

code style

R and whitespace

“R doesn’t care” until it does

- Some languages are very picky about where whitespace (like spaces between words, tabs, and line breaks) go. R doesn’t care about those things!

```
newDF$ratio<-(Feelings[Feelings$InGroup=="FALSE" | Feelings$InGroup=="No", ]$percents)-(Feelings[Feelings$InGroup=="TRUE" |  
Feelings$InGroup=="Yes", ]$percents)
```

```
newDF$ratio <- (Feelings[Feelings$InGroup == "FALSE" | Feelings$InGroup == "No", ]$percents) -  
                (Feelings[Feelings$InGroup == "TRUE" | Feelings$InGroup == "Yes", ]$percents)
```

- Code style suggests adding whitespace for readability.
 - around both sides of assignment operators, <- and =
 - around logical and boolean operators like ==, <, &, |
 - after a comma, just like in English
- But, no whitespace around parentheses for regular function calls

R and whitespace

“R doesn’t care” until it does

- Like most computer languages, R **does** care about spaces in variable names, file names, and folder (directory) names.
- Instead of spaces:
 - use snake_case with underscores to separate words (periods have additional meaning in R, so that can be confusing). This is what the tidyverse guide advocates!
 - or
 - use CamelCase and capitalize words. This may be more accessible to screen readers

Line length

Shorter lines are easier for humans to read

- Add line breaks wherever logical
- After %>% in dplyr chains
- After + in ggplot2 graphics

```
ggplot(comic_characters) + geom_histogram(aes(x = year, fill = publisher), binwidth = 1, color = "white", lwd = 0.1) +  
facet_wrap(~publisher) + theme_fivethirtyeight() + scale_fill_manual(values = c("#008fd5", "#ff2700")) + labs(title = "New Comic  
Book Characters Introduced Per Year")
```

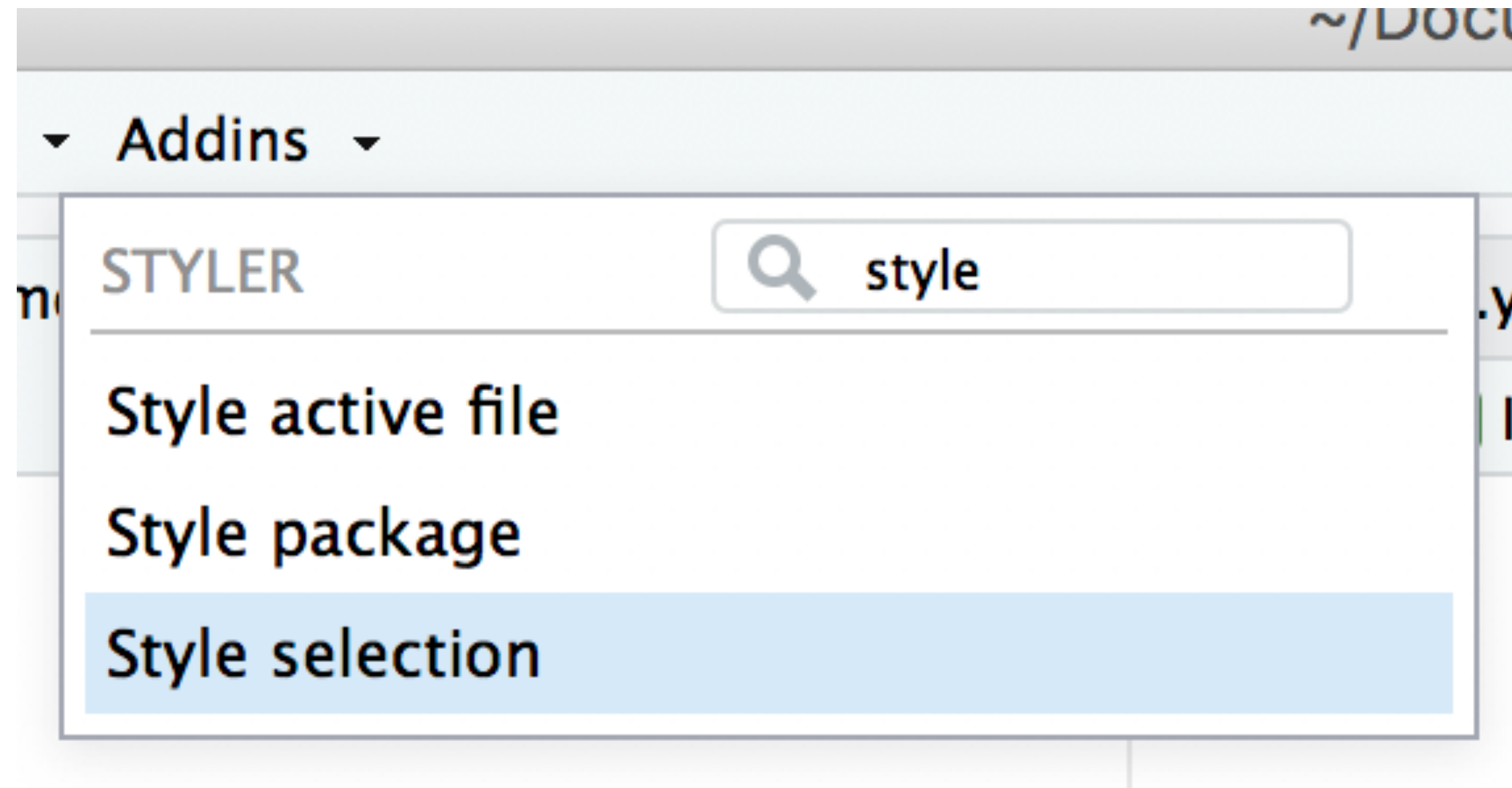
```
ggplot(comic_characters) +  
  geom_histogram(aes(x = year, fill = publisher), binwidth = 1, color = "white", lwd = 0.1) +  
  facet_wrap(~publisher) +  
  theme_fivethirtyeight() +  
  scale_fill_manual(values = c("#008fd5", "#ff2700")) +  
  labs(title = "New Comic Book Characters Introduced Per Year")
```

Styling code

The styler package make styling code easy!

```
library(styler)
```

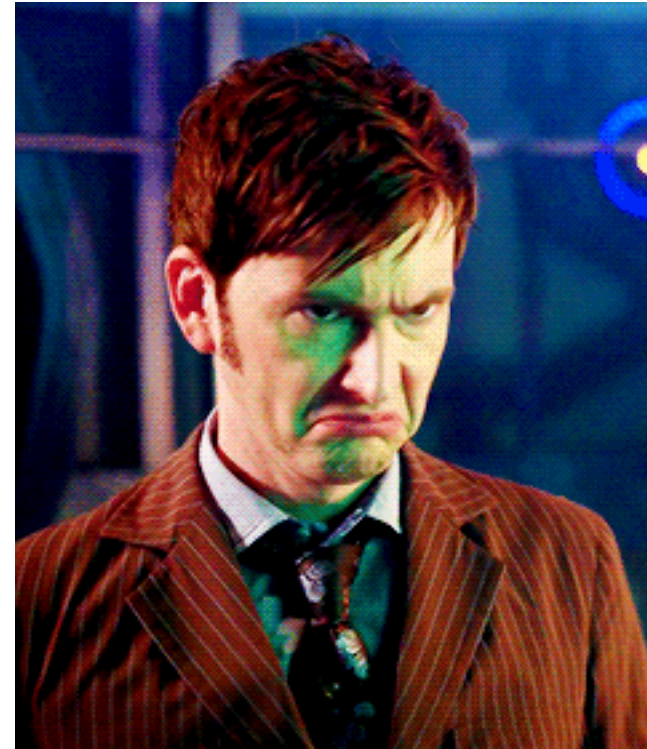
- Load the package
- Use the add-in!



Never use attach()

Where are y'all learning this?!

```
> cyl
Error: object 'cyl' not found
> attach(mtcars)
> cyl
[1] 6 6 4 6 8 6 8 4 4 6 6 8 8 8 8 8 8 4 4 4 4 8 8 8 8 4
```



- attach() is extremely bad R practice
- **Every time** you run attach() a new version of the data is attached into your Environment, but invisibly. This means you may have hundreds of versions of the same dataset floating around in there!


```
> attach(mtcars)
> cyl
[1] 6 6 4 6 8 6 8 4
```

```
> detach(mtcars)
```

```
> mtcars$cyl
[1] 6 6 4 6 8 6 8 4
```

```
> mtcars %>%
pull(cyl)
[1] 6 6 4 6 8 6 8 4
```



If you have been using `attach()` it's time to clean up

```
> search()  
[1] ".GlobalEnv"      "mtcars"          "package:ggthemes"  
[4] "package:fivethirtyeight" "package:lubridate" "package:forcats"  
[7] "package:stringr"  "package:dplyr"   "package:purrr"  
[10] "package:readr"    "package:tidyr"   "package:tibble"  
[13] "package:ggplot2"  "package:tidyverse" "tools:rstudio"  
[16] "package:stats"    "package:graphics" "package:grDevices"  
[19] "package:utils"    "package:datasets" "package:reprex"  
[22] "package:testthat" "package:devtools" "package:usethis"  
[25] "package:methods" "Autoloads"        "package:base"
```

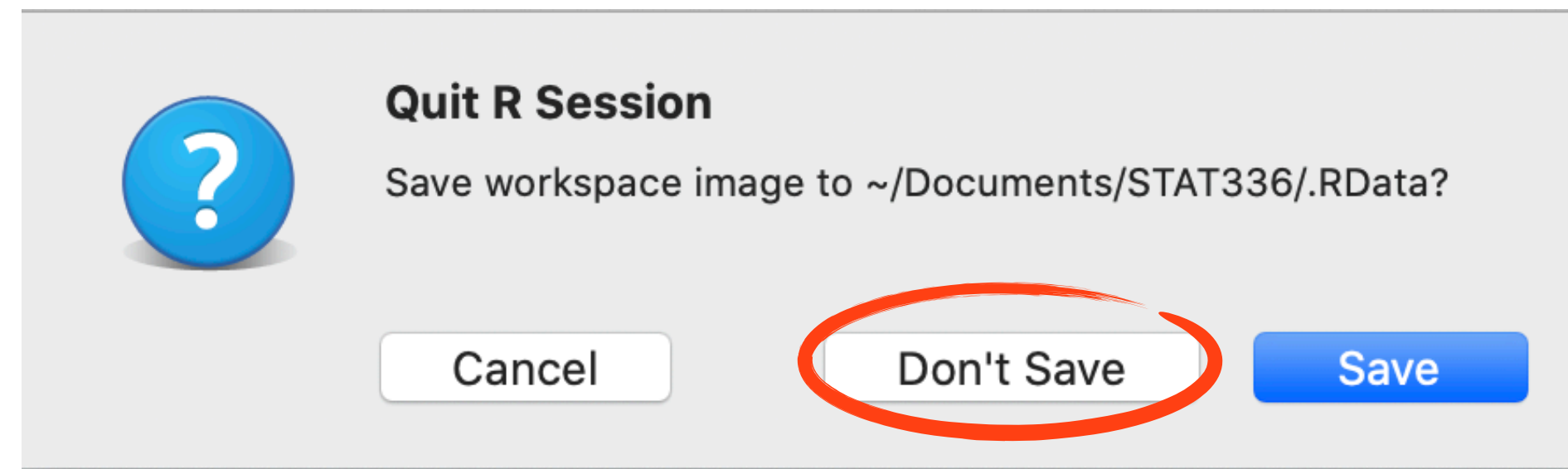
- Use `search()` to see what you have attached
 - `.GlobalEnv` and `Autoloads` are fine
 - anything that starts with `package:` or `tools:` is also fine
- Use `detach()` to remove any data that is attached

```
> detach(mtcars)
```


Don't save your Workspace

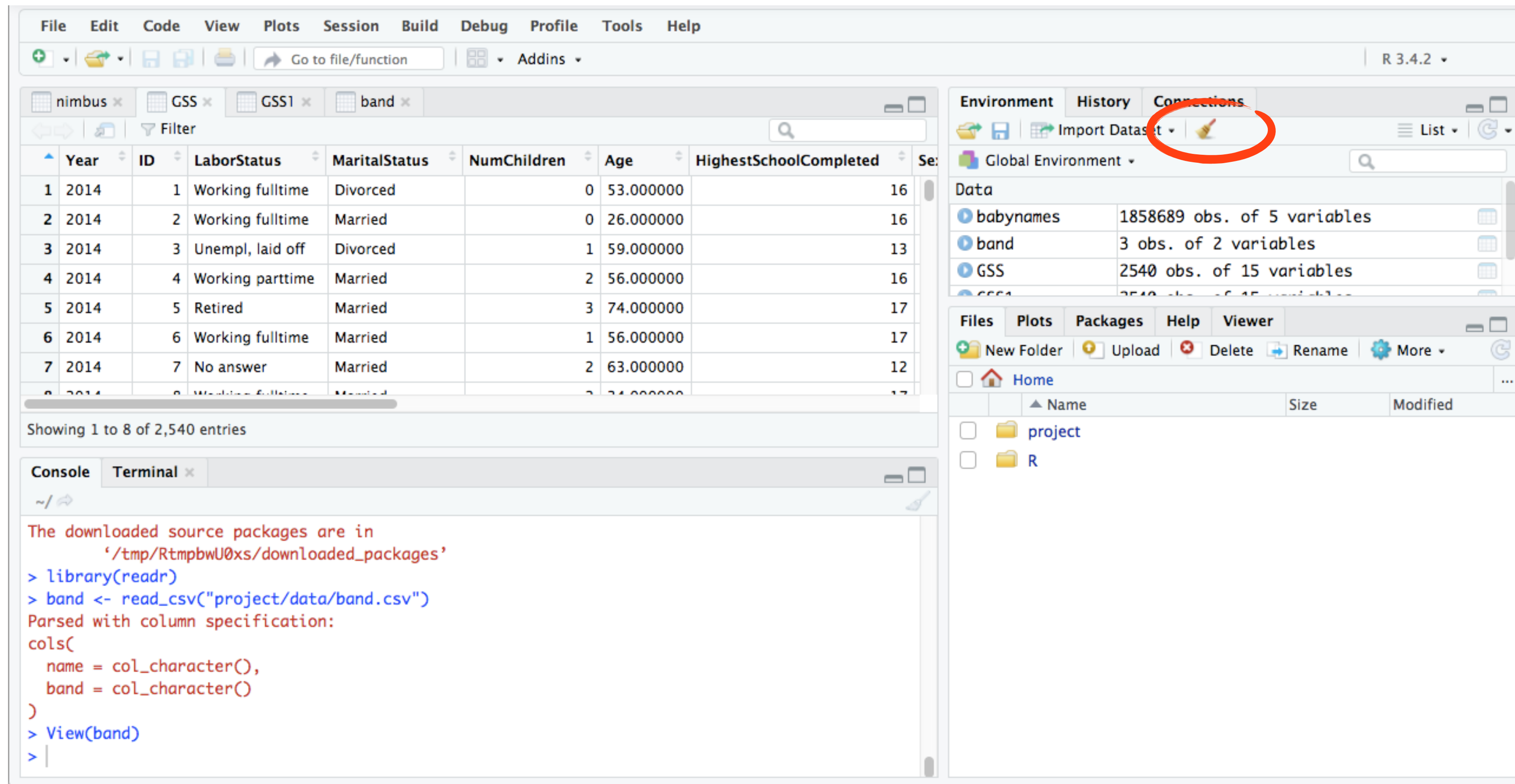
That's why we save code!

- When you quit RStudio, you should be selecting “Don't save” on the dialog box



- If you haven't been doing this, it's again time for a clean up!

First, clean up your Environment

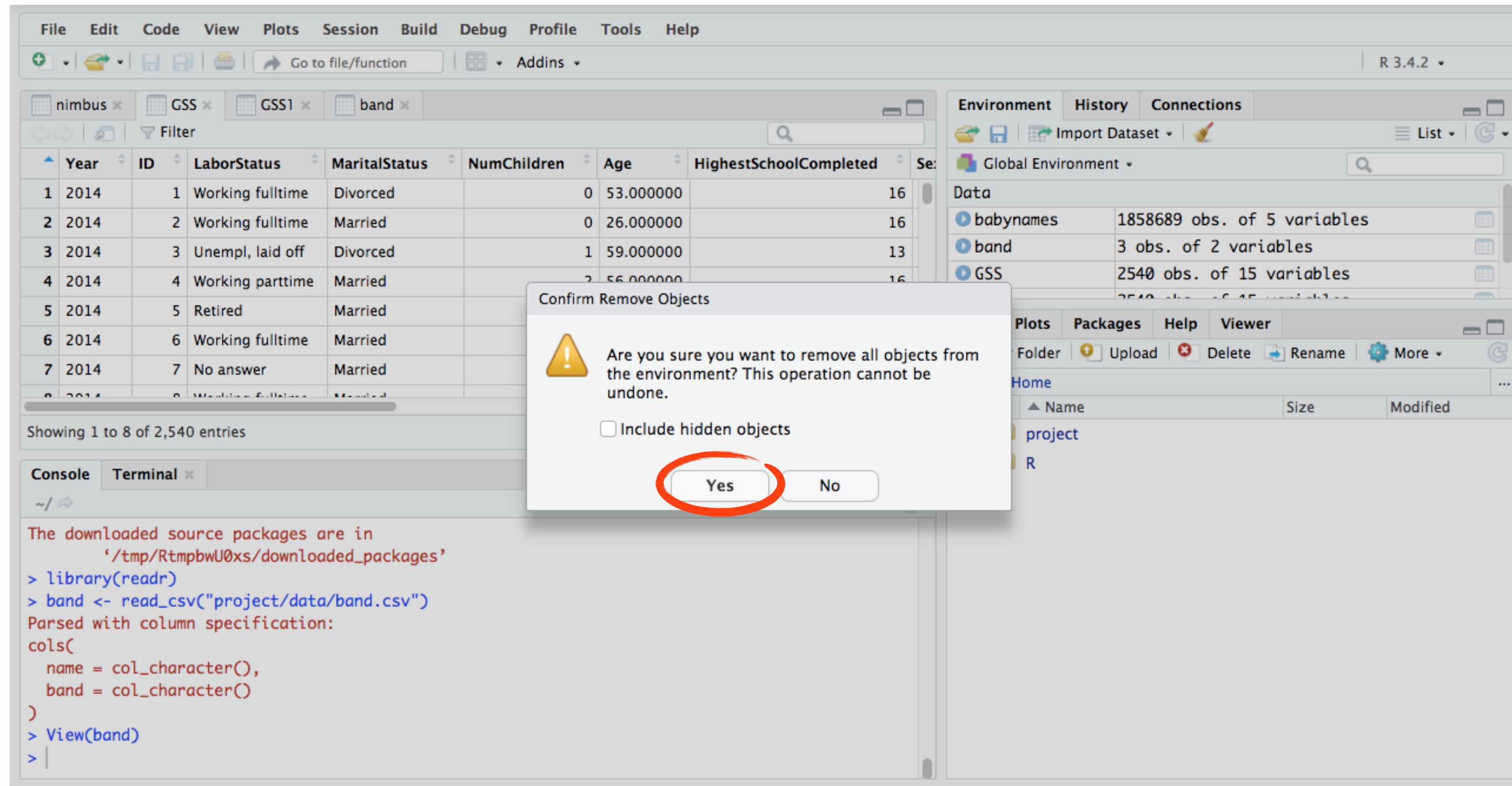


The screenshot shows the RStudio interface with the following components:

- Menu Bar:** File, Edit, Code, View, Plots, Session, Build, Debug, Profile, Tools, Help.
- Toolbar:** Includes icons for creating a new file, saving, and a search bar labeled "Go to file/function".
- Source Editor:** Displays a data frame with columns: Year, ID, LaborStatus, MaritalStatus, NumChildren, Age, HighestSchoolCompleted, and Sex. The first 8 rows are visible, showing data for the year 2014.
- Environment Pane:** Located on the right, it shows the "Global Environment" with a list of loaded objects: `babynames` (1858689 obs. of 5 variables), `band` (3 obs. of 2 variables), and `GSS` (2540 obs. of 15 variables). A red circle highlights the "Clean Environment" button (a broom icon) in the top right of this pane.
- Files Pane:** Located at the bottom right, it shows the file explorer with a tree view containing "project" and "R" folders.
- Console/Terminal:** At the bottom, it shows the R command prompt with the following code and output:

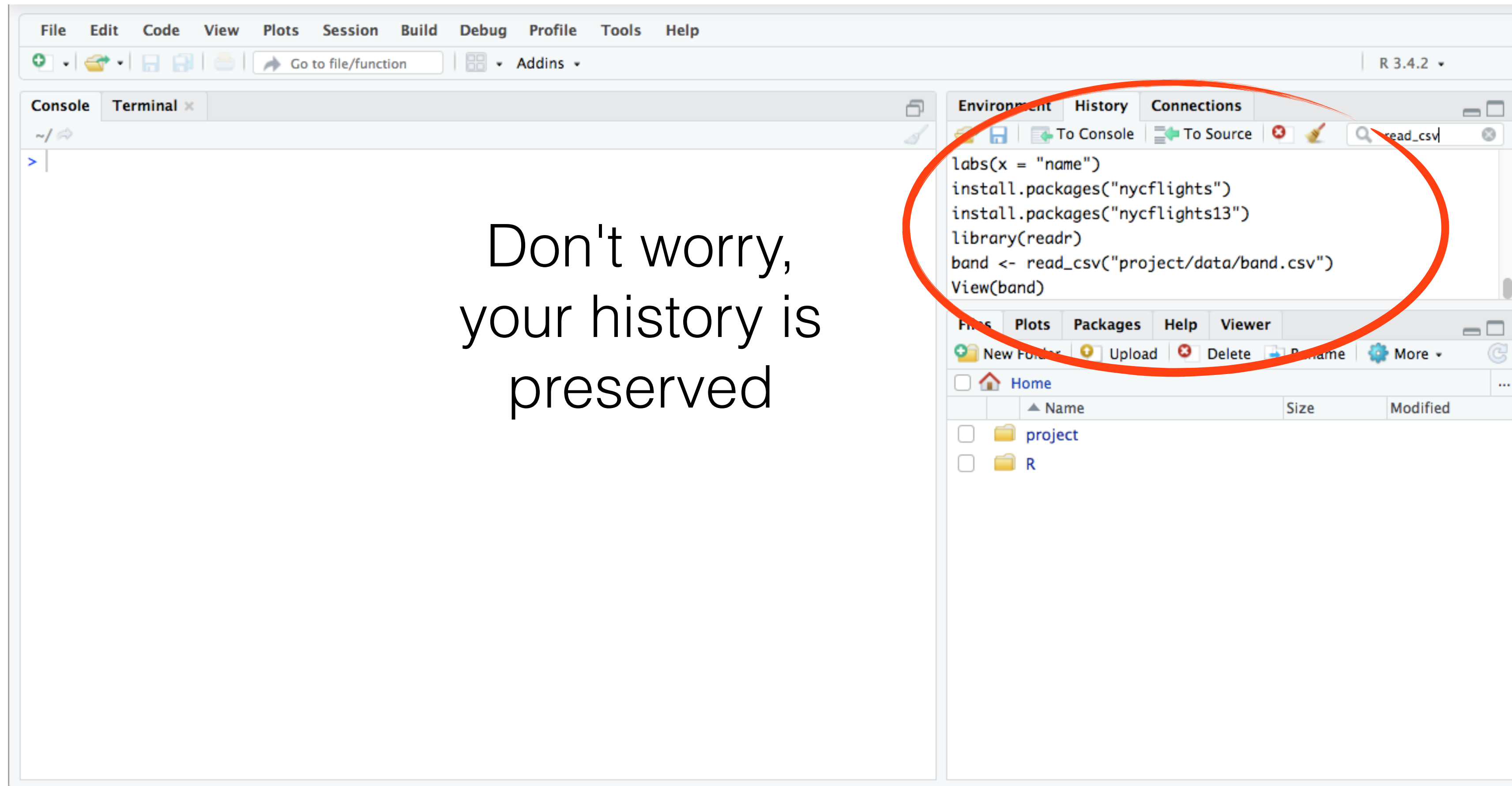
```
The downloaded source packages are in
'/tmp/RtmpbwU0xs/downloaded_packages'
> library(readr)
> band <- read_csv("project/data/band.csv")
Parsed with column specification:
cols(
  name = col_character(),
  band = col_character()
)
> View(band)
> |
```

It's okay to remove all, we've got the code

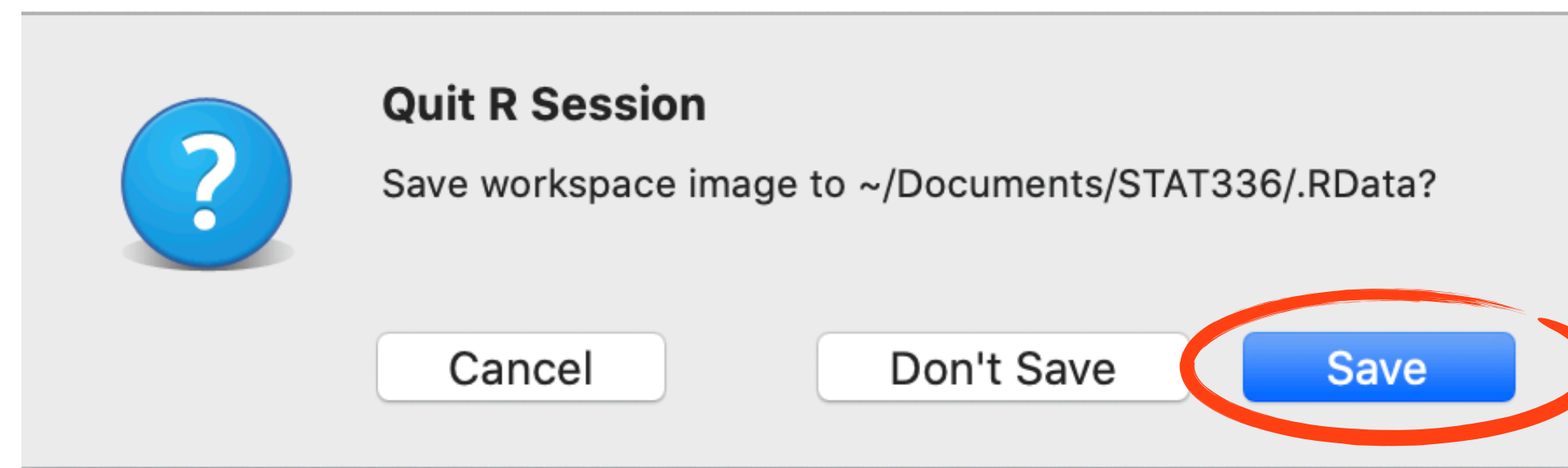


Your History is always preserved

Don't worry,
your history is
preserved

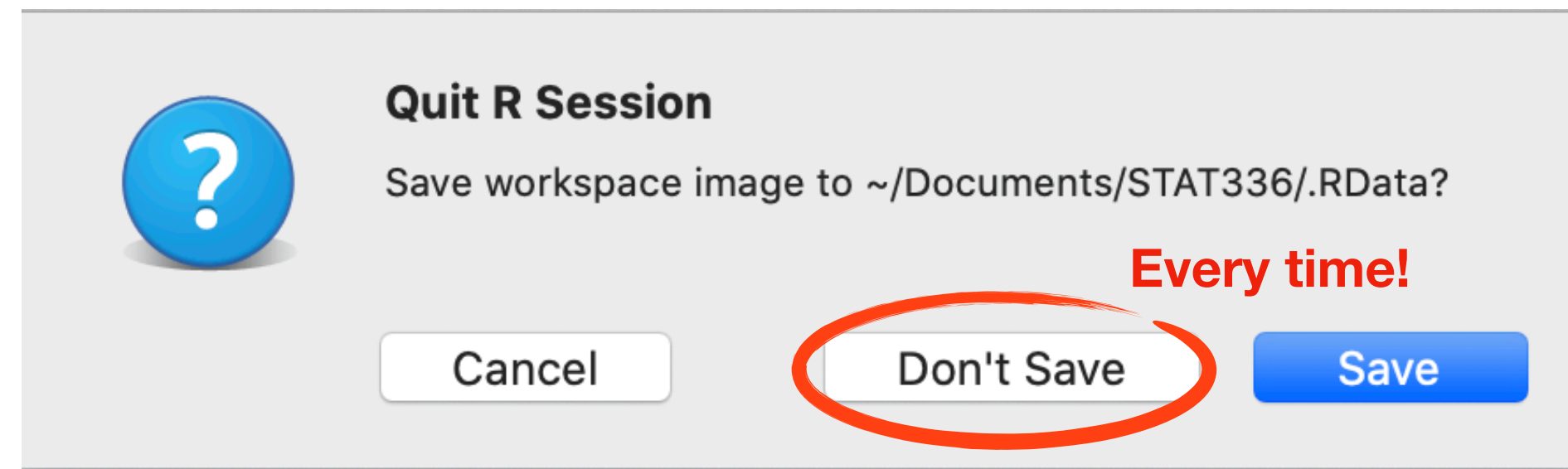


Now, quit RStudio and **this one time** say Save



Just this one time!!

Going forward, **always** say Don't Save



You also might want to close windows

