hadley Wickham <h.wickham@gmail.com>
Description: stringr is a set of simple wrappers that make R’s string functions more consistent, simpler and easier to use. It does this by ensuring that: function and argument names (and positions) are consistent, all functions deal with NA’s and zero length character appropriately, and the output data structures from each function matches the input data structures of other functions.
Imports: plyr
Depends: R (>= 2.11.0)
Suggests: testthat (>= 0.3)
License: GPL-2

https://github.com/hadley/devtools/wiki/Package-basics
4. Document your files
5. Run roxygenize()
The length of a string (in characters).

@param string input character vector
@return numeric vector giving number of characters in each element of the character vector. Missing strings have missing length.
@keywords character
@seealso \link{nchar} which this function wraps
@export
@examples
str_length(letters)
str_length(c("i", "like", "programming", NA))

str_length <- function(string) {
  string <- check_string(string)

  nc <- nchar(string, allowNA = TRUE)
  is.na(nc) <- is.na(string)
  nc
}

https://github.com/hadley/devtools/wiki/docs-function
6. Run `R CMD check`

7. Use `R CMD build` to create a package file

8. Upload file to CRAN:
   ```
   ftp -u ftp://cran.R-project.org/incoming/
   stringr_0.5.tar.gz
   ```

https://github.com/hadley/devtools/wiki/Release
You’re done!
Expect to be frustrated! (at first)
Development cycle
Exploratory programming

Identify the task

Modify and save code

Reload in R

Does it work?

NO

YES

Write an automated test
Confirmatory programming

Write an automated test

Modify and save code

Reload in R

Does it work?

You’re done
# These patterns are facilitated by the devtools package: https://github.com/hadley/devtools

# Once installed and set up you can easily:

# * Reload code and data
   load_all("stringr")

# * Run automated tests
   test("stringr")

# * Update documentation
   document("stringr")
Documentation
Function-level: gives precise details about individual functions

Package-level: gives details about how to use the functions to solve real (complex) problems.

Both are essential!
Function-level

Keep code and documentation close together with roxygen.

Writing documentation gets easier with time!
Package-level

Package help topic
NEWS
Vignettes + CITATION
Demos
README

https://github.com/hadley/devtools/wiki/docs-package
Next...
Learning more

Testing
Namespaces
Code style
Git + github
Learn from others

Read the source of other packages!

https://github.com/hadley/plyr
https://github.com/hadley/stringr
https://github.com/hadley/lubridate
https://github.com/hadley/evaluate
https://github.com/hadley/reshape
This work is licensed under the Creative Commons Attribution-Noncommercial 3.0 United States License. To view a copy of this license, visit http://creativecommons.org/licenses/by-nc/3.0/us/ or send a letter to Creative Commons, 171 Second Street, Suite 300, San Francisco, California, 94105, USA.