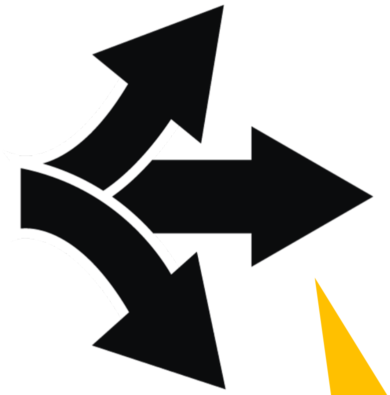
Why should you create a package?

Why should you create an R package?

Easily distribute your code

- Internally (e. g. in your team / company).
- Publicly (everyone can use it).



Document your code

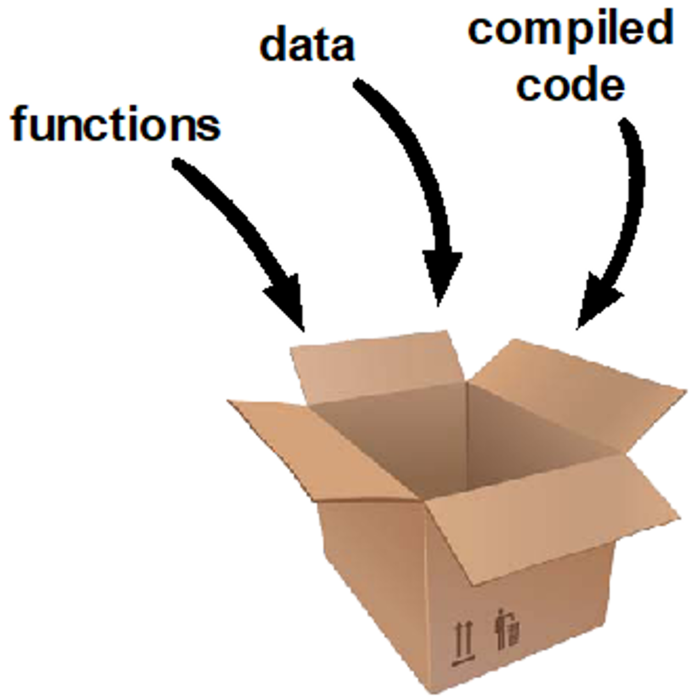
- Easy access to function documentation through the "man" files
- Possibility to add examples.

A good package fulfills several requirements:

- Easy package installation
- The basic directory structure is intuitive
- Detailed and comprehensive documentation exists

What is actually a package?

Create a collection of functions, data and compiled code:



Prepare your environment

Mac OS

Install "Command Line Tools":

1. Launch the Terminal (in "/Applications/Utilities").
2. Type `xcode-select --install`
3. Click on "Install".

Install MacTeX LaTeX:

http://www.tug.org/mactex/downloading.html

http://personality-project.org/r/makingpackages.html

Windows

Install "Rtools":

https://cran.rstudio.com/bin/windows/Rtools/

Install "LaTeX":

https://miktex.org/download

https://support.rstudio.com/hc/en-us/articles/200486498-Package-Development-Prerequisites

Linux

Install "core software development utilities" and "LaTeX":

```
sudo apt-get install r-base-dev texlive-full
```

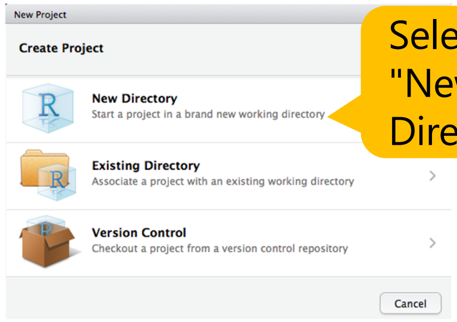
Some packages may require additional R build dependencies:

```
sudo apt-get build-dep r-base-core
```

https://support.rstudio.com/hc/en-us/articles/200486498-Package-Development-Prerequisites

Creating a package(1/3)
Let’s get started with your first package

Create a New Project in RStudio ("File" -> "New Project"):



New Project

Create Project

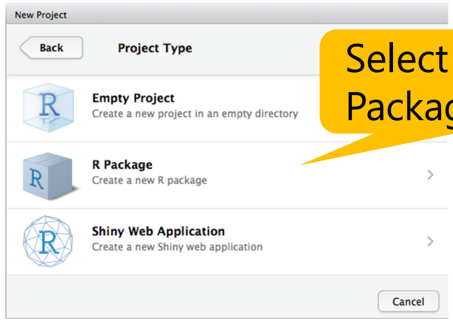
New Directory
Start a project in a brand new working directory

Existing Directory
Associate a project with an existing working directory

Version Control
Checkout a project from a version control repository

Cancel

Select "New Directory".



New Project

Back | Project Type

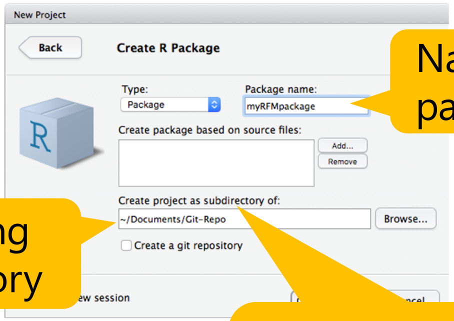
Empty Project
Create a new project in an empty directory

R Package
Create a new R package

Shiny Web Application
Create a new Shiny web application

Cancel

Select "R Package".



New Project

Back | Create R Package

Type: Package

Package name: myRFMpackage

Create package based on source files:

Add...
Remove

Create project as subdirectory of:
~/Documents/Git-Repo | Browse...

Create a git repository

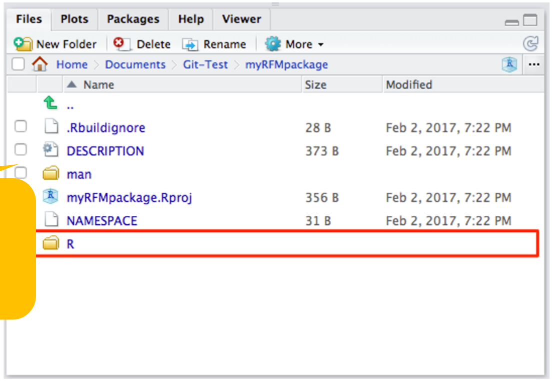
ew session

Name of your package

Working Directory

Include already written functions for your package here.

Put all our r functions in the "R" folder.



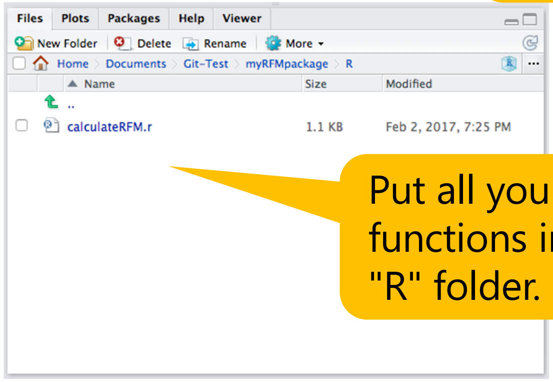
Files | Plots | Packages | Help | Viewer

New Folder | Delete | Rename | More

Home > Documents > Git-Test > myRFMpackage

Name	Size	Modified
..		
.Rbuildignore	28 B	Feb 2, 2017, 7:22 PM
DESCRIPTION	373 B	Feb 2, 2017, 7:22 PM
man		
myRFMpackage.Rproj	356 B	Feb 2, 2017, 7:22 PM
NAMESPACE	31 B	Feb 2, 2017, 7:22 PM
R		

Structure is automatically generated.



Files | Plots | Packages | Help | Viewer

New Folder | Delete | Rename | More

Home > Documents > Git-Test > myRFMpackage > R

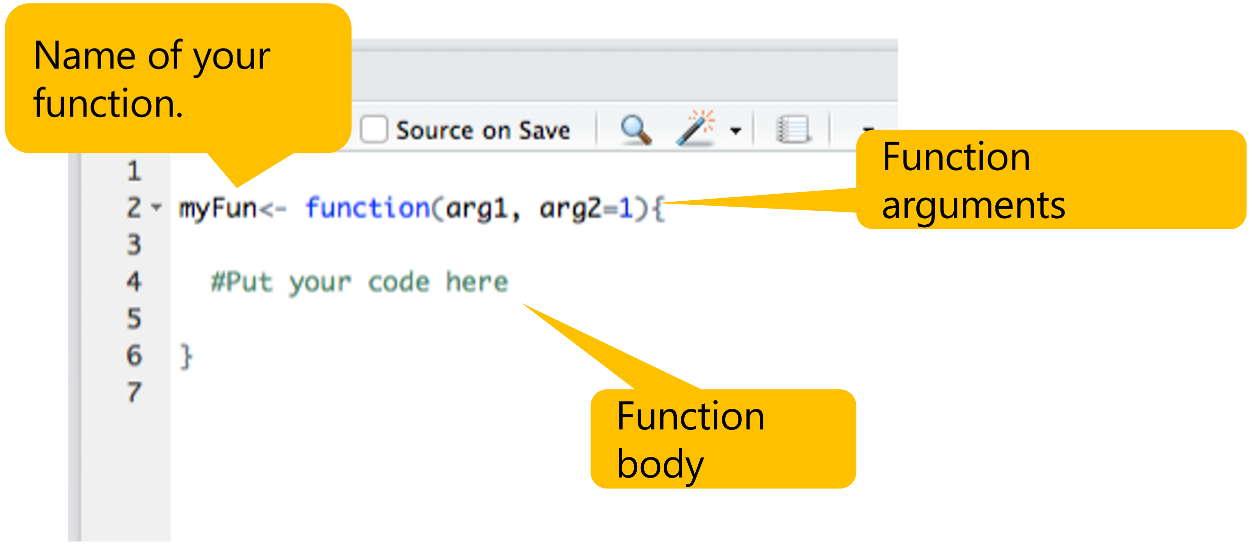
Name	Size	Modified
..		
calculateRFM.r	1.1 KB	Feb 2, 2017, 7:25 PM

Put all your r functions in the "R" folder.

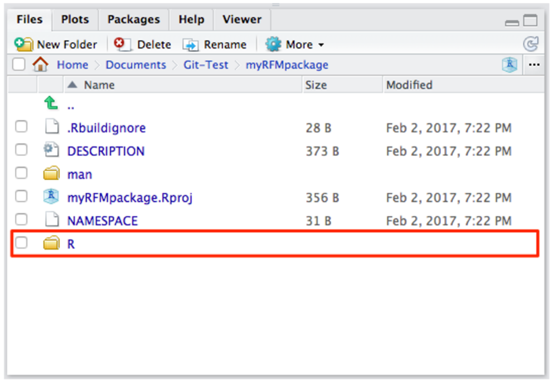
Modules containing the "real" package code

Store your actual module code such (classes and functions) in separate .R files.

Repetition - function structure in R:

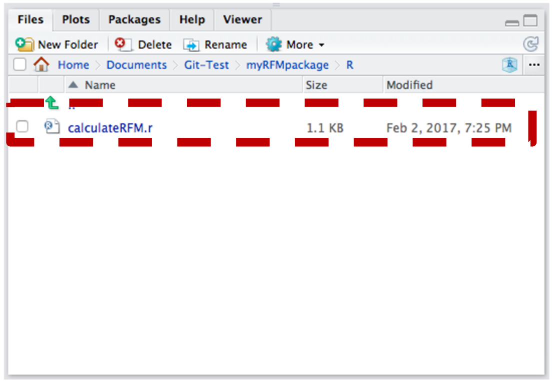


```
myFun<- function(arg1, arg2=1){

  #Put your code here

}
```



	Name	Size	Modified
	..		
	.Rbuildignore	28 B	Feb 2, 2017, 7:22 PM
	DESCRIPTION	373 B	Feb 2, 2017, 7:22 PM
	man		
	myRFMpackage.Rproj	356 B	Feb 2, 2017, 7:22 PM
	NAMESPACE	31 B	Feb 2, 2017, 7:22 PM
	R		

Files Plots Packages Help Viewer
New Folder Delete Rename More
Home > Documents > Git-Test > myRFMpackage

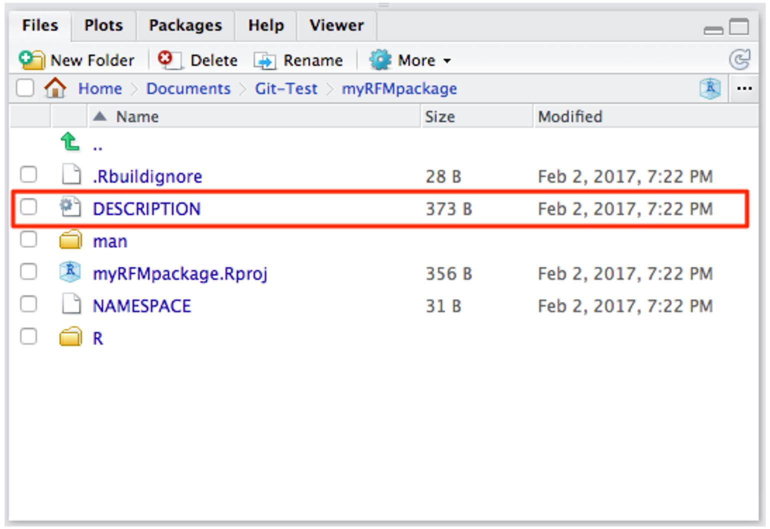


	Name	Size	Modified
	..		
	calculateRFM.r	1.1 KB	Feb 2, 2017, 7:25 PM

Files Plots Packages Help Viewer
New Folder Delete Rename More
Home > Documents > Git-Test > myRFMpackage > R

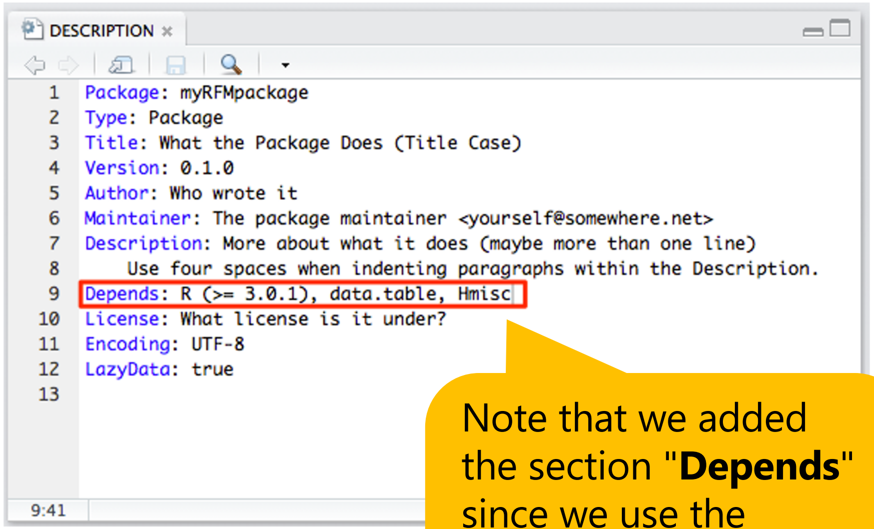
Creating a package(2/3)
Describe your package

Add additional information for your package



Files Plots Packages Help Viewer
New Folder Delete Rename More
Home > Documents > Git-Test > myRFMpackage

Name	Size	Modified
..		
.Rbuildignore	28 B	Feb 2, 2017, 7:22 PM
DESCRIPTION	373 B	Feb 2, 2017, 7:22 PM
man		
myRFMpackage.Rproj	356 B	Feb 2, 2017, 7:22 PM
NAMESPACE	31 B	Feb 2, 2017, 7:22 PM
R		

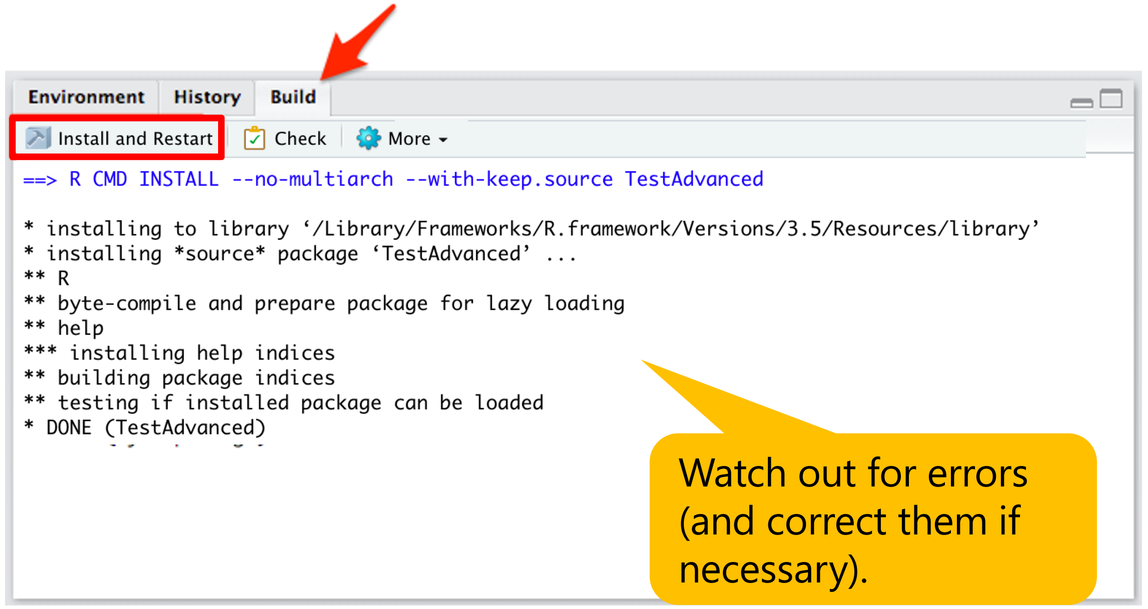


```
Package: myRFMpackage
Type: Package
Title: What the Package Does (Title Case)
Version: 0.1.0
Author: Who wrote it
Maintainer: The package maintainer <yourself@somewhere.net>
Description: More about what it does (maybe more than one line)
    Use four spaces when indenting paragraphs within the Description.
Depends: R (>= 3.0.1), data.table, Hmisc
License: What license is it under?
Encoding: UTF-8
LazyData: true
```

Note that we added the section "**Depends**" since we use the data.table and Hmisc package in our function.

Creating a package(3/3)
Rock n'Roll

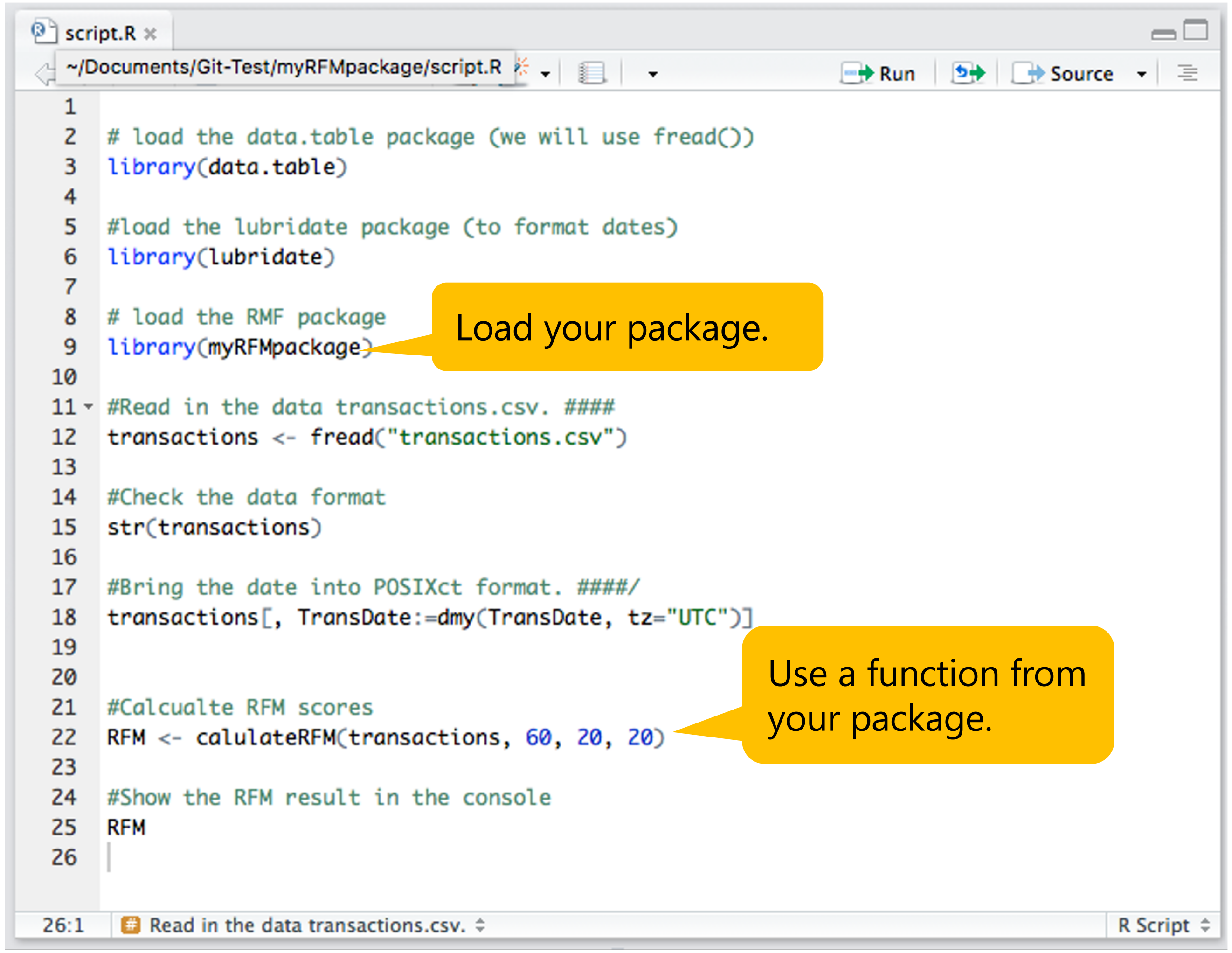
Build your package:



```
==> R CMD INSTALL --no-multiarch --with-keep.source TestAdvanced

* installing to library '/Library/Frameworks/R.framework/Versions/3.5/Resources/library'
* installing *source* package 'TestAdvanced' ...
** R
** byte-compile and prepare package for lazy loading
** help
*** installing help indices
** building package indices
** testing if installed package can be loaded
* DONE (TestAdvanced)
```

Your package is automatically added to your package repository and loaded (using the `library()` command).

Use functions from your package


script.R
~/Documents/Git-Test/myRFMpackage/script.R
Run
Source
Load your package.
Use a function from your package.
26:1
Read in the data transactions.csv.
R Script

```r

# load the data.table package (we will use fread())
library(data.table)

#load the lubridate package (to format dates)
library(lubridate)

# load the RMF package
library(myRFMpackage)

#Read in the data transactions.csv. ####
transactions <- fread("transactions.csv")

#Check the data format
str(transactions)

#Bring the date into POSIXct format. ####/
transactions[, TransDate:=dmy(TransDate, tz="UTC")]


#Calcualte RFM scores
RFM <- calulateRFM(transactions, 60, 20, 20)

#Show the RFM result in the console
RFM

```

Now it’s your turn!