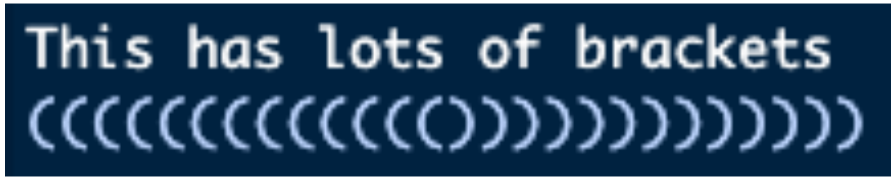
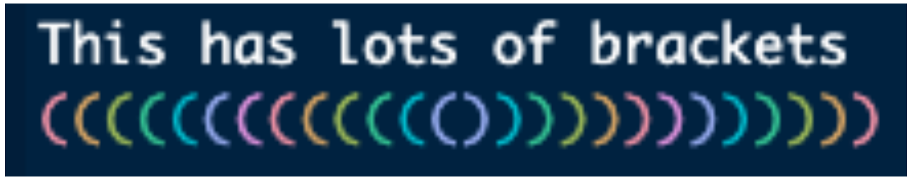
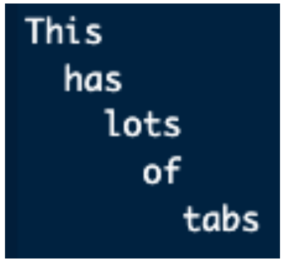
Rainbows



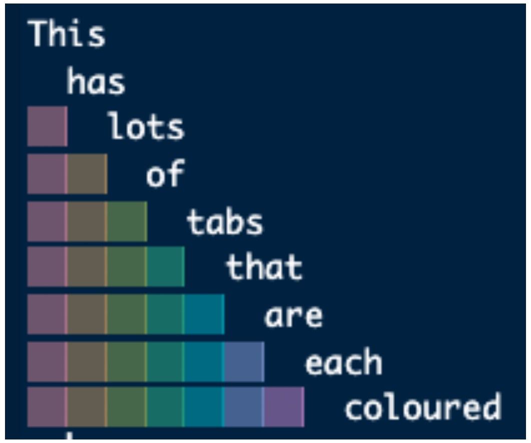
```
This has lots of brackets
((((((((((((()))))))))))))
```



```
This has lots of brackets
((((((((((((()))))))))))))
```

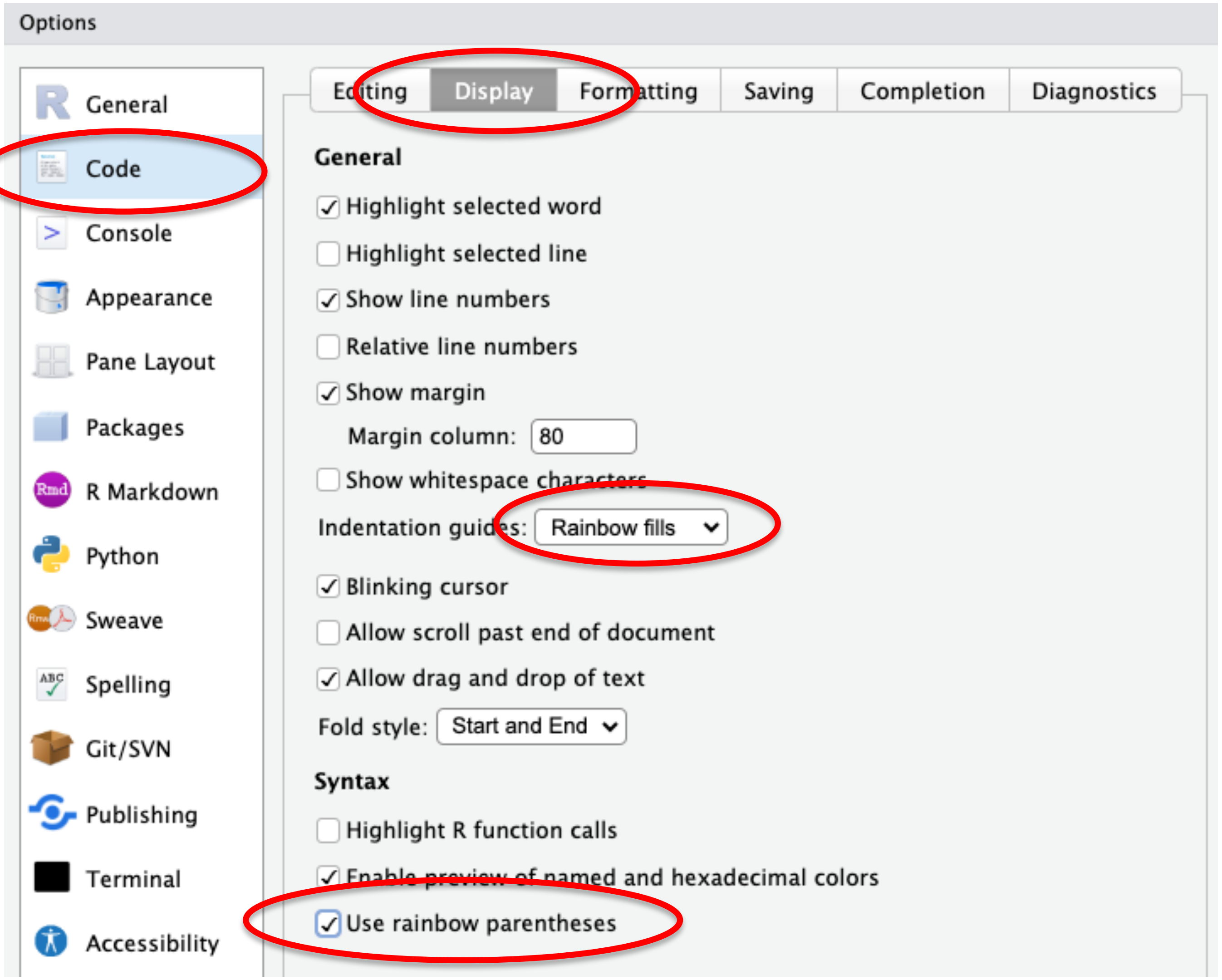


```
This
	has
		lots
			of
				tabs
```



```
This
	has
		lots
			of
				tabs
					that
						are
							each
								coloured
```

Rainbows



Options

General
Code
Console
Appearance
Pane Layout
Packages
R Markdown
Python
Sweave
Spelling
Git/SVN
Publishing
Terminal
Accessibility

Editing | Display | Formatting | Saving | Completion | Diagnostics

General

- [✓] Highlight selected word
- [ ] Highlight selected line
- [✓] Show line numbers
- [ ] Relative line numbers
- [✓] Show margin

Margin column: 80

- [ ] Show whitespace characters

Indentation guides: Rainbow fills

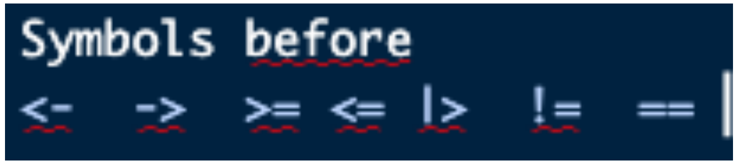
- [✓] Blinking cursor
- [ ] Allow scroll past end of document
- [✓] Allow drag and drop of text

Fold style: Start and End

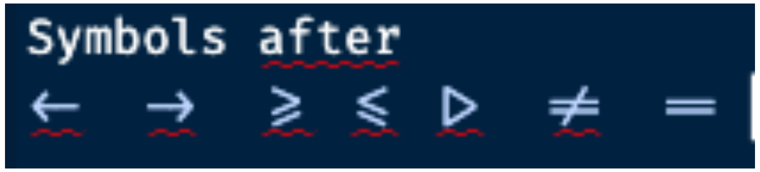
Syntax

- [ ] Highlight R function calls
- [✓] Enable preview of named and hexadecimal colors
- [✓] Use rainbow parentheses

Connected Symbols



```
Symbols before
<-   ->   >= <= |>   !=   ==
```

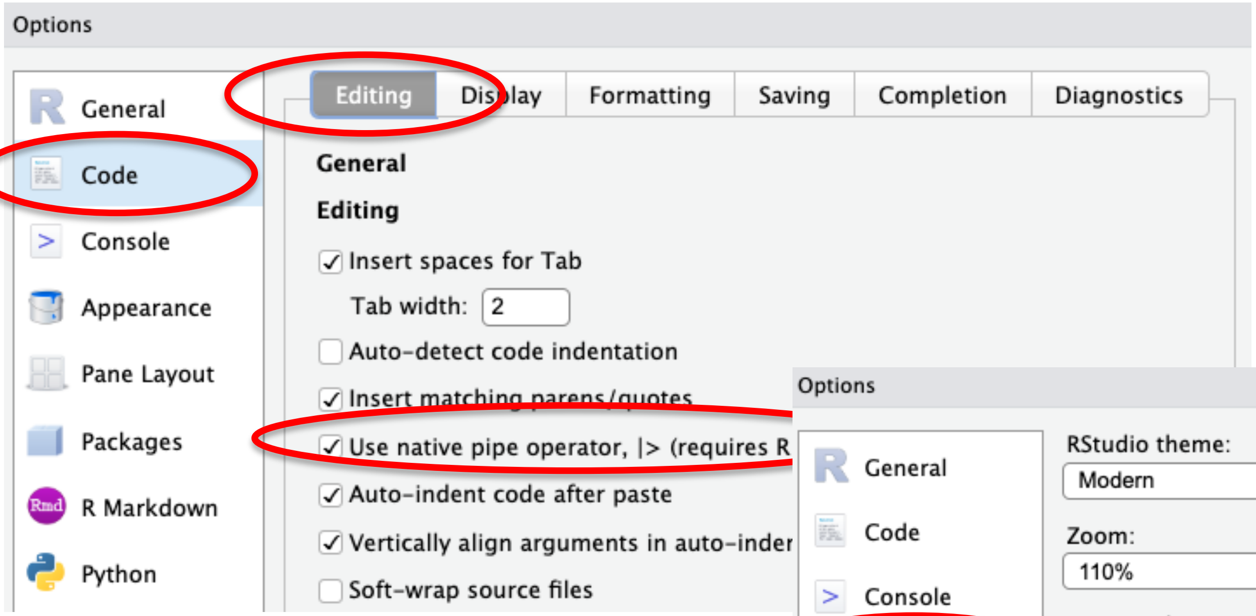


```
Symbols after
←   →   ≥ ≤ ▷   ≠   ==
```

Install font: ‘FiraCode-Regular’
https://github.com/tonsky/FiraCode

Install font: ‘JetBrainsMono-Regular’
https://www.jetbrains.com/lp/mono/

Connected Symbols



Options

General
Code
Console
Appearance
Pane Layout
Packages
R Markdown
Python

Editing | Display | Formatting | Saving | Completion | Diagnostics

General

Editing

[✓] Insert spaces for Tab

Tab width: 2

[ ] Auto-detect code indentation

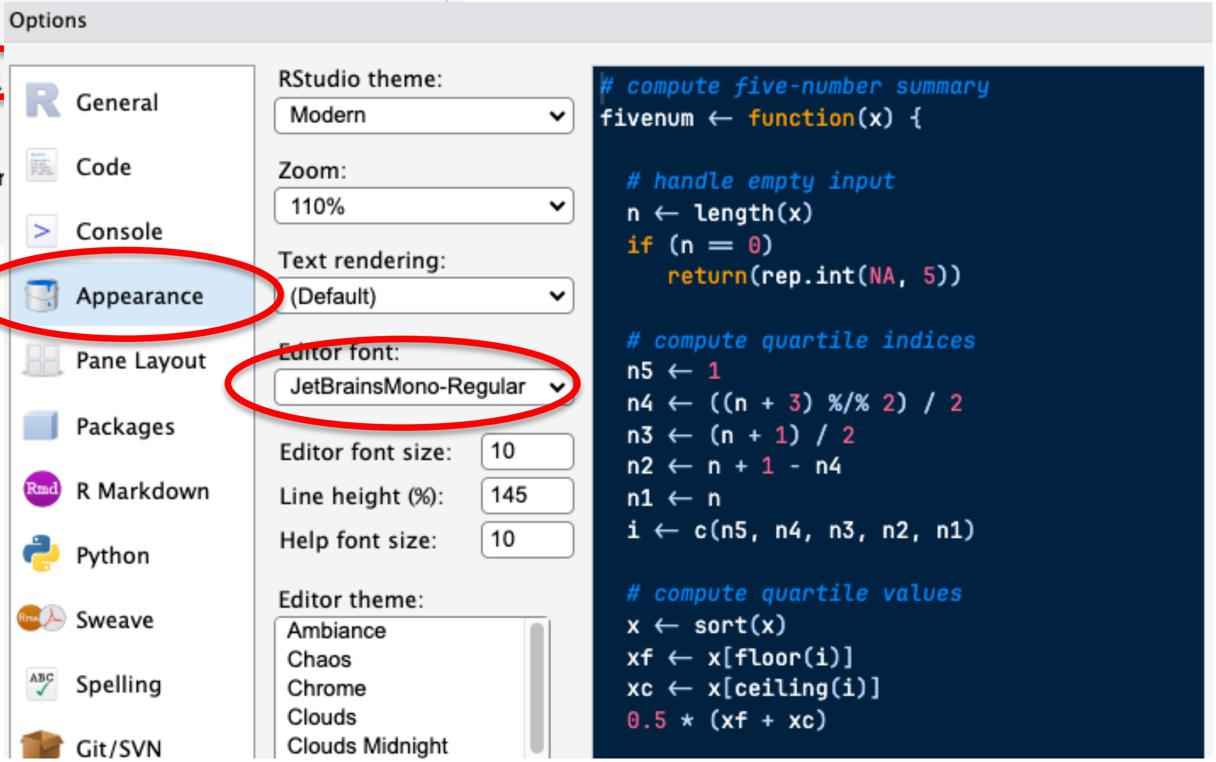
[✓] Insert matching parens/quotes

[✓] Use native pipe operator, |> (requires R

[✓] Auto-indent code after paste

[✓] Vertically align arguments in auto-inden

[ ] Soft-wrap source files



Options

General
Code
Console
Appearance
Pane Layout
Packages
R Markdown
Python
Sweave
Spelling
Git/SVN

RStudio theme: Modern

Zoom: 110%

Text rendering: (Default)

Editor font: JetBrainsMono-Regular

Editor font size: 10

Line height (%): 145

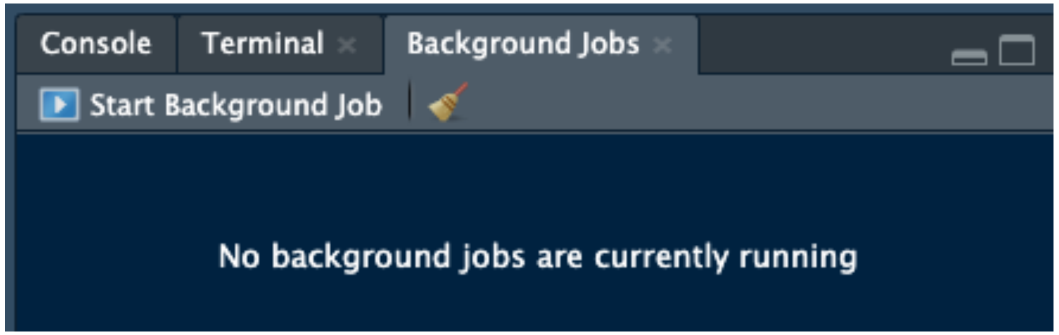
Help font size: 10

Editor theme:
Ambiance
Chaos
Chrome
Clouds
Clouds Midnight

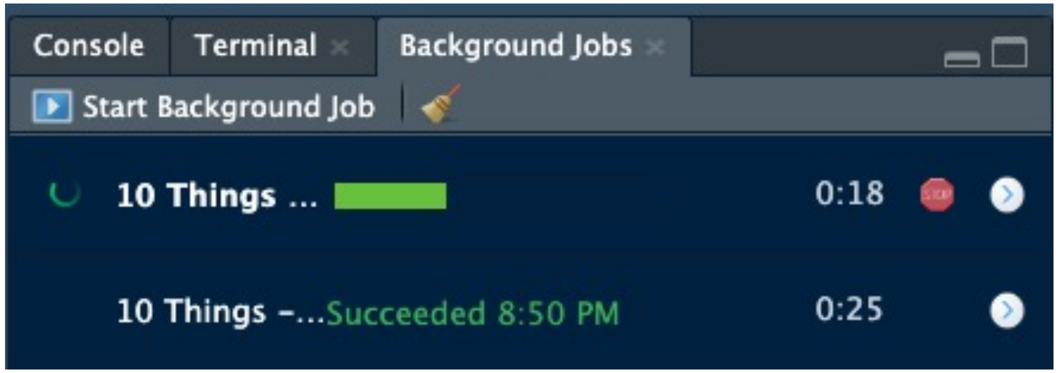
```
# compute five-number summary
fivenum ← function(x) {

  # handle empty input
  n ← length(x)
  if (n == 0)
    return(rep.int(NA, 5))

  # compute quartile indices
  n5 ← 1
  n4 ← ((n + 3) %/% 2) / 2
  n3 ← (n + 1) / 2
  n2 ← n + 1 - n4
  n1 ← n
  i ← c(n5, n4, n3, n2, n1)

  # compute quartile values
  x ← sort(x)
  xf ← x[floor(i)]
  xc ← x[ceiling(i)]
  0.5 * (xf + xc)
```

Background Jobs

RStudio®

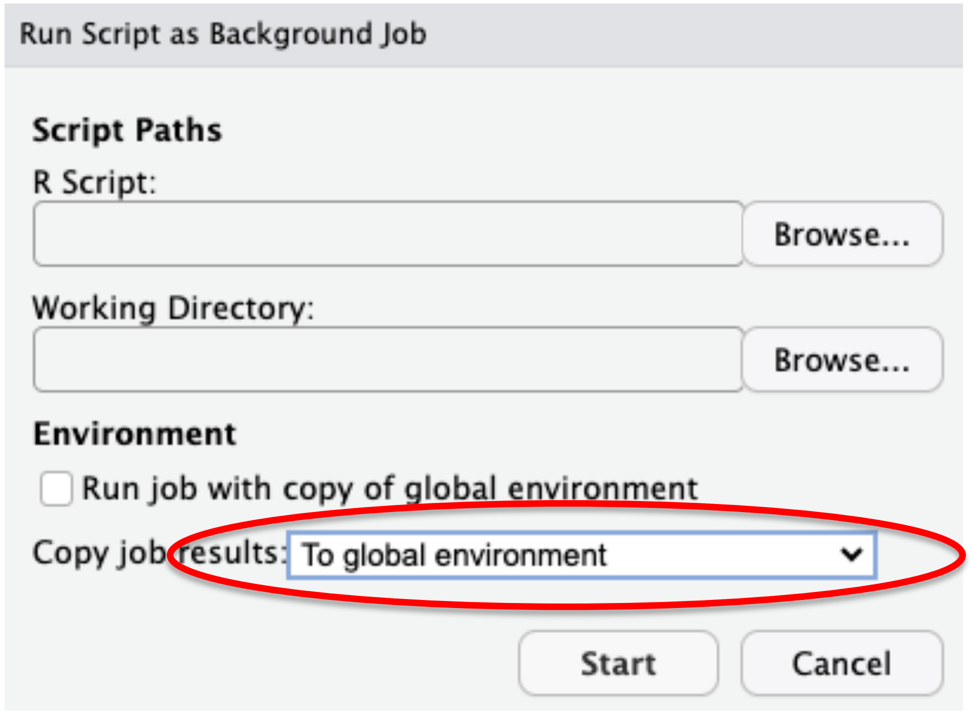


No background jobs are currently running



10 Things ... 0:18

10 Things –...Succeeded 8:50 PM 0:25



Run Script as Background Job

Script Paths

R Script:

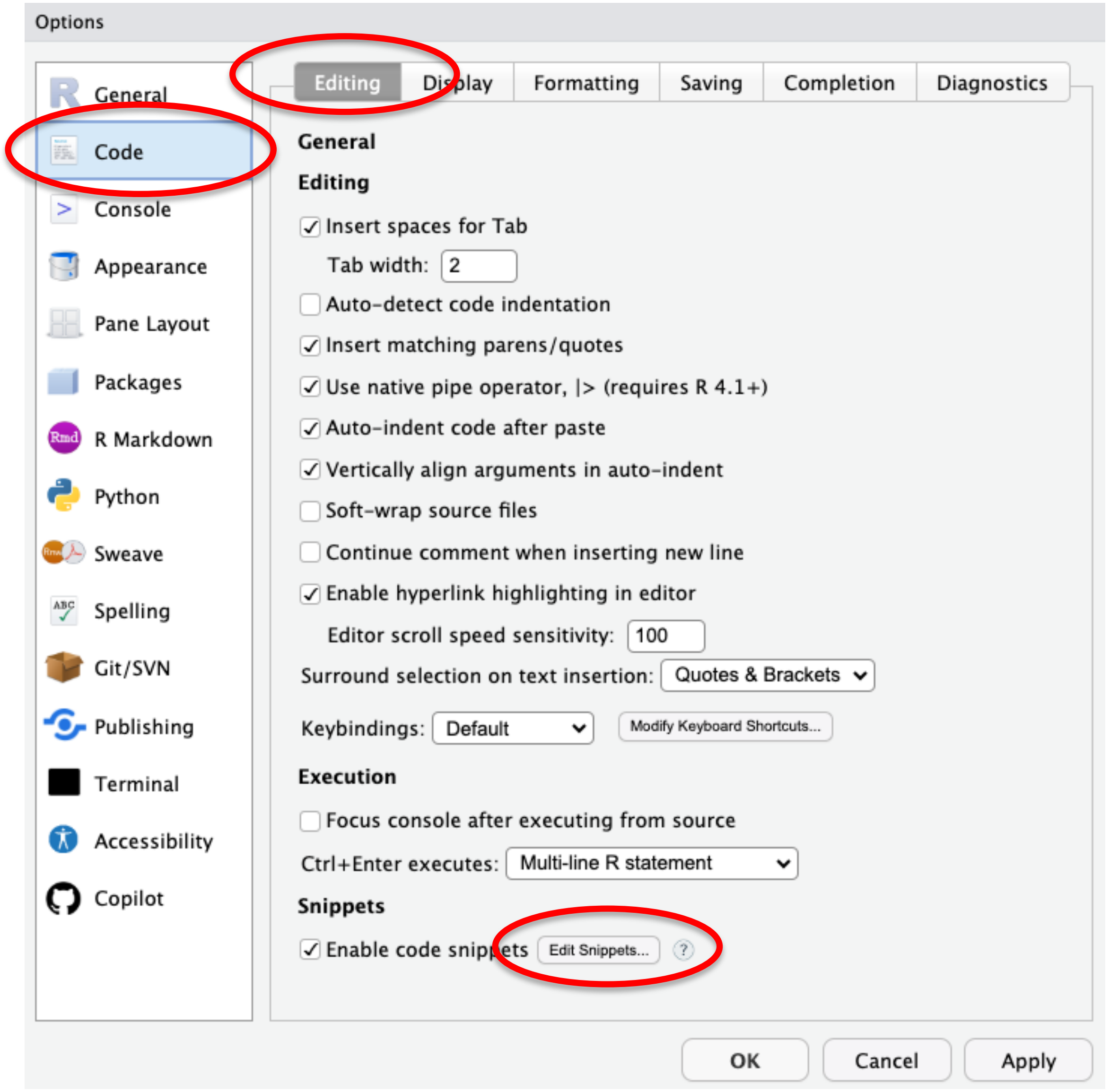
Working Directory:

Environment

☐ Run job with copy of global environment

Copy job results: To global environment

Snippets



Options

General

Editing

- [✓] Insert spaces for Tab
  - Tab width: 2
- [ ] Auto-detect code indentation
- [✓] Insert matching parens/quotes
- [✓] Use native pipe operator, |> (requires R 4.1+)
- [✓] Auto-indent code after paste
- [✓] Vertically align arguments in auto-indent
- [ ] Soft-wrap source files
- [ ] Continue comment when inserting new line
- [✓] Enable hyperlink highlighting in editor
  - Editor scroll speed sensitivity: 100

Surround selection on text insertion: Quotes & Brackets

Keybindings: Default

Execution

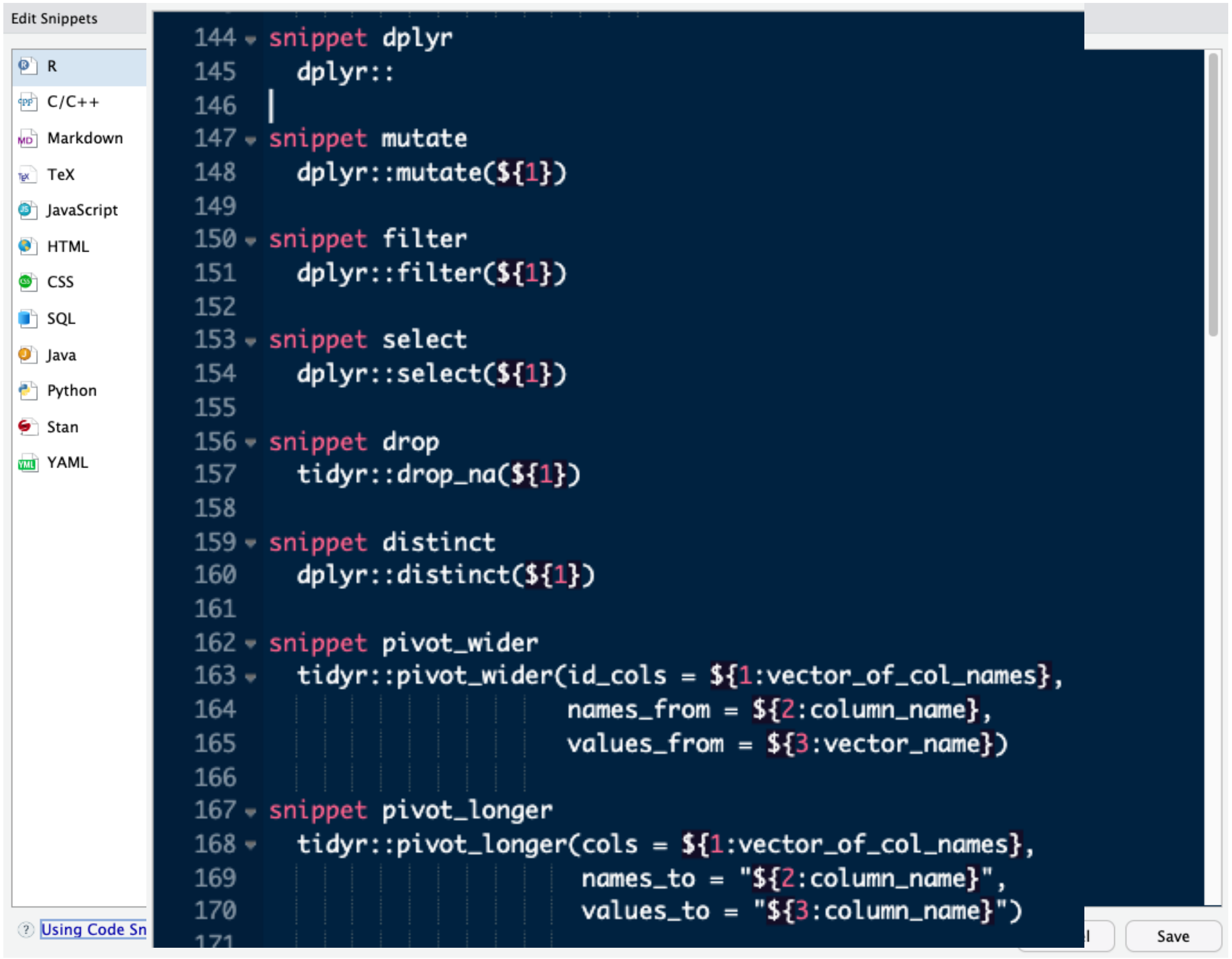
- [ ] Focus console after executing from source

Ctrl+Enter executes: Multi-line R statement

Snippets

- [✓] Enable code snippets

Snippets



```
snippet dplyr
  dplyr::

snippet mutate
  dplyr::mutate(${1})

snippet filter
  dplyr::filter(${1})

snippet select
  dplyr::select(${1})

snippet drop
  tidyr::drop_na(${1})

snippet distinct
  dplyr::distinct(${1})

snippet pivot_wider
  tidyr::pivot_wider(id_cols = ${1:vector_of_col_names},
                     names_from = ${2:column_name},
                     values_from = ${3:vector_name})

snippet pivot_longer
  tidyr::pivot_longer(cols = ${1:vector_of_col_names},
                      names_to = "${2:column_name}",
                      values_to = "${3:column_name}")
```

Git Version Control

Git was created in 2005 by Linus Torvalds. He writes....

git can mean anything, depending on your mood.....

- random three-letter combination that is pronounceable, and not actually used by any common UNIX command. The fact that it is a mispronunciation of **get** may or may not be relevant.

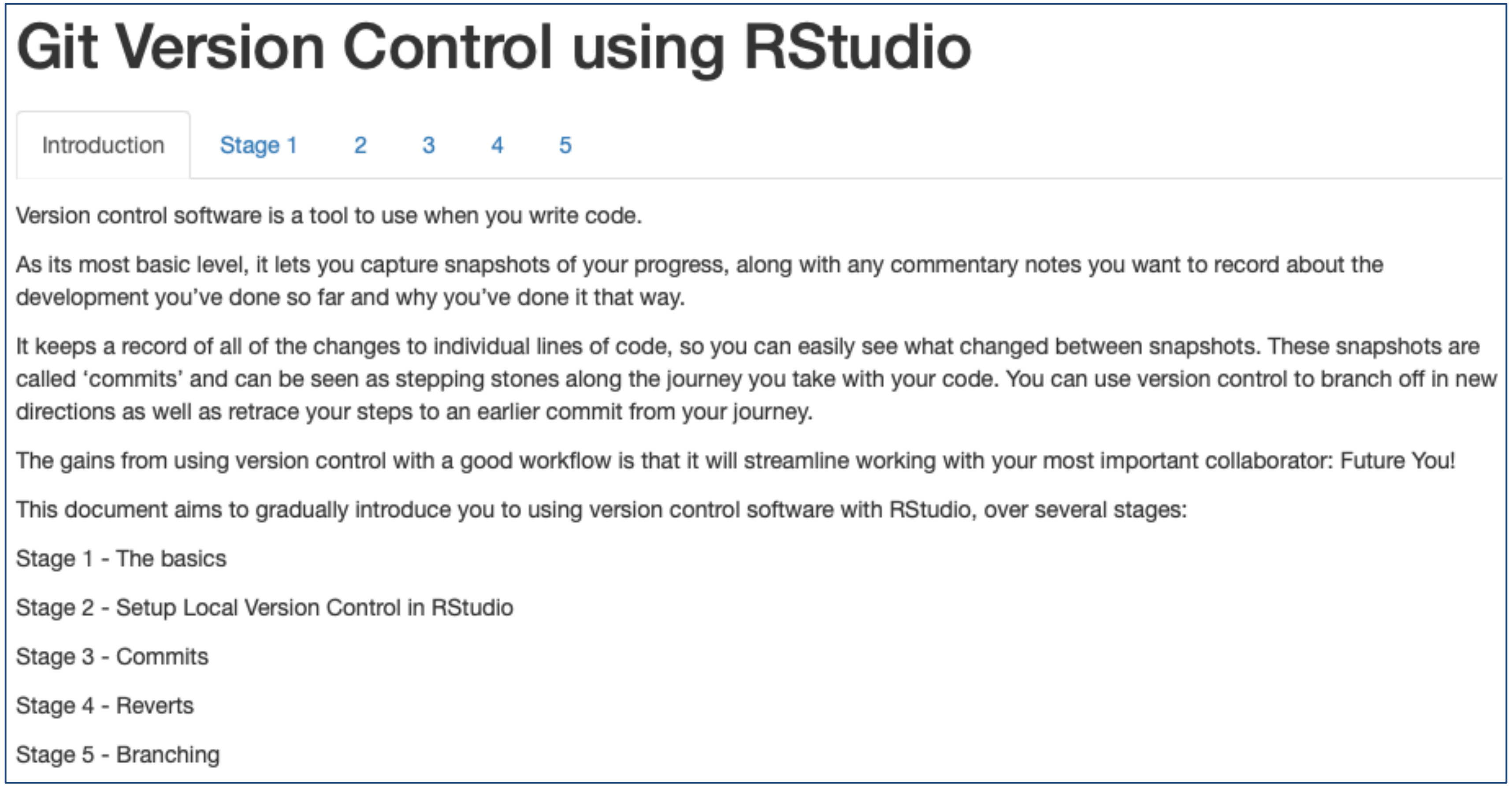
- stupid. contemptible and despicable. simple. Take your pick from the dictionary of slang

- "**g**lobal **i**nformation **t**racker": you're in a good mood, and it actually works for you. Angels sing, and a light suddenly fills the room

- "**g**oddamn **i**diotic **t**ruckload of sh*t": when it breaks

This is a stupid (but extremely fast) directory content manager. It doesn't do a whole lot, but what it **does** do is track directory contents efficiently.

Git Version Control



Git Version Control using RStudio

Introduction Stage 1 2 3 4 5

Version control software is a tool to use when you write code.

As its most basic level, it lets you capture snapshots of your progress, along with any commentary notes you want to record about the development you've done so far and why you've done it that way.

It keeps a record of all of the changes to individual lines of code, so you can easily see what changed between snapshots. These snapshots are called 'commits' and can be seen as stepping stones along the journey you take with your code. You can use version control to branch off in new directions as well as retrace your steps to an earlier commit from your journey.

The gains from using version control with a good workflow is that it will streamline working with your most important collaborator: Future You!

This document aims to gradually introduce you to using version control software with RStudio, over several stages:

Stage 1 - The basics

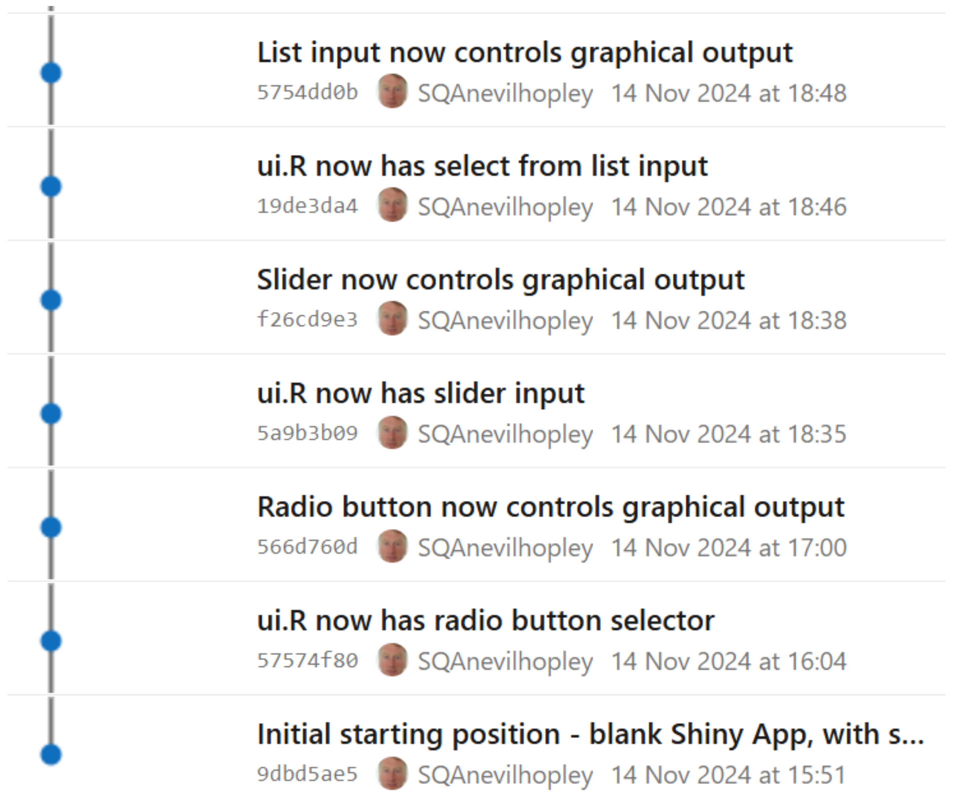
Stage 2 - Setup Local Version Control in RStudio

Stage 3 - Commits

Stage 4 - Reverts

Stage 5 - Branching

github.com/NevilHopley/tenthings



List input now controls graphical output
5754dd0b SQAnevilhopley 14 Nov 2024 at 18:48

ui.R now has select from list input
19de3da4 SQAnevilhopley 14 Nov 2024 at 18:46

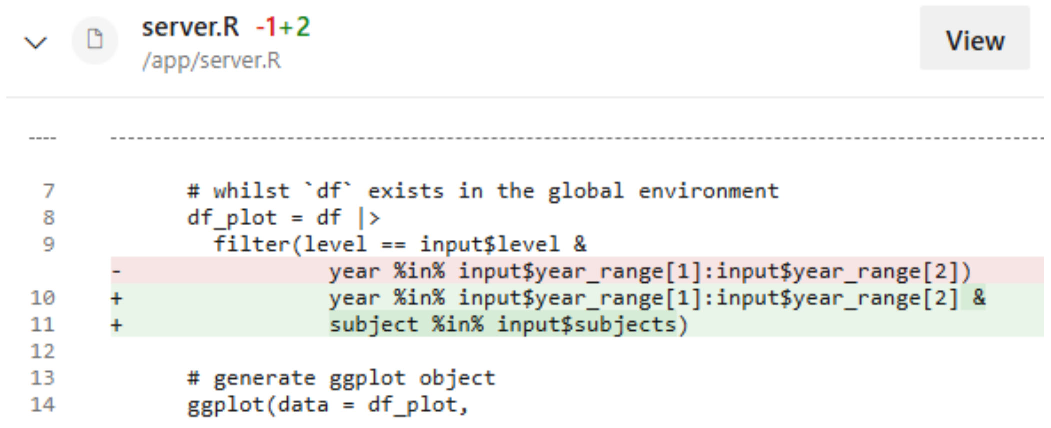
Slider now controls graphical output
f26cd9e3 SQAnevilhopley 14 Nov 2024 at 18:38

ui.R now has slider input
5a9b3b09 SQAnevilhopley 14 Nov 2024 at 18:35

Radio button now controls graphical output
566d760d SQAnevilhopley 14 Nov 2024 at 17:00

ui.R now has radio button selector
57574f80 SQAnevilhopley 14 Nov 2024 at 16:04

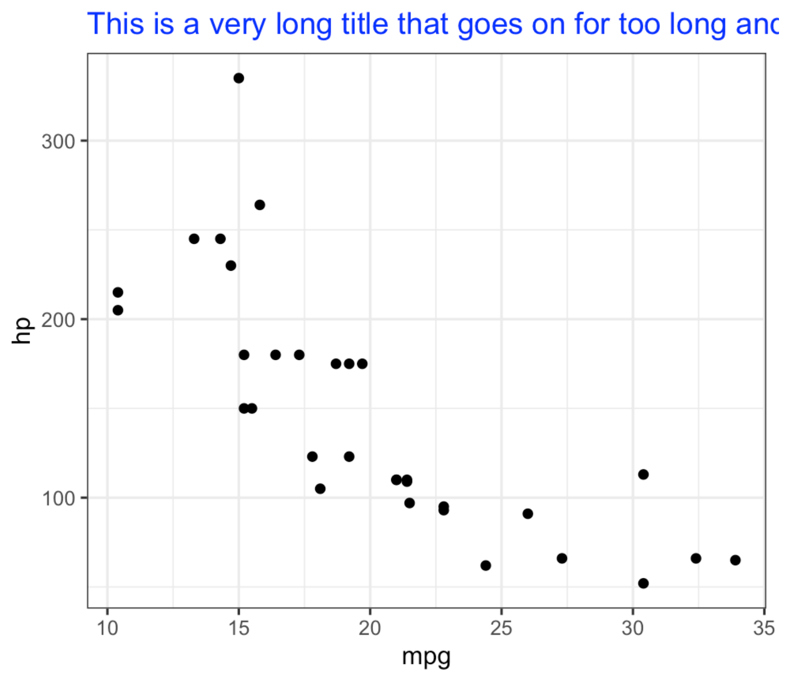
Initial starting position - blank Shiny App, with s...
9dbd5ae5 SQAnevilhopley 14 Nov 2024 at 15:51

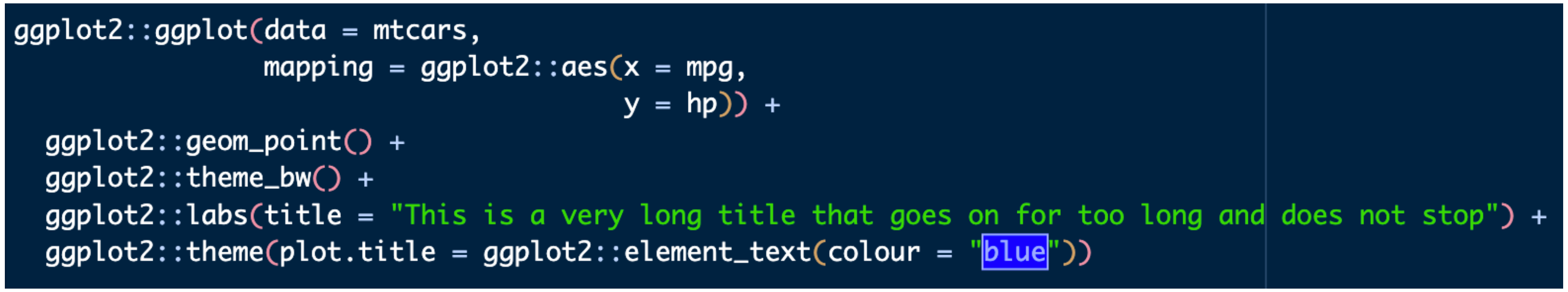


server.R -1+2
/app/server.R

```
      # whilst `df` exists in the global environment
      df_plot = df |>
        filter(level == input$level &
-                year %in% input$year_range[1]:input$year_range[2])
+                year %in% input$year_range[1]:input$year_range[2] &
+                subject %in% input$subjects)

      # generate ggplot object
      ggplot(data = df_plot,
```

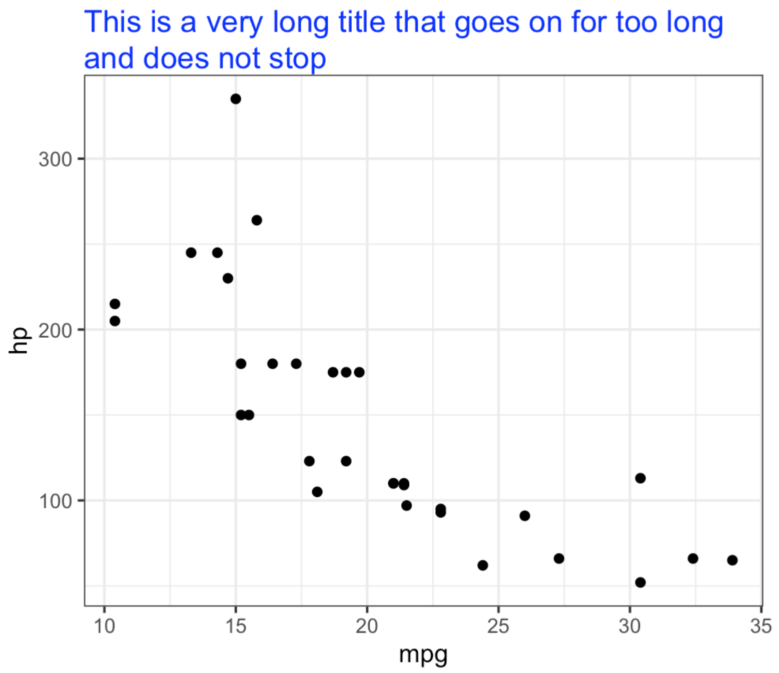
Titles and Legends

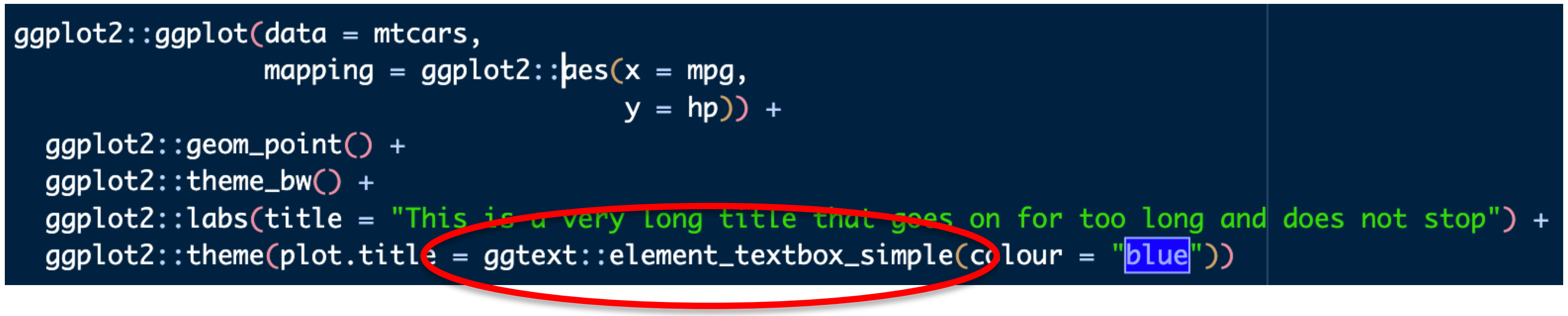




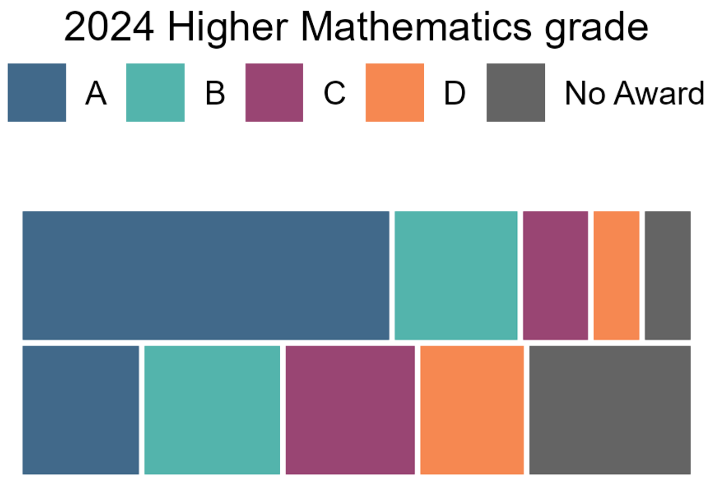
```
ggplot2::ggplot(data = mtcars,
                mapping = ggplot2::aes(x = mpg,
                                       y = hp)) +
  ggplot2::geom_point() +
  ggplot2::theme_bw() +
  ggplot2::labs(title = "This is a very long title that goes on for too long and does not stop") +
  ggplot2::theme(plot.title = ggplot2::element_text(colour = "blue"))
```

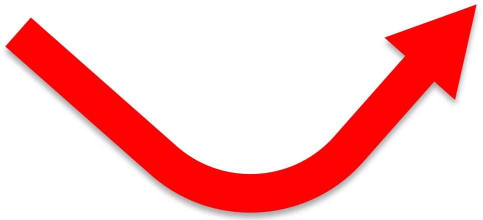
Titles and Legends

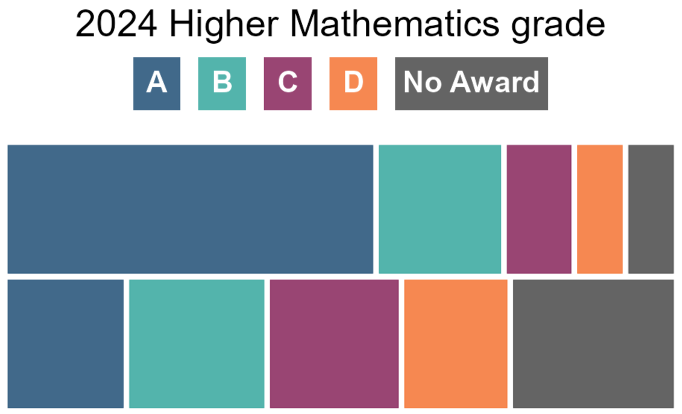


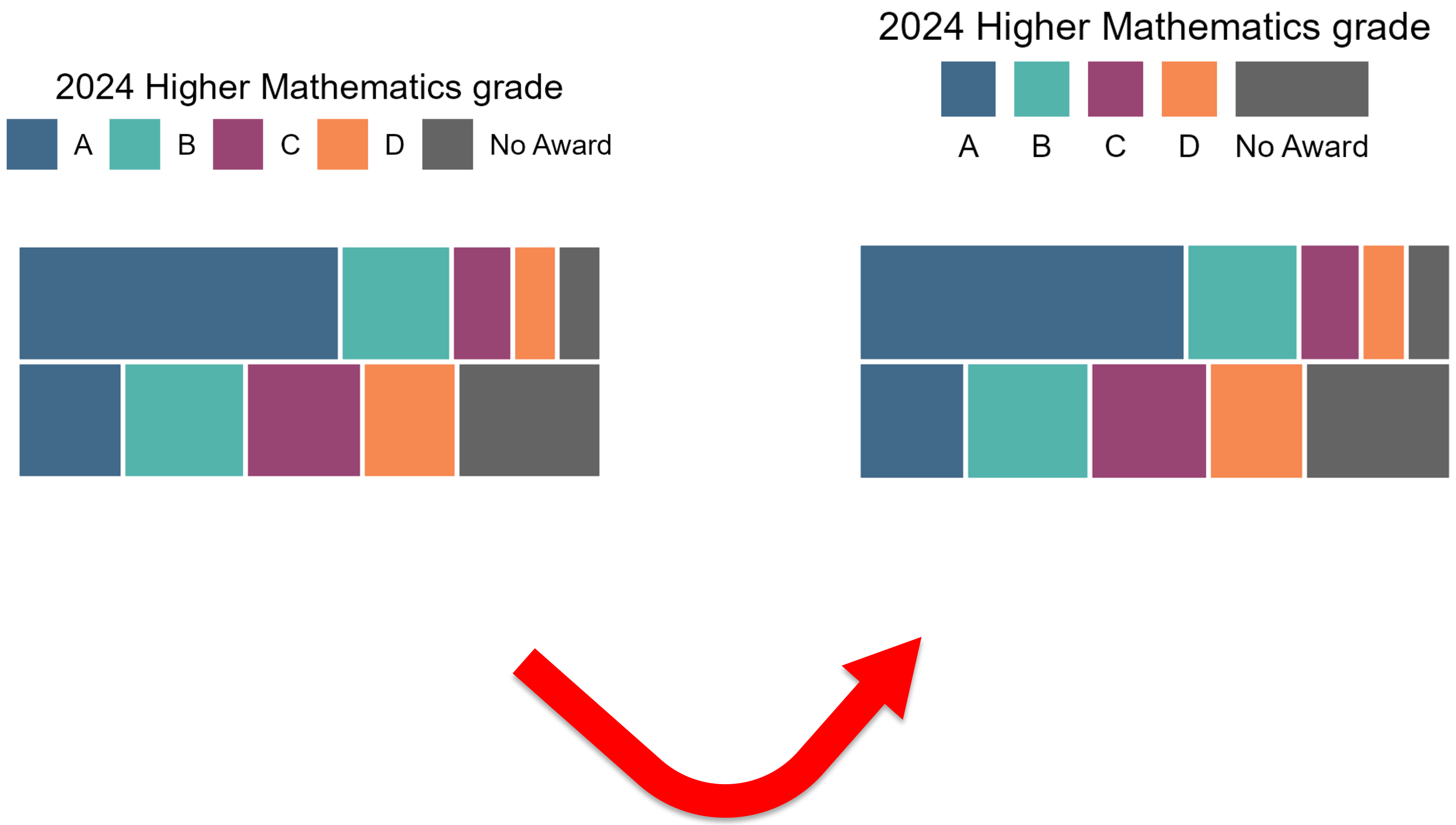


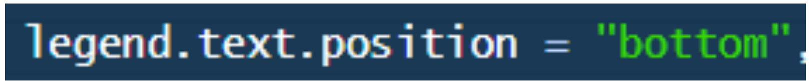
```
ggplot2::ggplot(data = mtcars,
                mapping = ggplot2::aes(x = mpg,
                                       y = hp)) +
  ggplot2::geom_point() +
  ggplot2::theme_bw() +
  ggplot2::labs(title = "This is a very long title that goes on for too long and does not stop") +
  ggplot2::theme(plot.title = ggtext::element_textbox_simple(colour = "blue"))
```


2024 Higher Mathematics grade
A
B
C
D
No Award

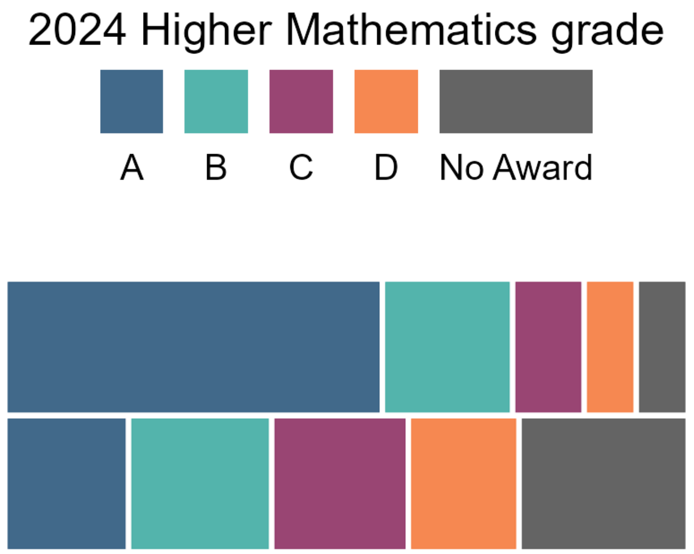


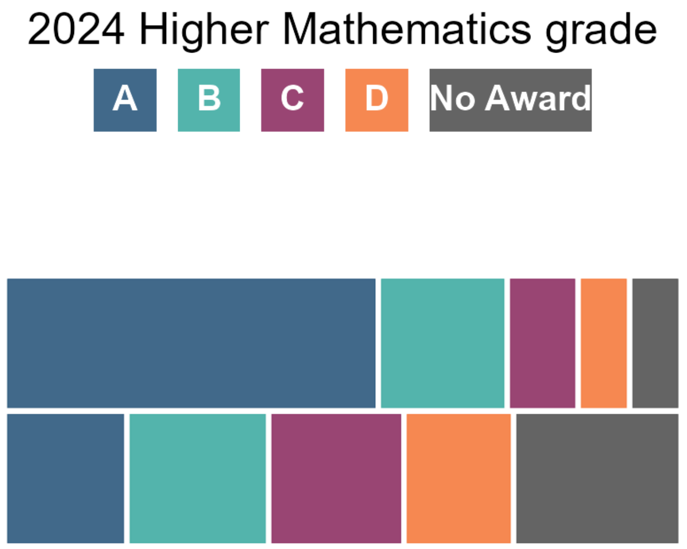

2024 Higher Mathematics grade
A
B
C
D
No Award

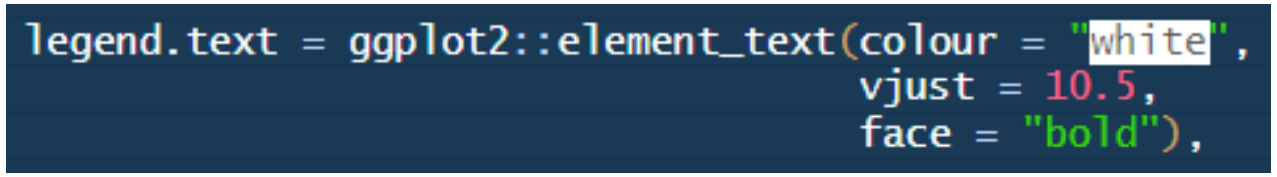




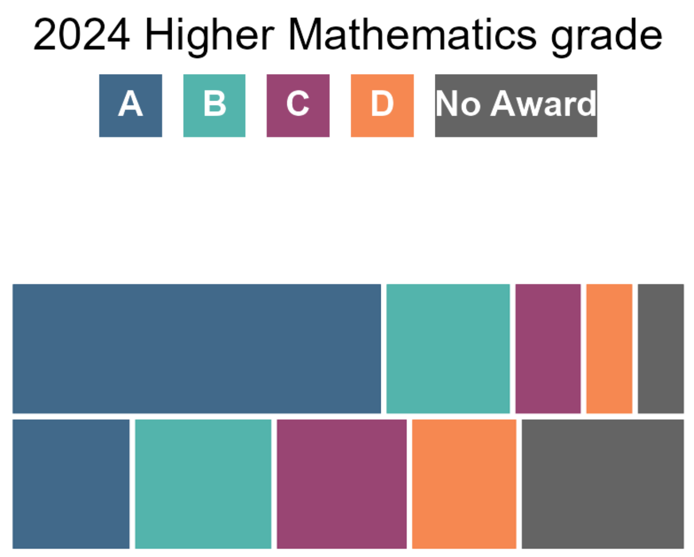
```
legend.text.position = "bottom",
```

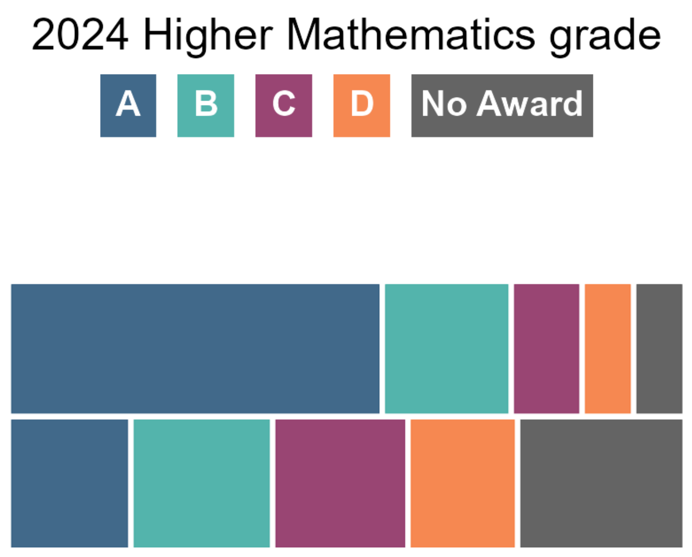


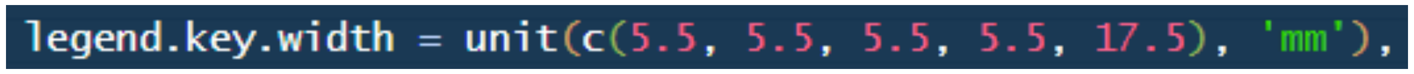




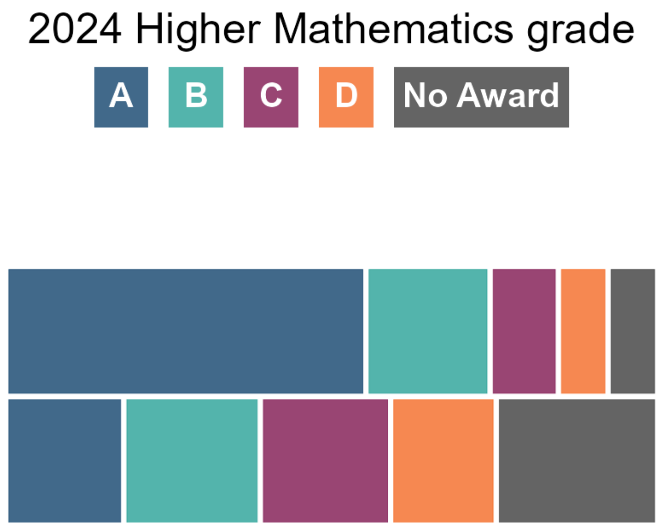
```
legend.text = ggplot2::element_text(colour = "white",
                                    vjust = 10.5,
                                    face = "bold"),
```

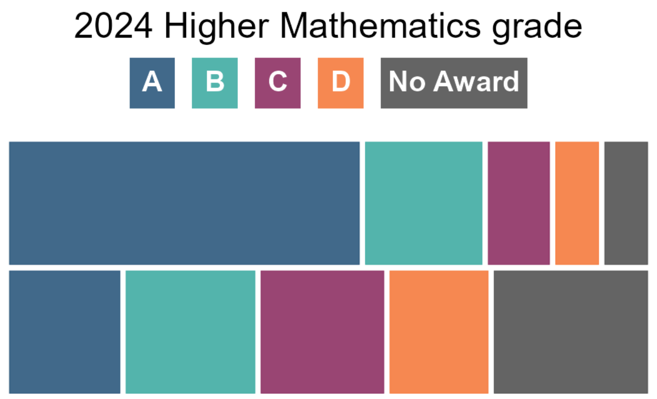


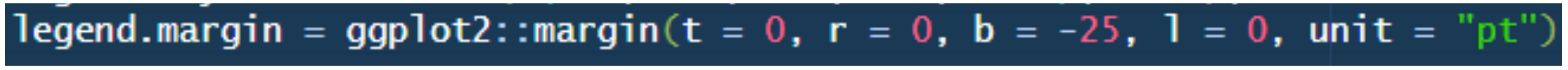




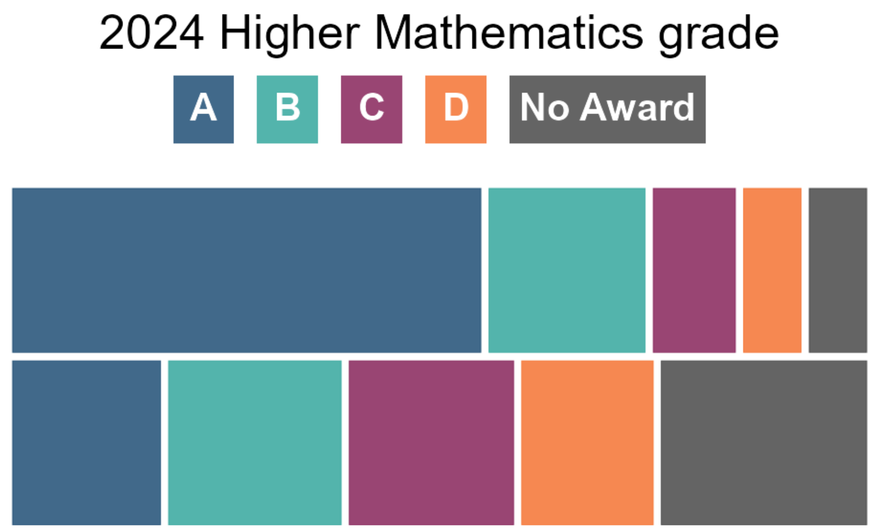
```
legend.key.width = unit(c(5.5, 5.5, 5.5, 5.5, 17.5), 'mm'),
```

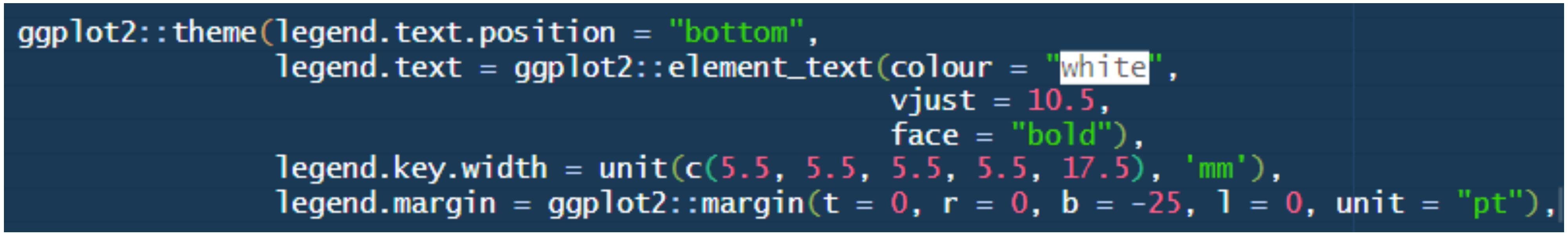






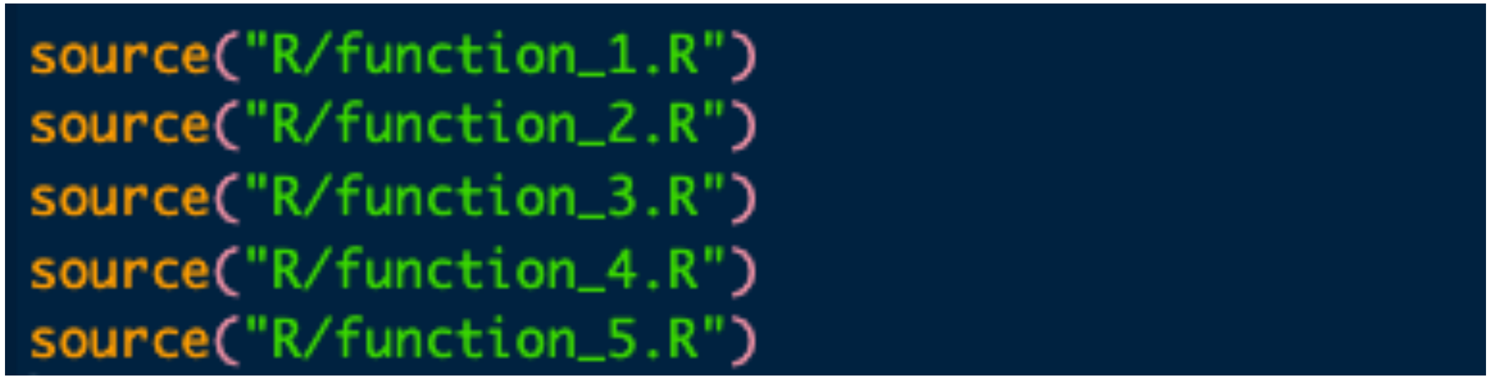
```
legend.margin = ggplot2::margin(t = 0, r = 0, b = -25, l = 0, unit = "pt")
```



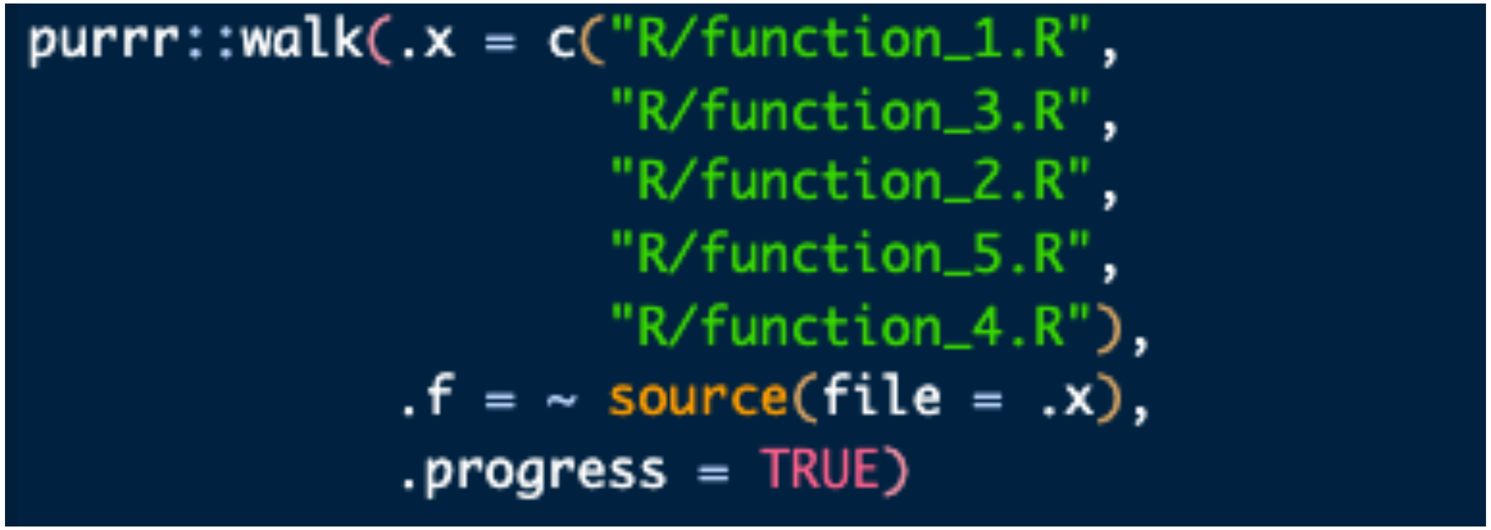


```
ggplot2::theme(legend.text.position = "bottom",
               legend.text = ggplot2::element_text(colour = "white",
                                                    vjust = 10.5,
                                                    face = "bold"),
               legend.key.width = unit(c(5.5, 5.5, 5.5, 5.5, 17.5), 'mm'),
               legend.margin = ggplot2::margin(t = 0, r = 0, b = -25, l = 0, unit = "pt"),
```

Multiple Sourcing

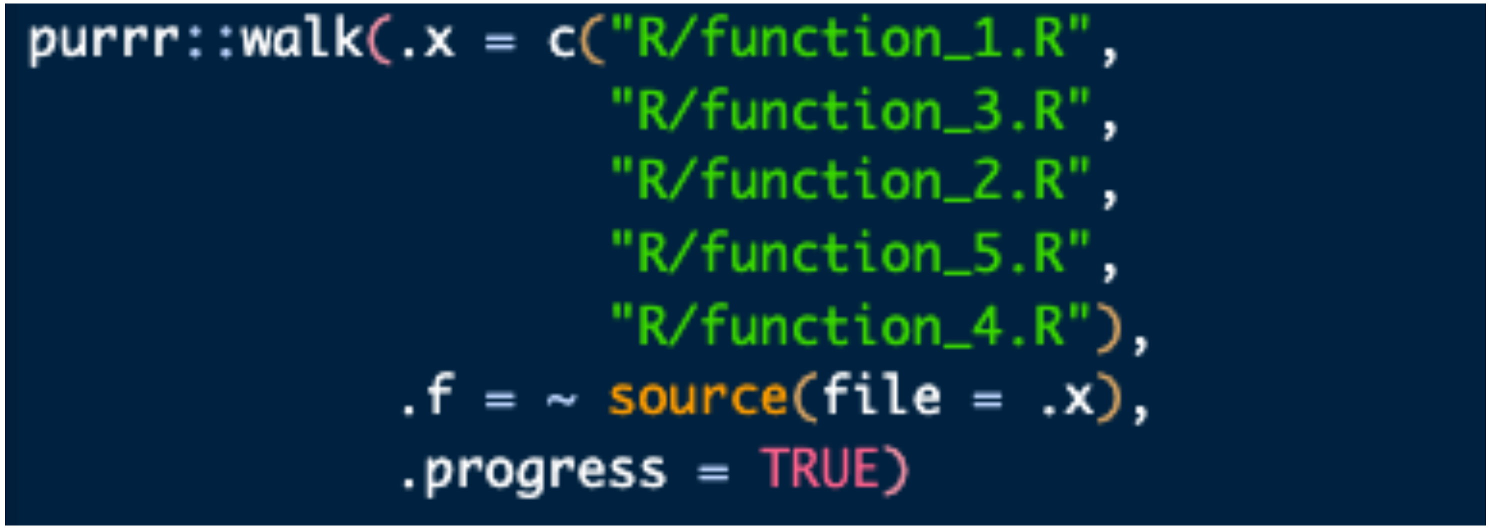


```r
source("R/function_1.R")
source("R/function_2.R")
source("R/function_3.R")
source("R/function_4.R")
source("R/function_5.R")
```

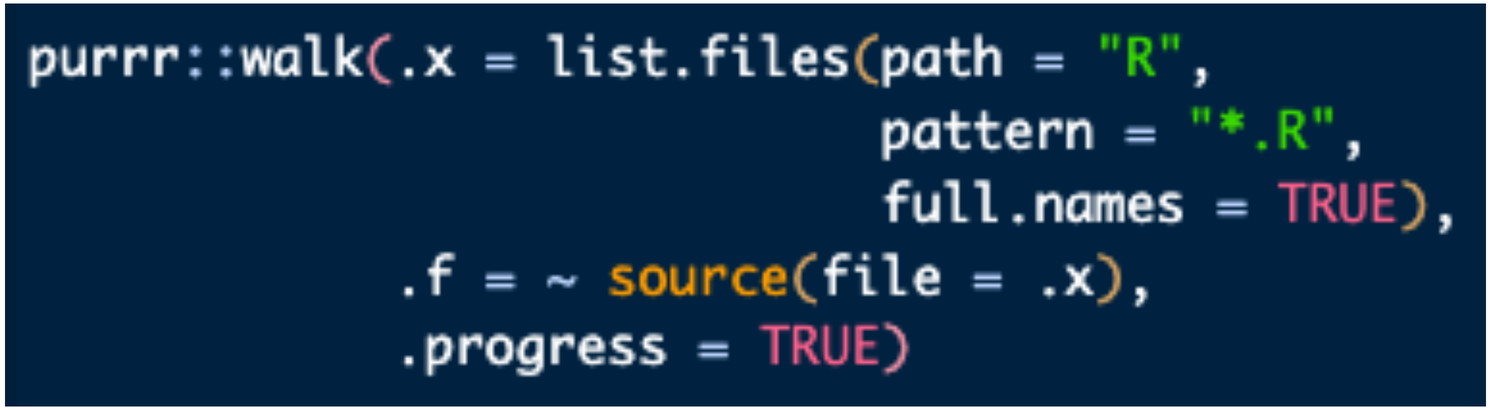


```r
purrr::walk(.x = c("R/function_1.R",
                   "R/function_3.R",
                   "R/function_2.R",
                   "R/function_5.R",
                   "R/function_4.R"),
            .f = ~ source(file = .x),
            .progress = TRUE)
```

Multiple Sourcing

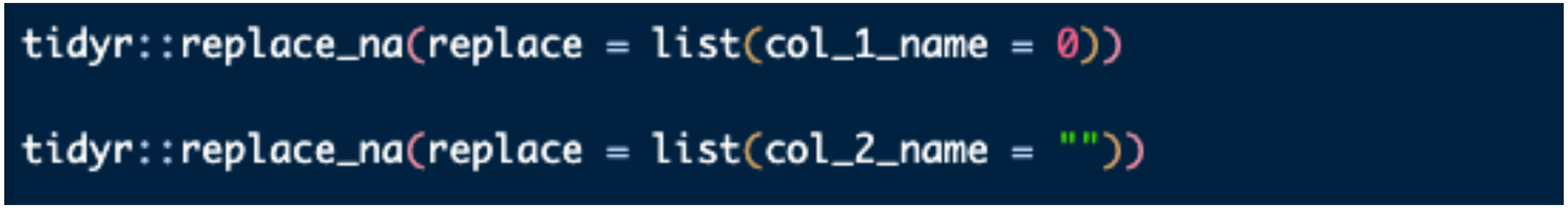


```r
purrr::walk(.x = c("R/function_1.R",
                   "R/function_3.R",
                   "R/function_2.R",
                   "R/function_5.R",
                   "R/function_4.R"),
            .f = ~ source(file = .x),
            .progress = TRUE)
```

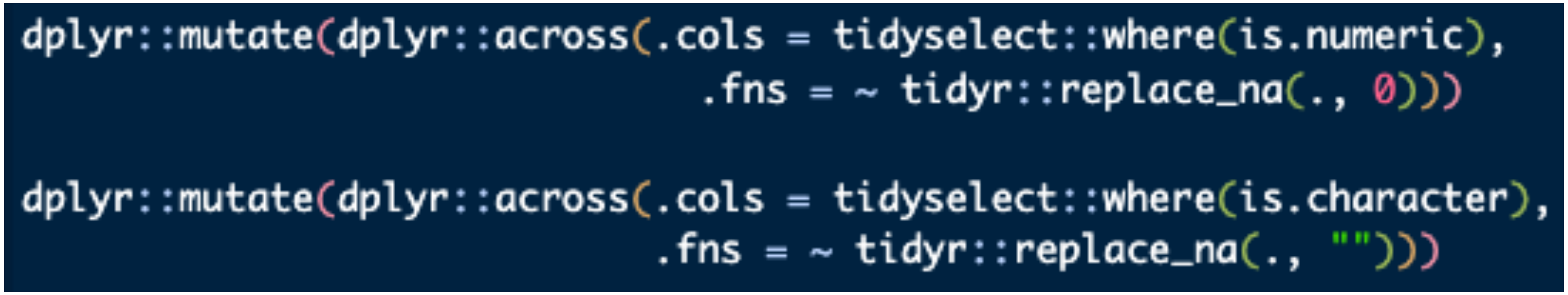


```r
purrr::walk(.x = list.files(path = "R",
                            pattern = "*.R",
                            full.names = TRUE),
            .f = ~ source(file = .x),
            .progress = TRUE)
```

Replace all NAs

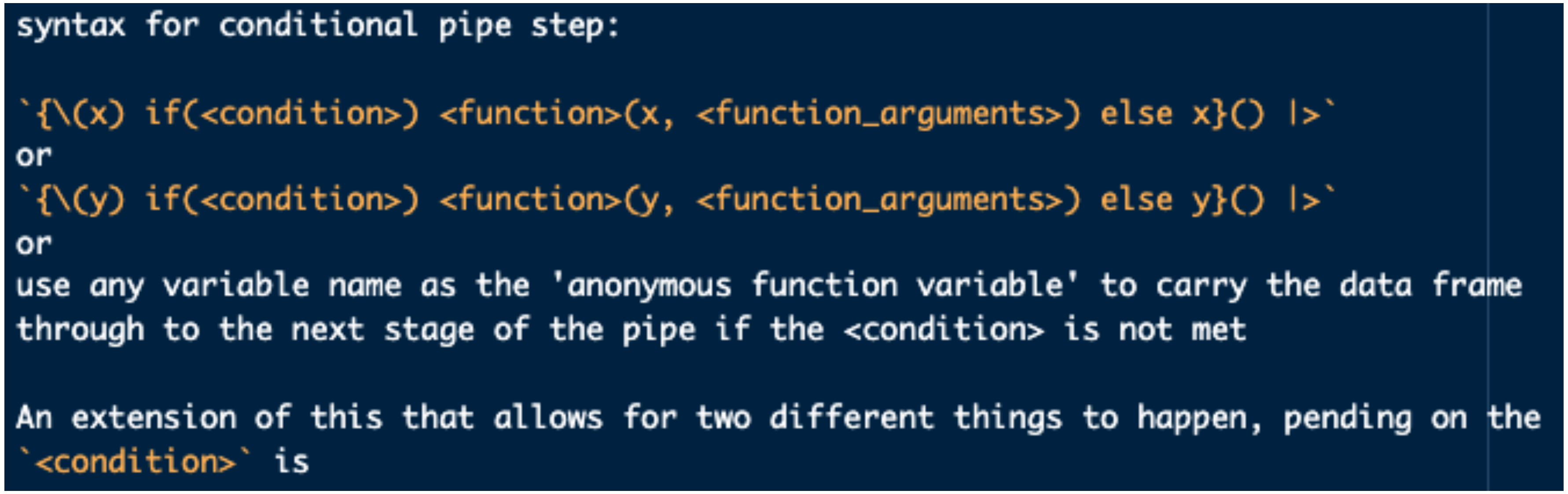


```r
tidyr::replace_na(replace = list(col_1_name = 0))

tidyr::replace_na(replace = list(col_2_name = ""))
```

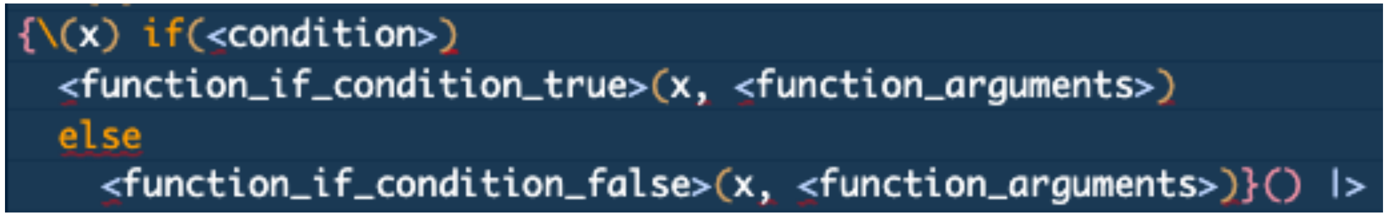


```r
dplyr::mutate(dplyr::across(.cols = tidyselect::where(is.numeric),
                              .fns = ~ tidyr::replace_na(., 0)))

dplyr::mutate(dplyr::across(.cols = tidyselect::where(is.character),
                            .fns = ~ tidyr::replace_na(., "")))
```

Conditional Piping



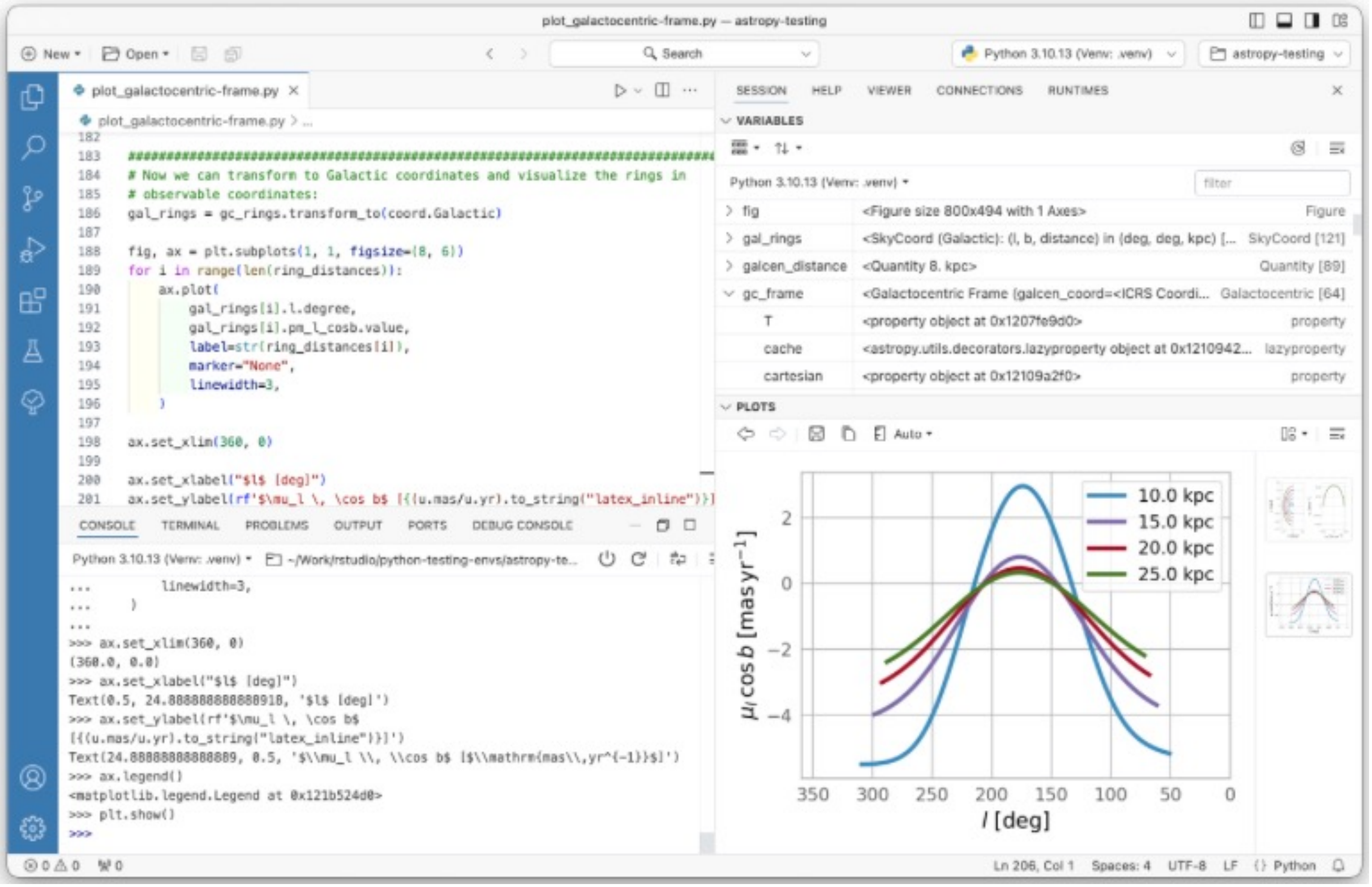
```
syntax for conditional pipe step:

`{\(x) if(<condition>) <function>(x, <function_arguments>) else x}() |>`
or
`{\(y) if(<condition>) <function>(y, <function_arguments>) else y}() |>`
or
use any variable name as the 'anonymous function variable' to carry the data frame
through to the next stage of the pipe if the <condition> is not met

An extension of this that allows for two different things to happen, pending on the
`<condition>` is
```



```
{\(x) if(<condition>)
  <function_if_condition_true>(x, <function_arguments>)
  else
    <function_if_condition_false>(x, <function_arguments>)}() |>
```

Positron

https://positron.posit.co


plot_galactocentric-frame.py — astropy-testing
New
Open
Search
Python 3.10.13 (Venv: .venv)
astropy-testing
plot_galactocentric-frame.py
SESSION
HELP
VIEWER
CONNECTIONS
RUNTIMES
VARIABLES
PLOTS
Auto
CONSOLE
TERMINAL
PROBLEMS
OUTPUT
PORTS
DEBUG CONSOLE
Python 3.10.13 (Venv: .venv)
~/Work/rstudio/python-testing-envs/astropy-te...
10.0 kpc
15.0 kpc
20.0 kpc
25.0 kpc
$\mu_l \cos b$ [mas yr^{-1}]
l [deg]
Ln 206, Col 1
Spaces: 4
UTF-8
LF
Python

```python
##########################################################################
# Now we can transform to Galactic coordinates and visualize the rings in
# observable coordinates:
gal_rings = gc_rings.transform_to(coord.Galactic)

fig, ax = plt.subplots(1, 1, figsize=(8, 6))
for i in range(len(ring_distances)):
    ax.plot(
        gal_rings[i].l.degree,
        gal_rings[i].pm_l_cosb.value,
        label=str(ring_distances[i]),
        marker="None",
        linewidth=3,
    )

ax.set_xlim(360, 0)

ax.set_xlabel("$l$ [deg]")
ax.set_ylabel(rf'$\mu_l \, \cos b$ [{(u.mas/u.yr).to_string("latex_inline")}]
```

fig	<Figure size 800x494 with 1 Axes>	Figure
gal_rings	<SkyCoord (Galactic): (l, b, distance) in (deg, deg, kpc) [...	SkyCoord [121]
galcen_distance	<Quantity 8. kpc>	Quantity [89]
gc_frame	<Galactocentric Frame (galcen_coord=<ICRS Coordi...	Galactocentric [64]
T	<property object at 0x1207fe9d0>	property
cache	<astropy.utils.decorators.lazyproperty object at 0x1210942...	lazyproperty
cartesian	<property object at 0x12109a2f0>	property

```
...         linewidth=3,
...     )
...
>>> ax.set_xlim(360, 0)
(360.0, 0.0)
>>> ax.set_xlabel("$l$ [deg]")
Text(0.5, 24.888888888888918, '$l$ [deg]')
>>> ax.set_ylabel(rf'$\mu_l \, \cos b$
[{(u.mas/u.yr).to_string("latex_inline")}]')
Text(24.88888888888889, 0.5, '$\\mu_l \\, \\cos b$ [$\\mathrm{mas\\,yr^{-1}}$]')
>>> ax.legend()
<matplotlib.legend.Legend at 0x121b524d0>
>>> plt.show()
>>>
```