Design and Implementation of Online Experiments

nodeGame.org

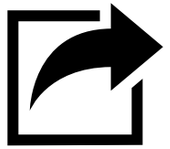
Stefano Balietti

MZES and Heidelberg

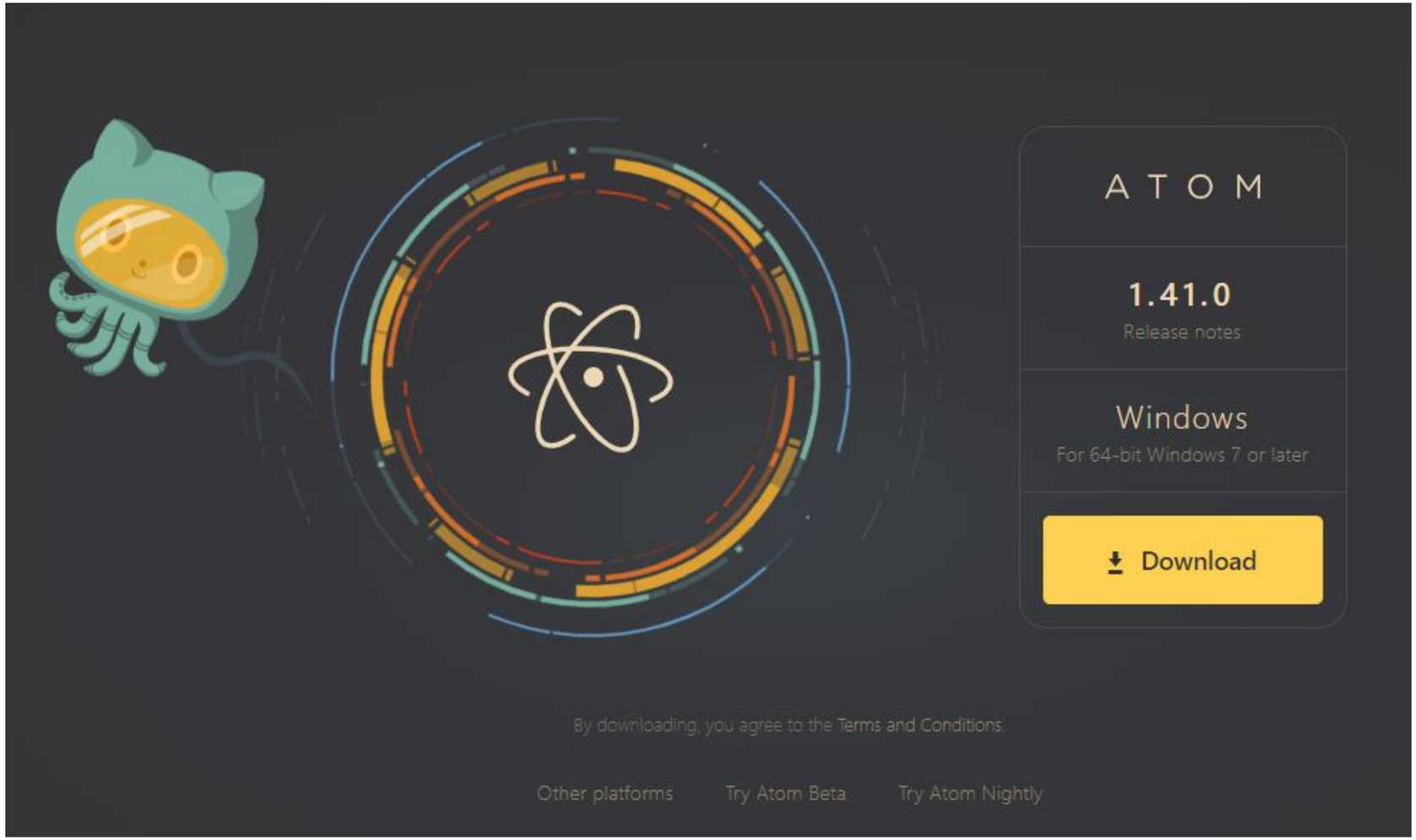
ATOM: Configuration and Getting Started Guide

@balietti
@nodegameorg
stefanobalietti.com@gmail.com

nodeGame recommends: ATOM



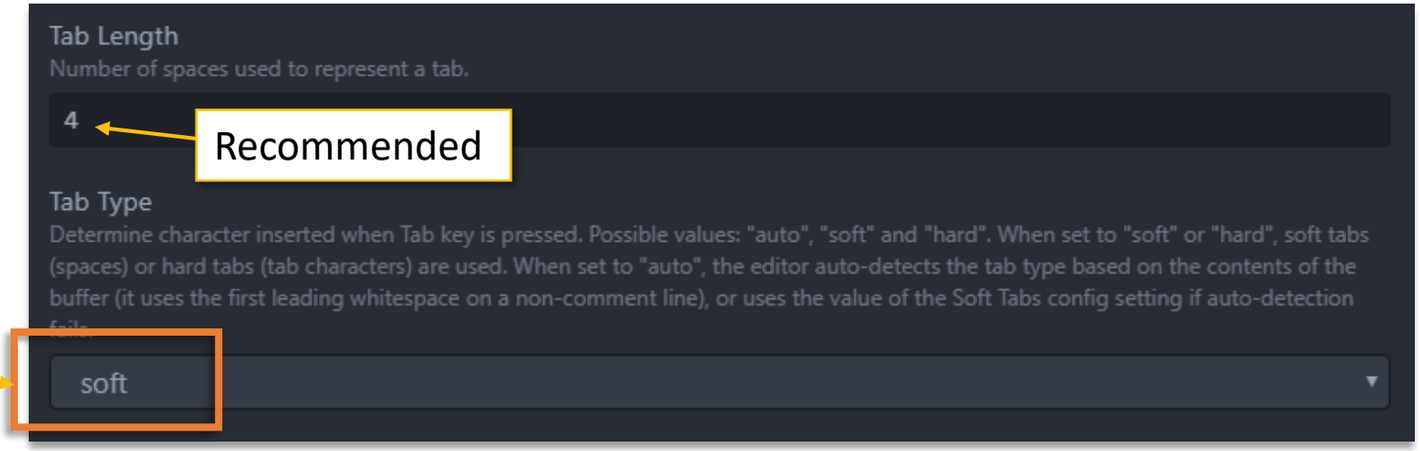
Atom.io



ATOM Editor Settings

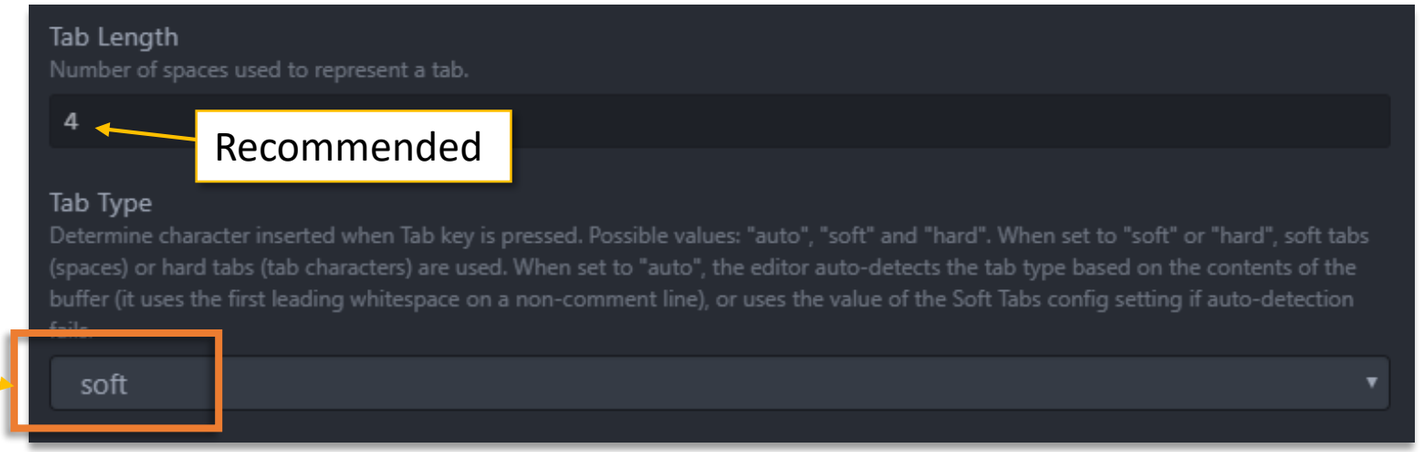
Open the Settings Panel
from the menu **File/Settings**

Make sure all your TABs are automatically converted into spaces.



ATOM Editor Settings

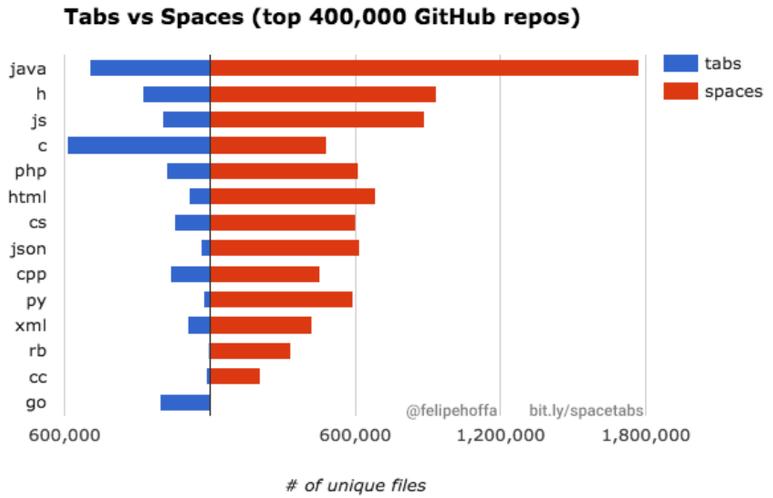
Open the Settings Panel
from the menu **File/Settings**



Make sure all your TABs are automatically converted into spaces.

Why? Because TABs are evil!

Welcome to the
Spaces vs. Tabs Debate



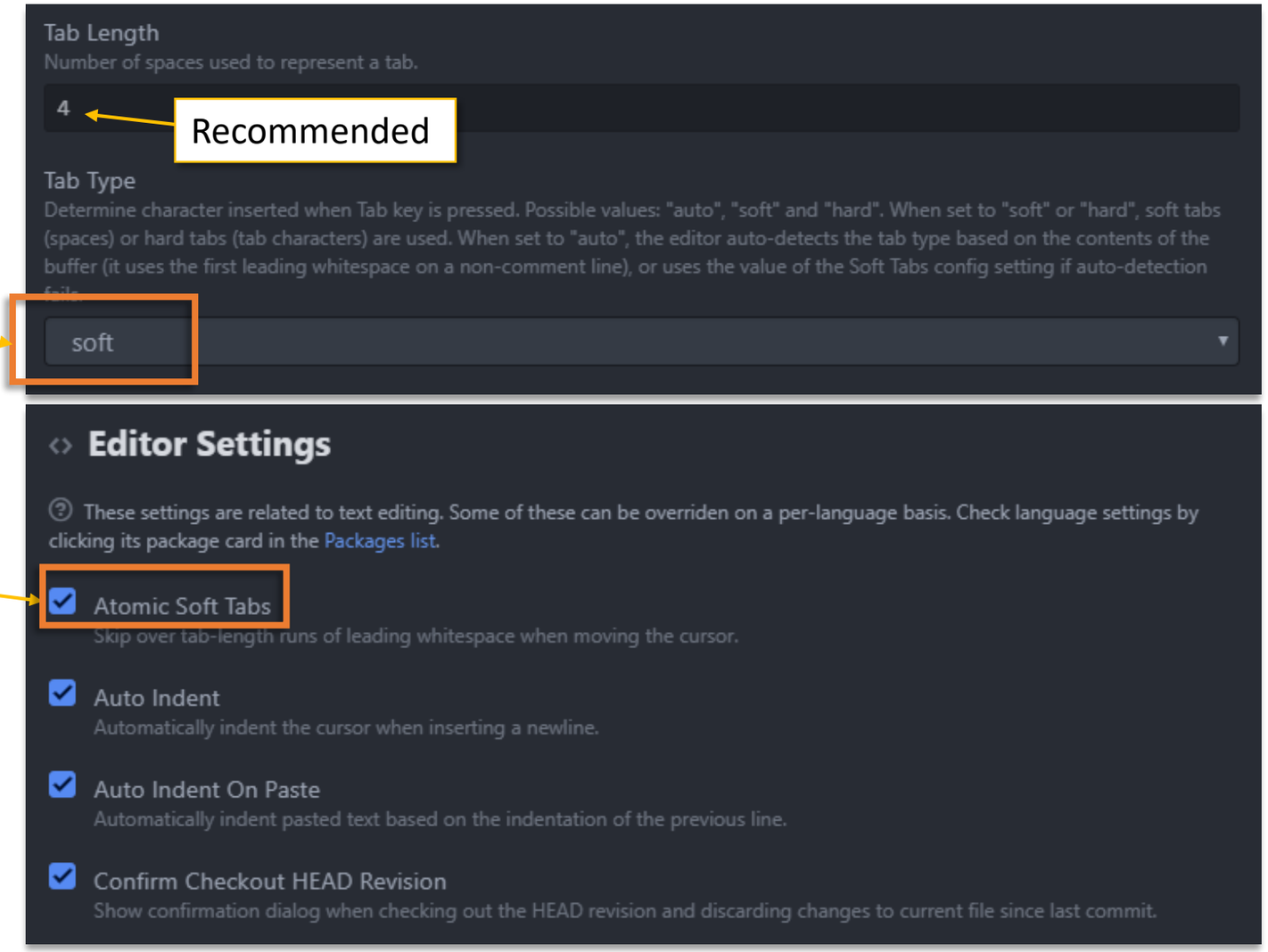
https://thenewstack.io/spaces-vs-tabs-a-20-year-debate-and-now-this-what-the-hell-is-wrong-with-go/

ATOM Editor Settings

Open the Settings Panel
from the menu **File/Settings**

Make sure all your TABs are automatically converted into spaces.

The best of the two worlds.



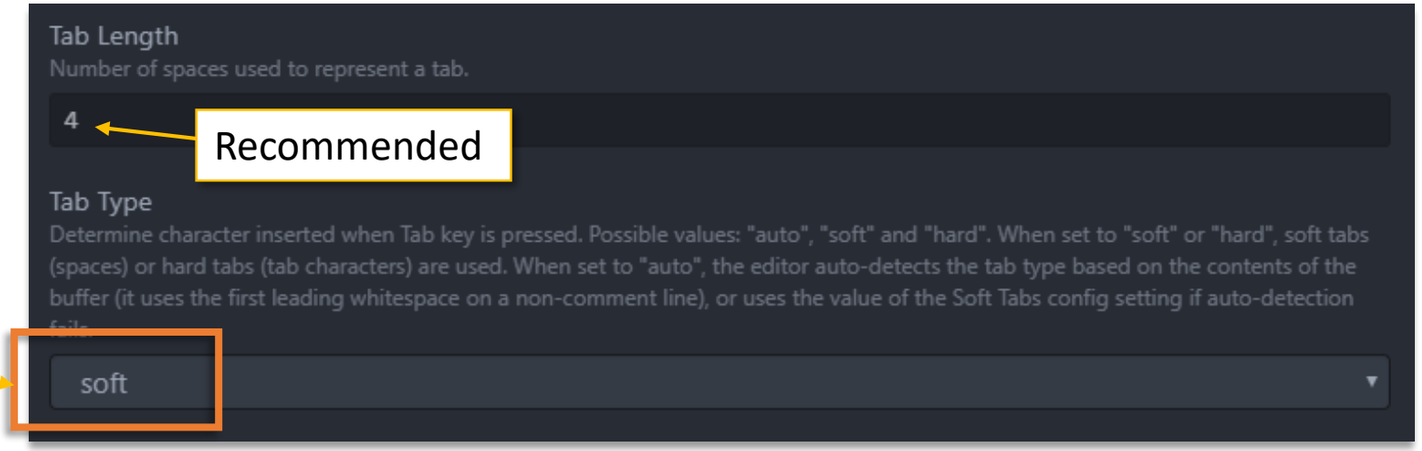
ATOM Editor Settings

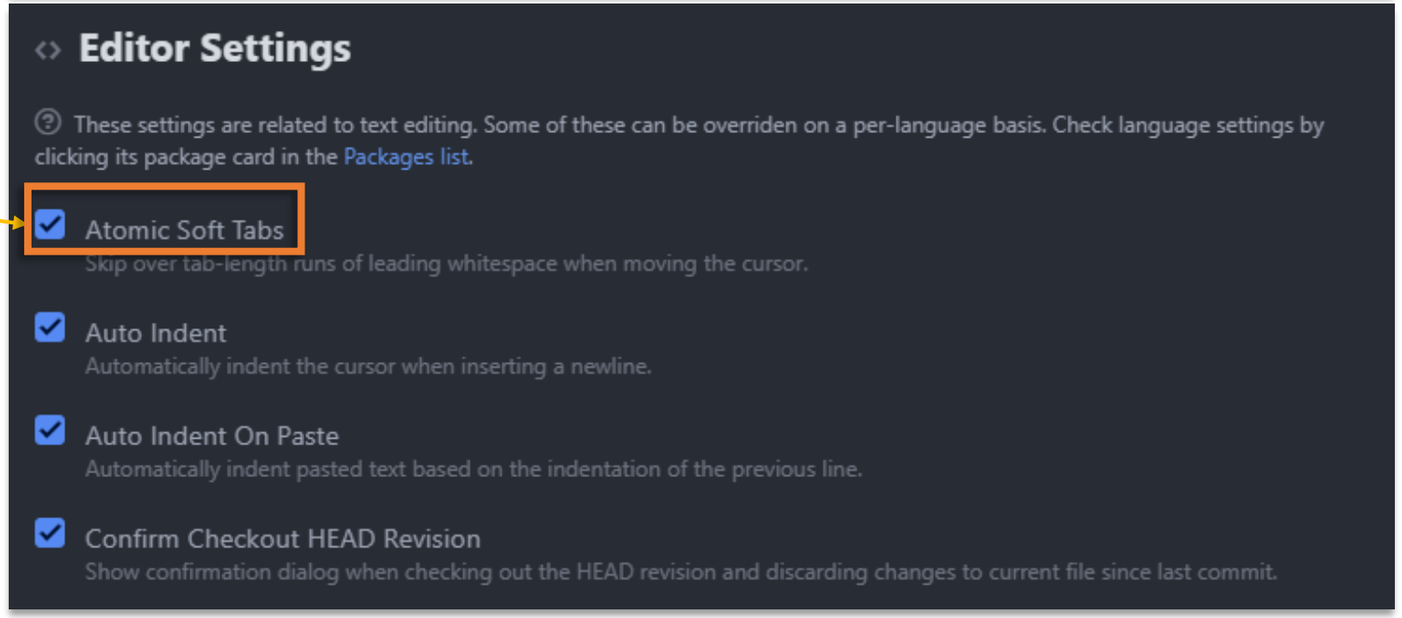
All other options on this slide strongly recommended

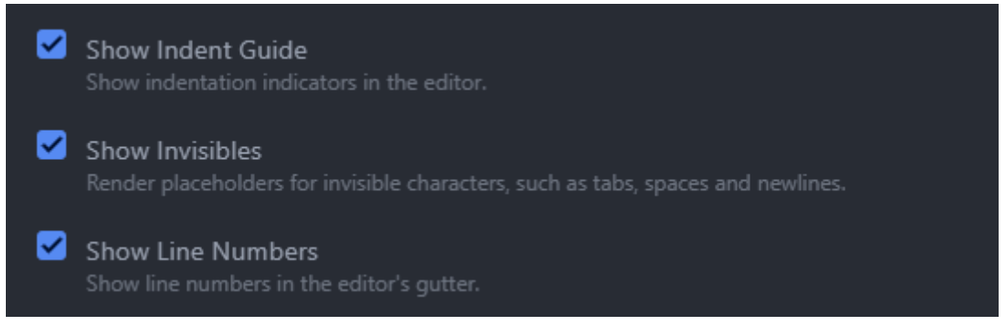
Open the Settings Panel from the menu **File/Settings**

Make sure all your TABs are automatically converted into spaces.

The best of the two worlds.

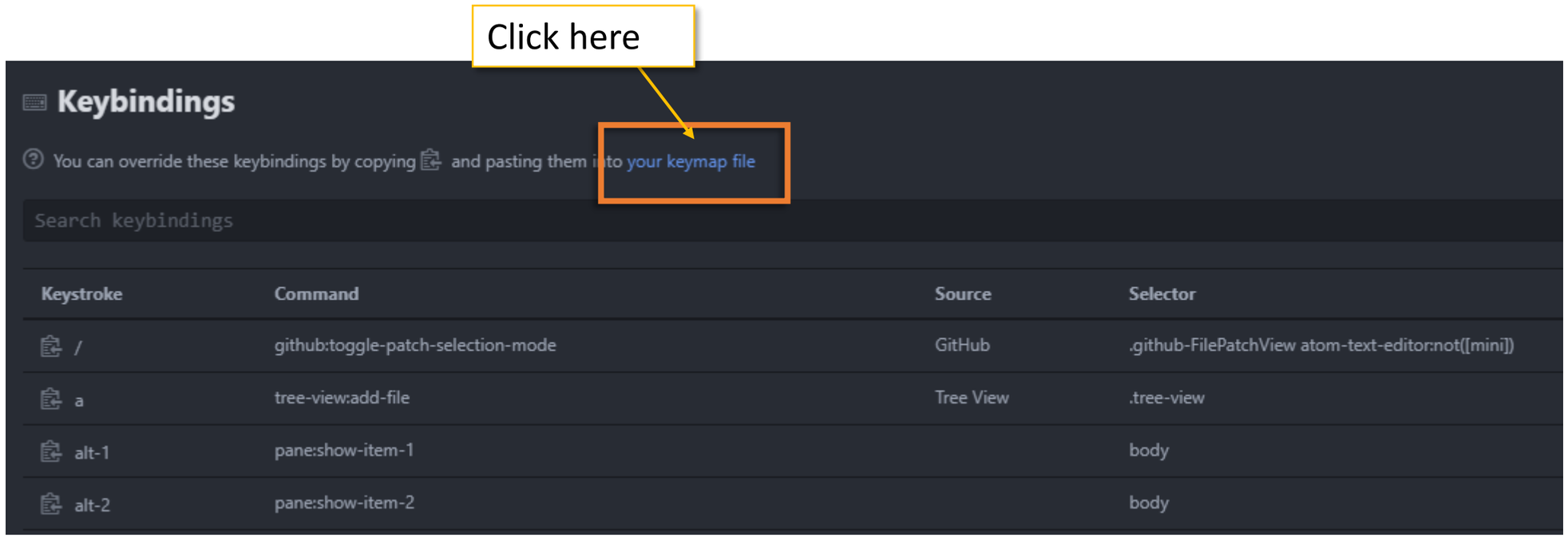






ATOM Keybindings

Keybindings are shortcuts to commands used often.
You can customize ATOM they way you like, but one shortcut is a **MUST**. Which one?



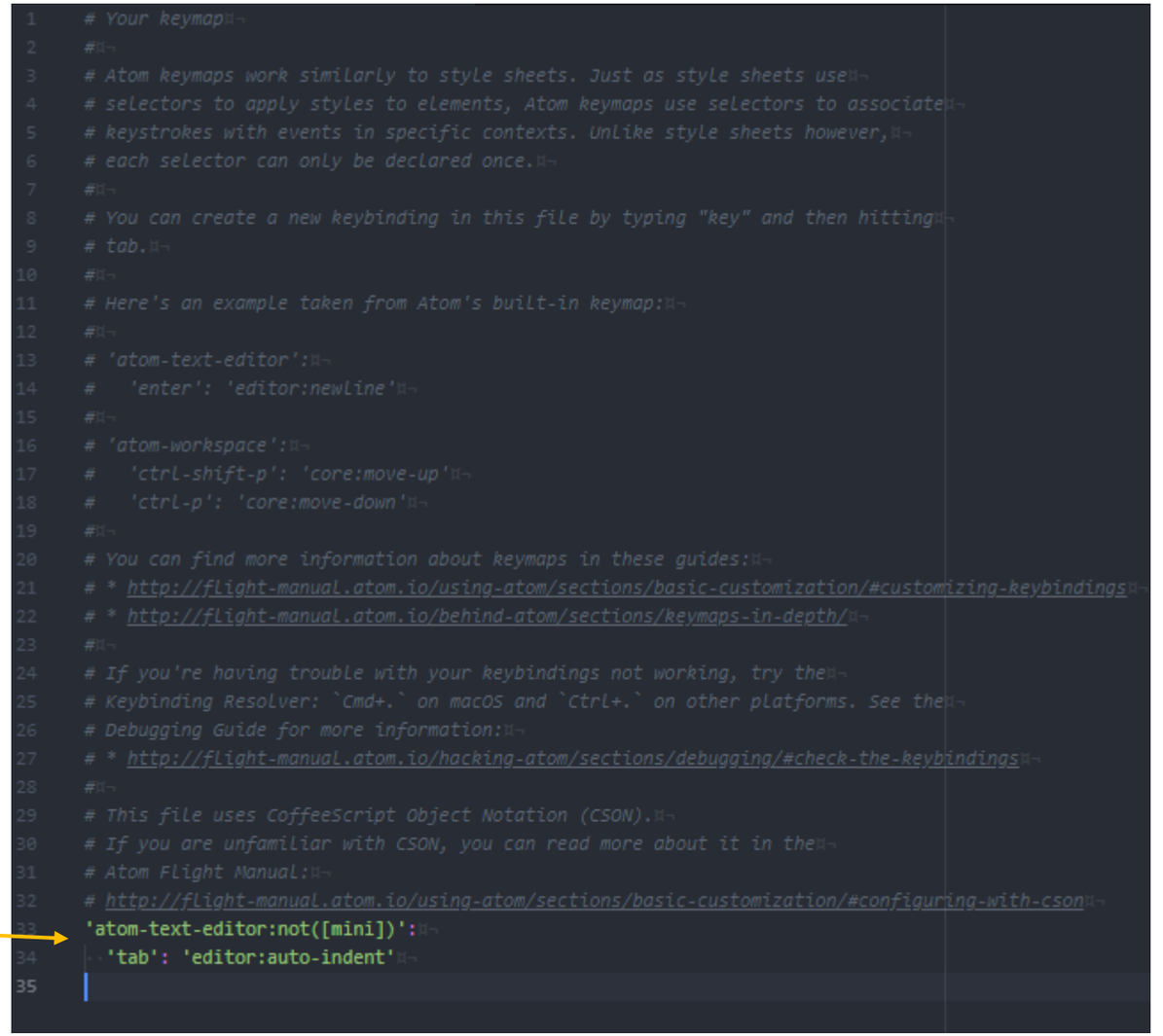
ATOM Keybindings

Set TAB to auto-indent.

Select the text you want to indent, press TAB, enjoy properly indented code.

Correct and consistent **indentation** is the key to write high-quality code.

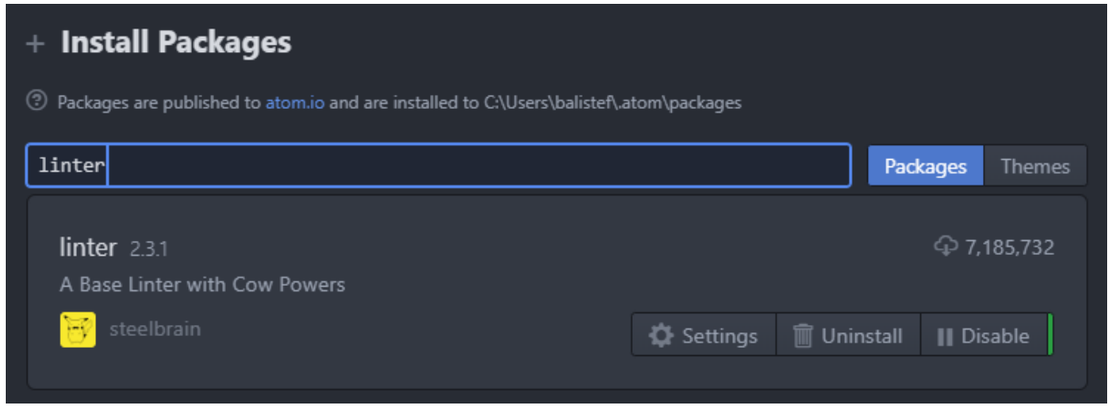
```
'atom-text-editor:not([mini])':
  'tab': 'editor:auto-indent'
```

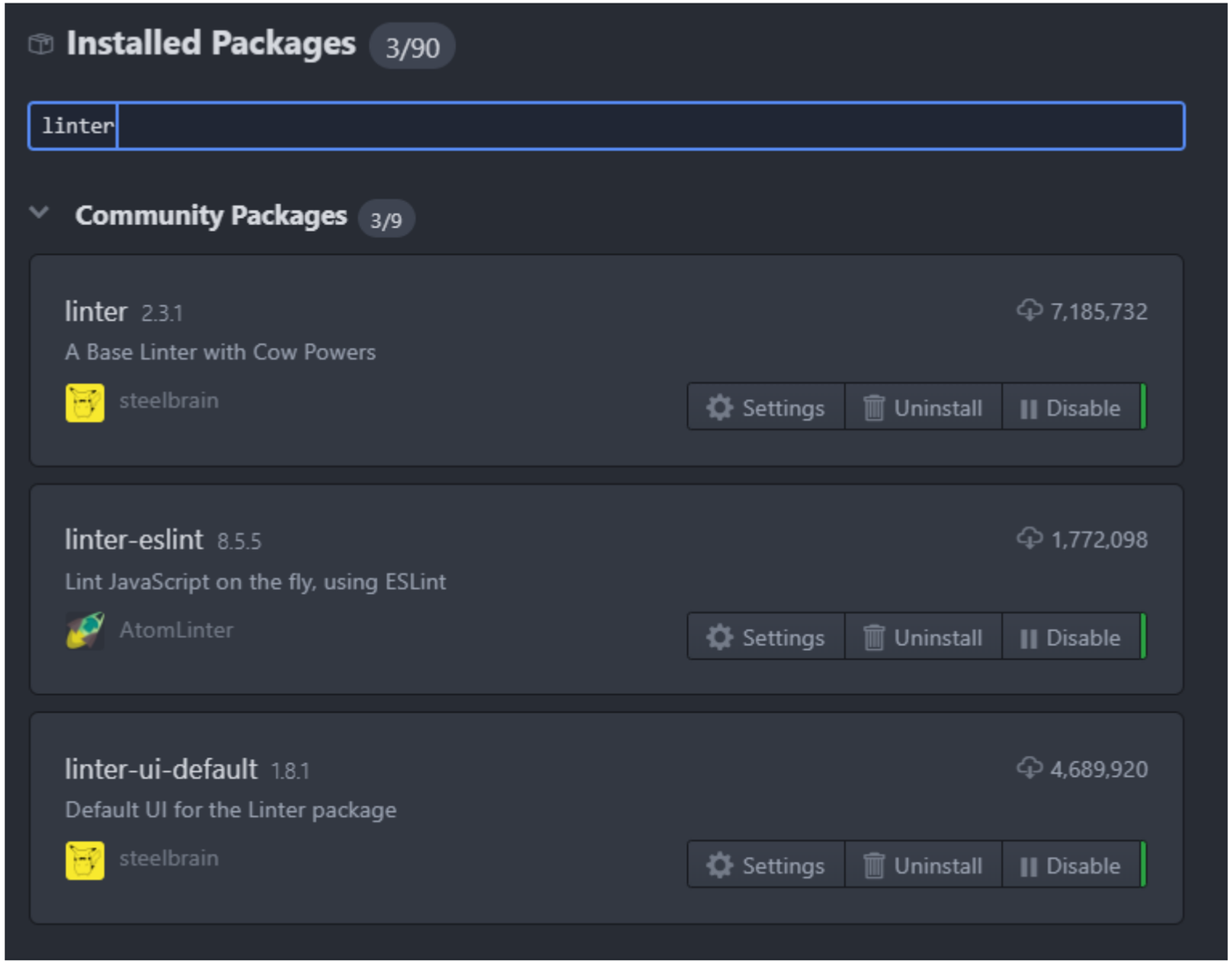


ATOM Packages

Packages extend ATOM's basic functionalities.
You can customize ATOM they way you like, but ~~one~~ 3 packages ~~is~~ are a **MUST**. Which ones?

ATOM Linters





Let's face it. Coding isn't easy… Wouldn't it be wonderful to have someone telling you spotting your mistakes for you? 🧐

You need a **Linter**! (and all 3 of those packages)

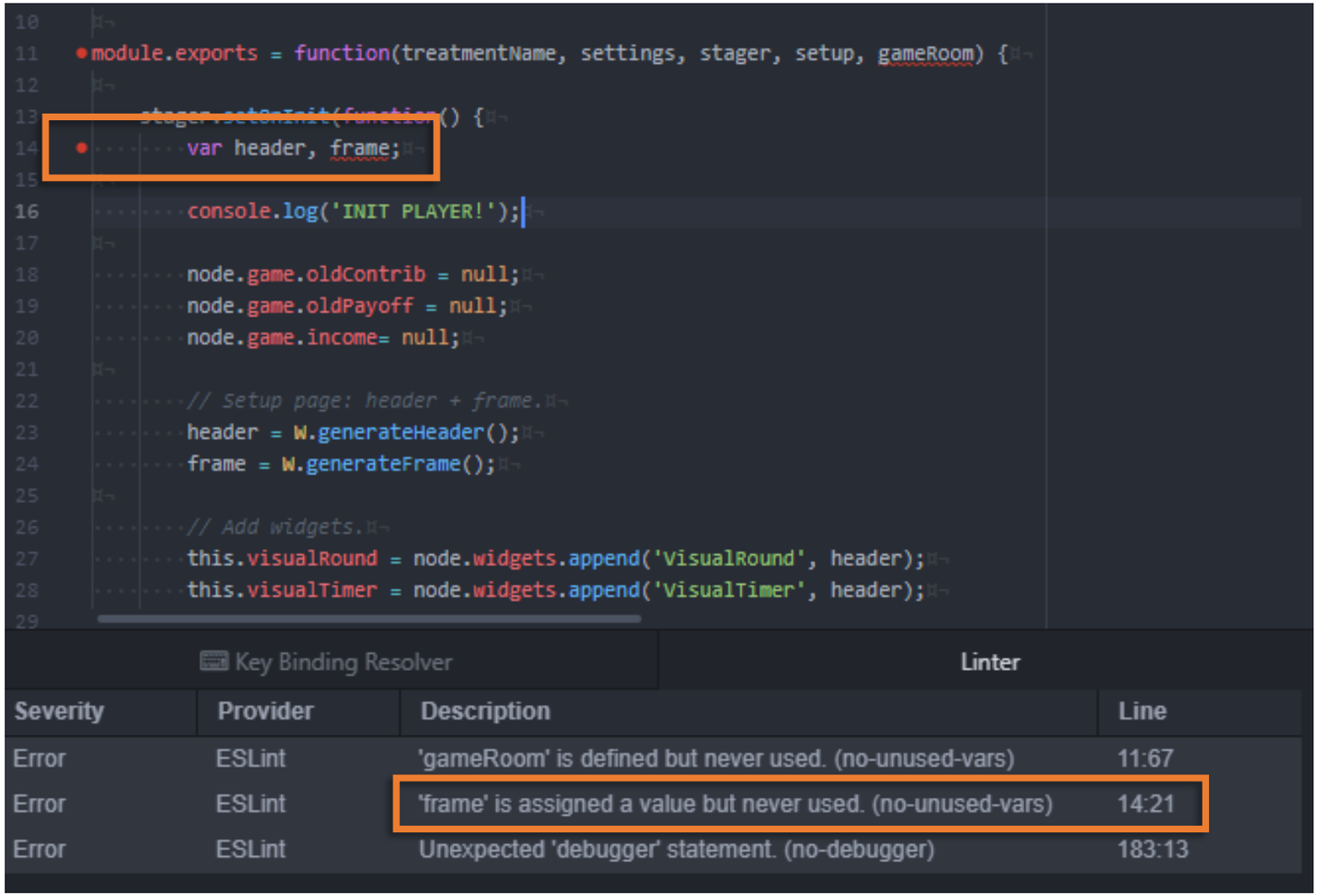
ATOM Linters

Red dot next to the line containing an error or warning.

Explanation for the error in the tooltip and in the bottom panel.

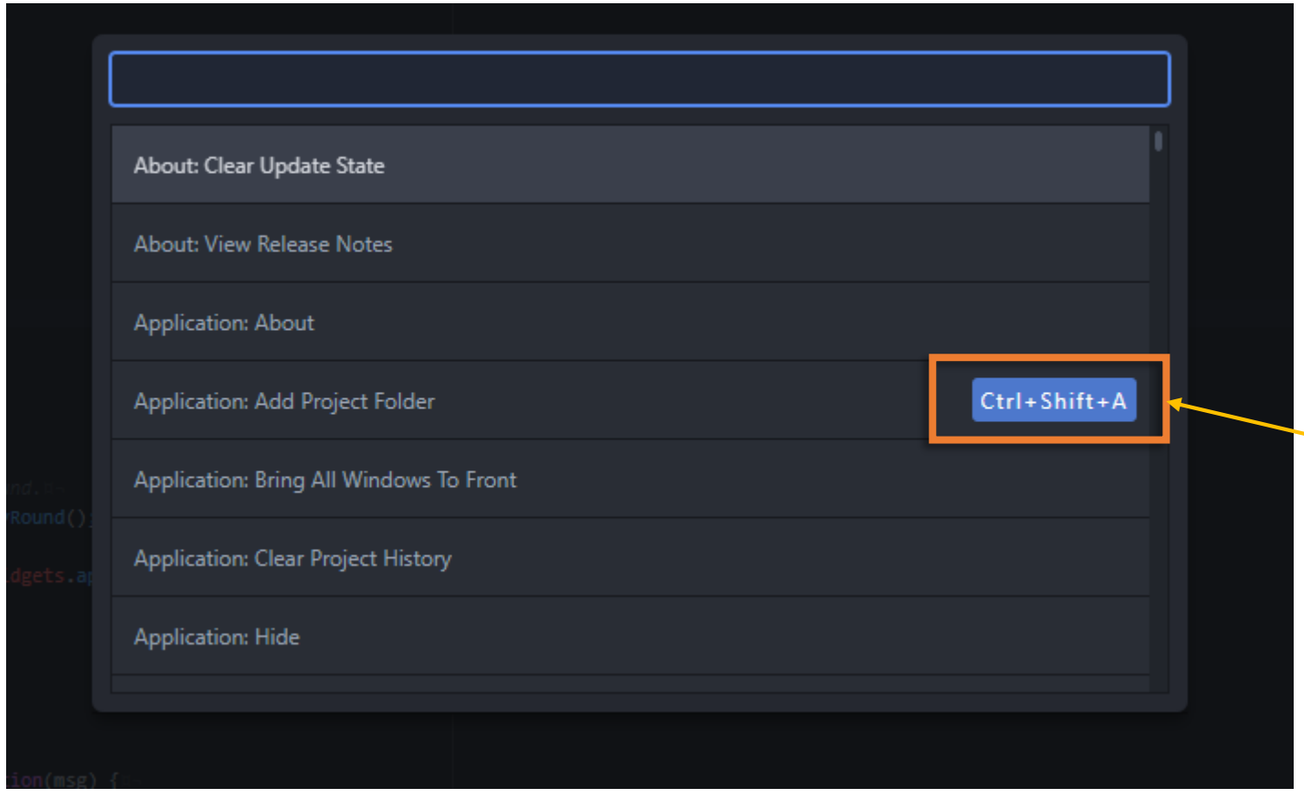
But wait…only if you added a .eslintrc.js file in your project.

Get one here: https://github.com/nodeGame/eslintrc/blob/master/.eslintrc.js



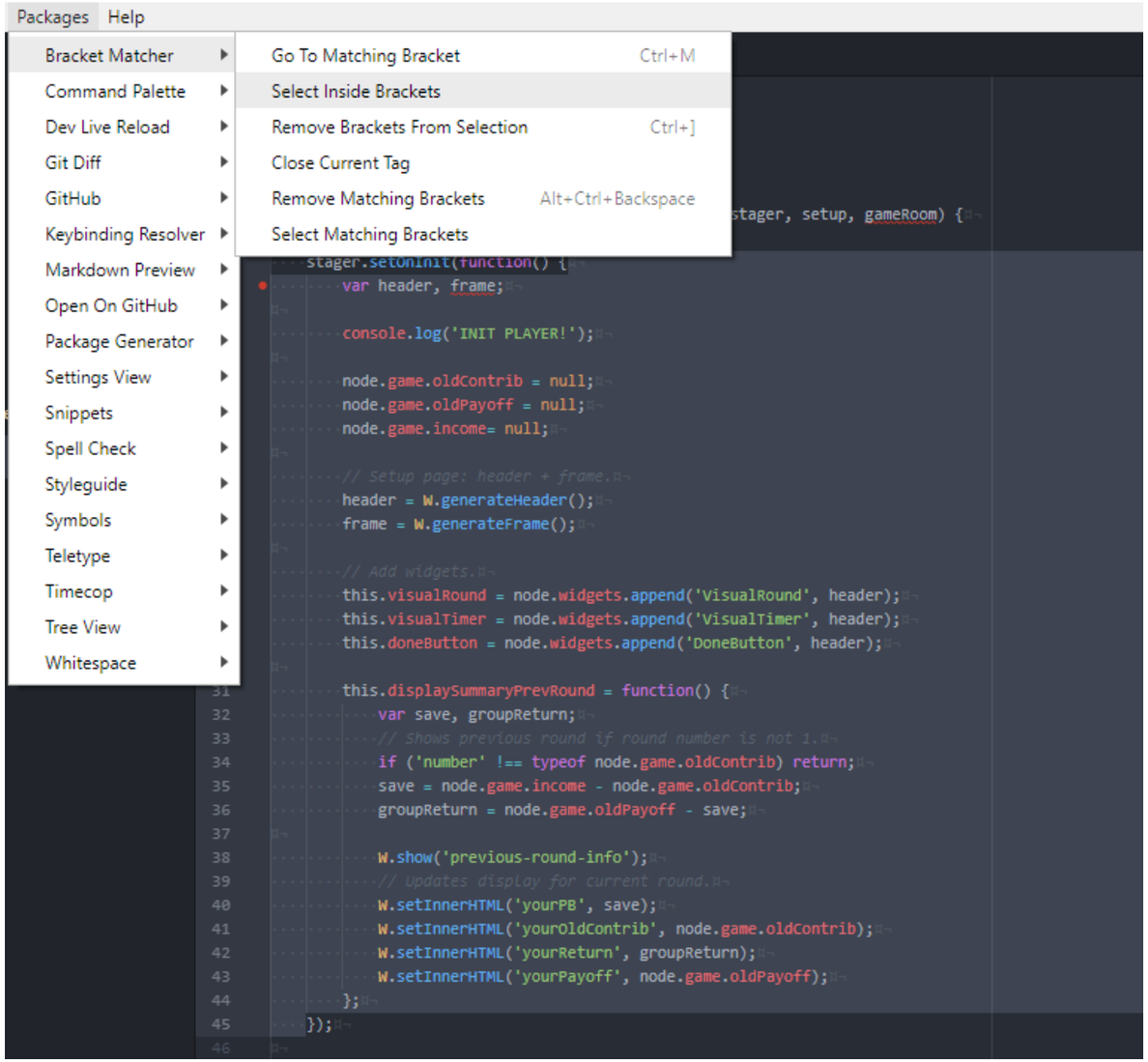
Learn New Commands and Shortcuts

Hold keys **CTRL-SHIFT-P** to open a new menu with all available commands and shortcuts (when available)



Try this shortcut to open a new project

Bracket Matcher



It is really to miss closing a parenthesis, but it is really hard to find it!

The Bracket Matcher package is here for you to help!

Hint: If you use this command a lot, what about creating a shortcut? Try it yourself following the instructions in the Keybinding slides.

Git and GitHub Integration

If you work in a team, but also if you are a lone developer, you will need:

- version controlling system,

https://git-scm.com/

- an online repository

https://github.com

You are lucky!
ATOM offers native **integration** with both!

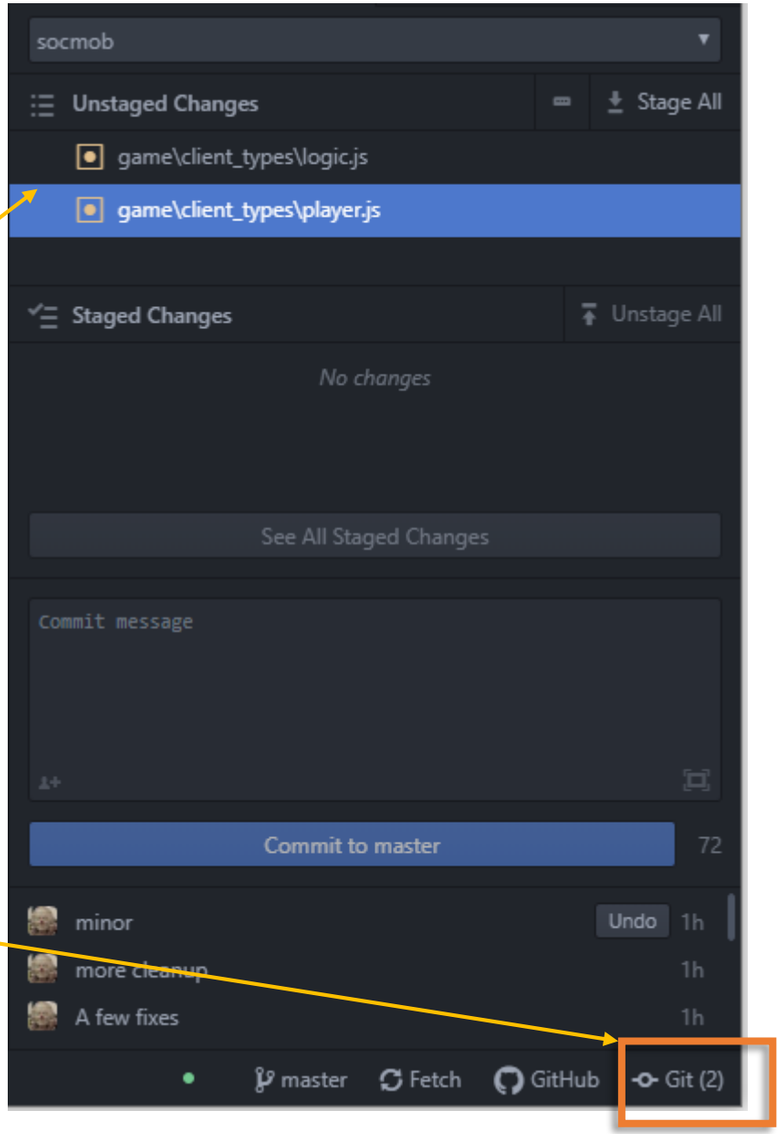
Let's learn how to use it.

Git and GitHub Integration

1. Open the Git Pane clicking on the bottom right icon.
2. Click on a file to verify the changes

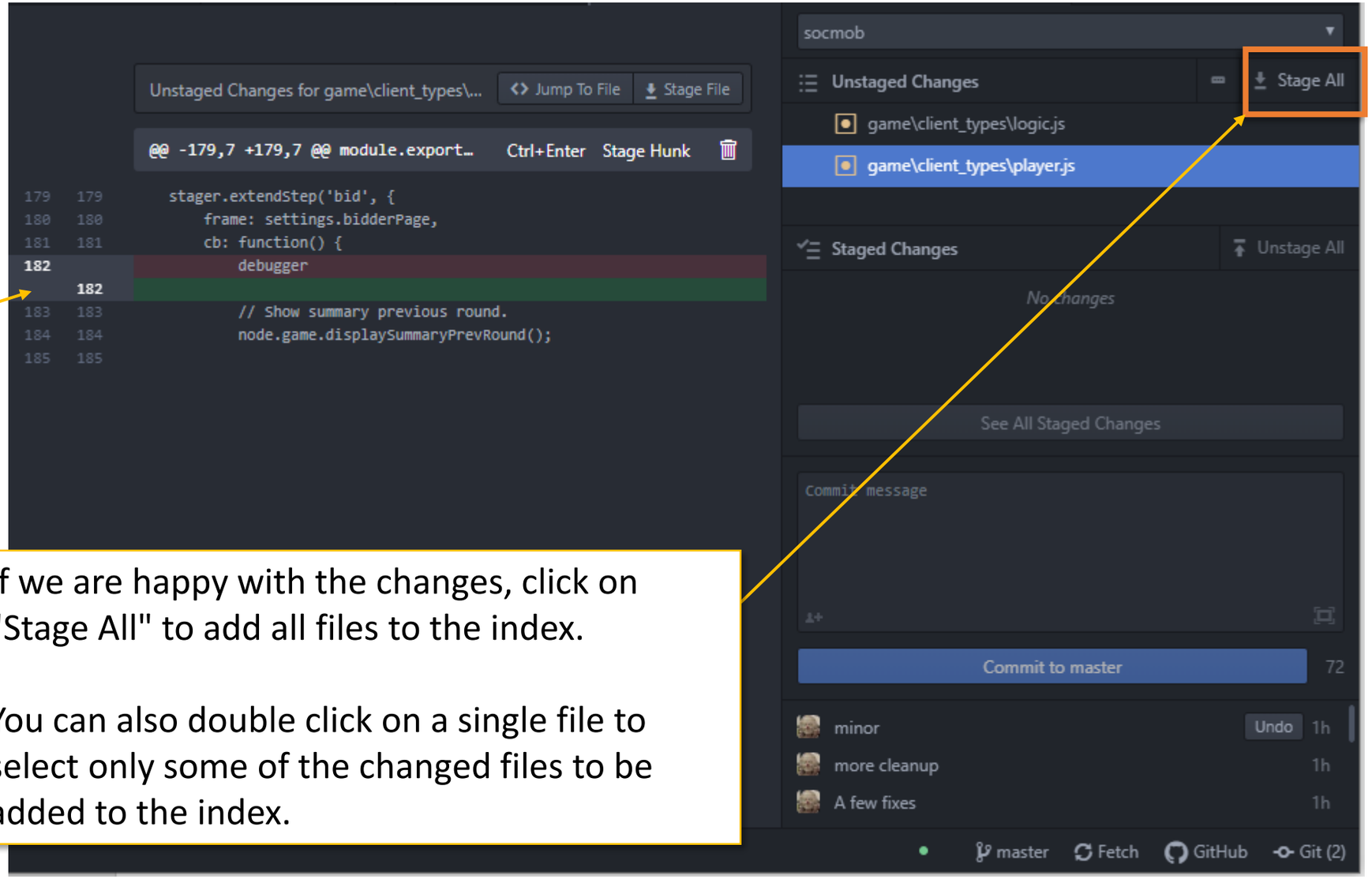
These are the files that contain changes.

The number in parenthesis counts how many files are modified ("unstaged").



Git and GitHub Integration

In file player.js we removed a line that contained a debugger statement ("red") and replaced it with an empty line ("green")



If we are happy with the changes, click on "Stage All" to add all files to the index.

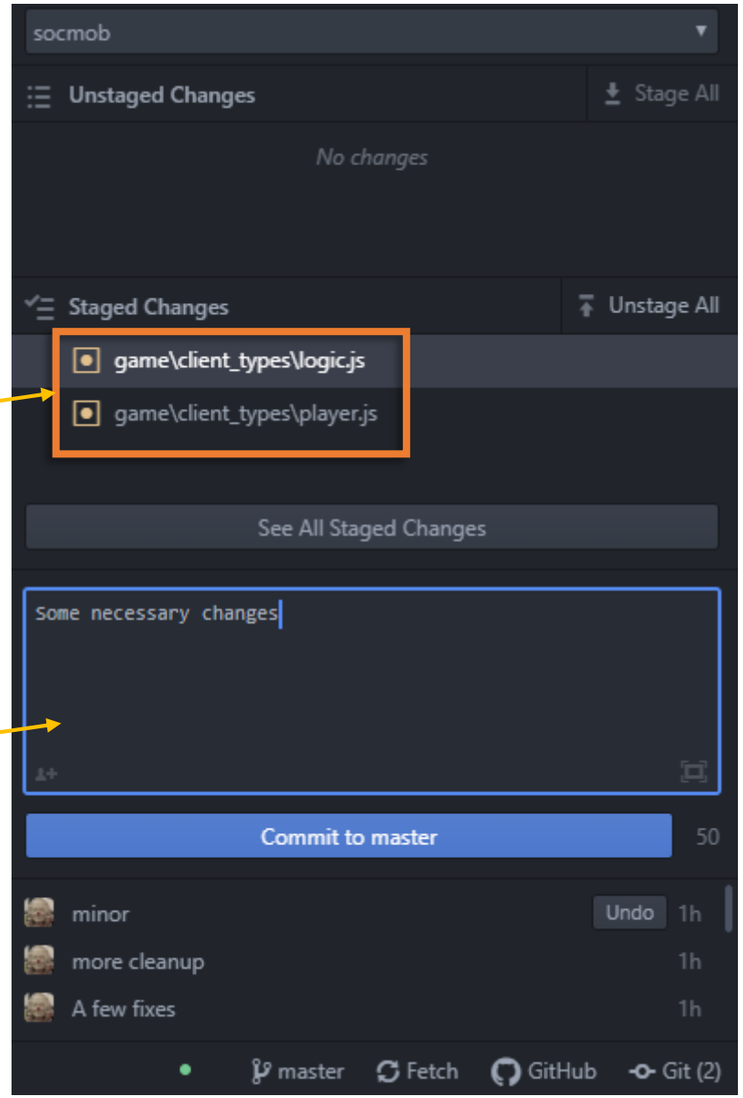
You can also double click on a single file to select only some of the changed files to be added to the index.

Git and GitHub Integration

1. Open the Git Pane clicking on the bottom right icon.
2. Click on a file to verify the changes
3. Click on Stage All
4. Add a commit message

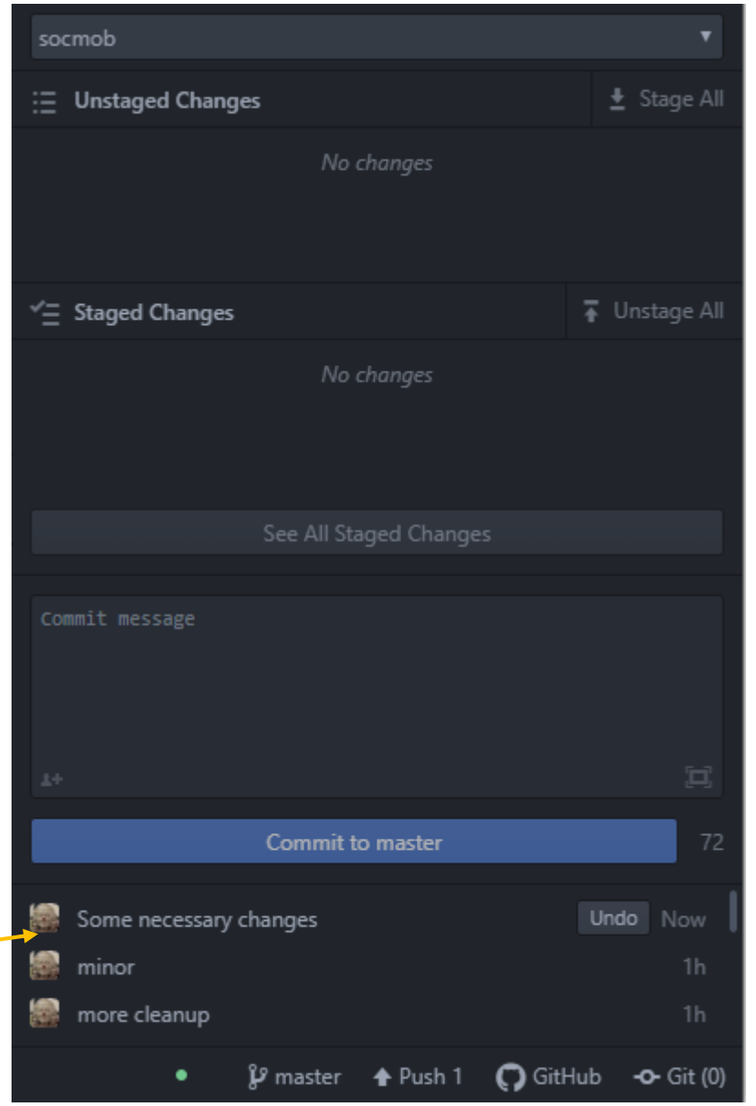
Here are the files ready to be "committed" to the index.

Here is the commit message: a short text describing what you changed.



Git and GitHub Integration

1. Open the Git Pane clicking on the bottom right icon.
2. Click on a file to verify the changes
3. Click on Stage All
4. Add a commit message
5. Click "Commit to master"



Here is your last commit has been added to your **local** index.

Git and GitHub Integration

1. Open the Git Pane clicking on the bottom right icon.
2. Click on a file to verify the changes
3. Click on Stage All
4. Add a commit message
5. Click "Commit to master"
6. Click to "Push" to upload your staged files to the online Github repository.

Note! The Github repository is available if you have a Github account and you do either:

- Follow online instructions to link an online repository to your local repository
- Clone an online repository.

See the GitHub guide for more info.