R Package and devtools

TOTAL POINTS 10

1. Which of the following are good reasons to build an R Package? 1 point

- [x] R functions are only available for use in other programming languages (Python, C++) if they're included in a package.
- [x] Users need to understand how your R functions work in order to use them and providing users with a package allows them to read your R code.
- [x] R Packages require documentation and the package structure allows you to distribute functions and documentation together.
- [x] An R package is a good method for distributing functions to users.

2. Which of the following files and folders are required in an R package? 1 point

- [x] The R/ directory.
- [x] A NAMESPACE file.
- [x] An inst/ directory.
- [x] The man/ directory.
- [x] A DESCRIPTION file.
- [x] A README.md file.

3. Which of the following files and subdirectories will be included in the initial package directory if you create a new package using the `create` function from `devtools`? 1 point

- [x] `R/` subdirectory
- [x] NAMESPACE

- [x] README.md
- [x] R project file (package name plus a `.Rproj` extension)
- [x] README.Rmd
- [x] `data/` subdirectory
- [x] .gitignore
- [x] `man/` subdirectory
- [x] `vignettes/` subdirectory
- [x] .Rbuildignore
- [x] DESCRIPTION

4. Which of the following functions from the `devtools` package are you likely to use often, rather than just once per package, when building a package? 1 point

- [x] `load_all`
- [x] `create`
- [x] `use_readme_rmd`
- [x] `document`

5. What is the purpose of the DESCRIPTION file in a package? 1 point

- ( ) It describes some of the R project options for the package.
- (•) It provides metadata on the package, including the package name, version number, authors, and dependencies.
- ( ) It provides a tutorial overview of how to use the package and is typically written in R Markdown.
- ( ) It describes all the functions that are exported by the package.

6. Which of the following statements correctly describes how R functions should be defined with the package directory? 1 point

- [x] They should all be defined in separate R scripts (one per function) in the `R/` subdirectory.
- [ ] They should all be defined in a single R script, saved as the NAMESPACE file.
- [ ] They should all be defined in either a single or multiple R scripts in the `R/` subdirectory.
- [ ] They should all be defined in separate R scripts in the `man/` subdirectory.

7. What does the ::: operator do? 1 point

- [ ] It allows access to unexported functions in a package
- [x] It hides functions in a package to make them inaccessible to the user
- [ ] It exports functions from a package for others to use
- [ ] It prevents users from modifying the code in a function

8. How is attaching a package namespace different from loading a namespace? 1 point

- [x] Attaching a namespace makes functions visible only to the package that attached the namespace
- [ ] Loading a namespace places a package's functions in the global environment
- [ ] Attaching a namespace places the namespace on the search list
- [ ] Loading a namespace makes unexported functions in a package visible to the user

9. For packages that require C code, what should be installed on your system? 1 point

- [ ] A perl interpreter
- [ ] A Fortran compiler
- [x] A C compiler from Xcode, Rtools, or equivalent package
- [ ] The appropriate header files

10. What is the purpose of the Imports field in the DESCRIPTION file? 1 point

- (●) to indicate the packages whose functions will be used in your package
- ( ) to indicate the version of R that is required for your package
- ( ) to provide a brief description of what your package does
- ( ) to indicate the license under which your package is made available

---

☑ I, **JAY PATEL**, understand that submitting work that isn't my own may result in permanent failure of this course or deactivation of my Coursera account.

Learn more about Coursera's Honor Code