R/RStudio/RCpp Tips and Tricks:

1. **Auto Completion**: Begin typing an R command and then hit the tab key: R Studio will fill out the rest of the command for you.
2. **Close parenthesis and quotation highlighting:** RStudio takes care of unclosed parenthesis.
3. **Disable automatic insertion of closing parenthesis/quotation mark**: By default, R Studio will automatically insert a closing parenthesis or quotation mark when you begin typing an opening parenthesis or quotation mark. This can be turned off from the 'Tools'.
4. **Enable code wrapping**: Any part of your R script that would usually fall off the side of your screen is, instead, continued on the following line.
5. **Installing packages from menu:** Packages can be installed from the side menu.
6. **Enabling packages from side-menu:** A convenient way to load up R packages in R Studio is to simply check/uncheck the relevant boxes in the 'Packages' side-menu. This menu also allows you to quickly take inventory of all your active packages, which may be useful in situations where you need to free up some memory or debug a package incompatibility.
7. **Set working directory from side-menu:** Instead of typing the setwd() command you can set your working directly via the 'Files' side-menu by first navigating to your desired file location and then simply selecting the 'Set As Working Directory' option from the 'More' drop-down menu.
8. **Copy+paste from plot view:** It is possible to copy and paste a plot image directly from the zoom'd plot view without writing it to a .pdf or .png file first. Simply "Zoom" an active plot from the side-menu and right-click the plot view and select the "Copy Image" command.
9. **Quick Commenting:** Last, but certainly not least, you can comment out your code in R Studio by highlighting the offensive lines and simply hitting ctrl+shift+c. To un-comment, simply re-highlight and hit ctrl+shift+c again.

Creating a package from scratch in R:

1. Install "devtools" and "roxygen2".
2. Make sure your default directory is empty.
3. Use *setwd()* to set your current working dir where you want to create the package.
4. Use *create("myPackage")* to create an empty package called "mypackage" in your desired location.
5. Edit the .R file that is created.
6. Add documentation using the roxygen2 documentation package.
7. Process your documentation to using the *document()* command.
8. Install your package: Can be done using command or the GUI on the right side.
9. To use GUI, build the package using, *R CMD build mynewpackage* and then create the .tar file using the command *R CMD INSTALL mynewpackage_0.1.tar.gz*
10. Check Documentation.
11. Enjoy!