LEARNING

atom-editor

#atom-editor

Table of Contents

About

You can share this PDF with anyone you feel could benefit from it, downloaded the latest version from: atom-editor

It is an unofficial and free atom-editor ebook created for educational purposes. All the content is extracted from Stack Overflow Documentation, which is written by many hardworking individuals at Stack Overflow. It is neither affiliated with Stack Overflow nor official atom-editor.

The content is released under Creative Commons BY-SA, and the list of contributors to each chapter are provided in the credits section at the end of this book. Images may be copyright of their respective owners unless otherwise specified. All trademarks and registered trademarks are the property of their respective company owners.

Use the content presented in this book at your own risk; it is not guaranteed to be correct nor accurate, please send your feedback and corrections to info@zzzprojects.com

Chapter 1: Getting started with atom-editor

Remarks

This section provides an overview of what atom-editor is, and why a developer might want to use it.

It should also mention any large subjects within atom-editor, and link out to the related topics. Since the Documentation for atom-editor is new, you may need to create initial versions of those related topics.

Versions

Version	Release Date
1.12.8	2017-01-03
1.12.7	2016-12-08
1.12.6	2016-11-25
1.12.5	2016-11-21
1.12.4	2016-11-17
1.12.3	2016-11-15
1.12.2	2016-11-11
1.12.1	2016-11-10
1.12.0	2016-11-09
1.11.2	2016-10-17
1.11.1	2016-10-11
1.11.0	2016-10-11
1.10.2	2016-09-07
1.10.1	2016-09-06
1.10.0	2016-08-30
1.9.9	2016-08-17
1.9.8	2016-08-10

Version	Release Date
1.9.7	2016-08-08
1.9.6	2016-08-06
1.9.5	2016-08-05
1.9.4	2016-08-04
1.9.3	2016-08-04
1.9.2	2016-08-03
1.9.1	2016-08-02
1.9.0	2016-07-29
1.8.0	2016-06-06
1.7.4	2016-05-20
1.7.3	2016-04-28
1.7.2	2016-04-19
1.7.1	2016-04-14
1.7.0	2016-04-13
1.6.2	2016-04-01
1.6.1	2016-03-29
1.6.0	2016-03-17
1.5.4	2016-03-01
1.5.3	2016-02-12
1.5.2	2016-02-11
1.5.1	2016-02-10
1.5.0	2016-02-09
1.4.3	2016-02-02
1.4.2	2016-01-28
1.4.1	2016-01-25

Version	Release Date
1.4.0	2016-01-13
1.3.3	2016-01-08
1.3.2	2015-12-16
1.3.1	2015-12-10
1.3.0	2015-12-09
1.2.4	2015-11-20
1.2.3	2015-11-19
1.2.2	2015-11-18
1.2.1	2015-11-13
1.2.0	2015-11-12
1.1.0	2015-10-29
1.0.19	2015-09-25
1.0.18	2015-09-24
1.0.17	2015-09-24
1.0.16	2015-09-23
1.0.15	2015-09-18
1.0.14	2015-09-18
1.0.13	2015-09-17
1.0.12	2015-09-17
1.0.11	2015-09-02
1.0.10	2015-09-01
1.0.9	2015-08-28
1.0.8	2015-08-27
1.0.7	2015-08-12
1.0.6	2015-08-11

Version	Release Date
1.0.5	2015-08-06
1.0.4	2015-08-04
1.0.3	2015-07-28
1.0.2	2015-07-08
1.0.1	2015-07-08
1.0.0	2015-06-25
0.211.0	2015-06-22
0.210.0	2015-06-17
0.209.0	2015-06-11
0.208.0	2015-06-09
0.207.0	2015-06-05
0.206.0	2015-06-04
0.205.0	2015-06-02
0.204.0	2015-05-28
0.203.0	2015-05-28
0.202.0	2015-05-27
0.201.0	2015-05-21
0.200.0	2015-05-19
0.199.0	2015-05-14
0.198.0	2015-05-07
0.197.0	2015-05-07
0.196.0	2015-04-30
0.195.0	2015-04-30
0.194.0	2015-04-22
0.193.0	2015-04-21

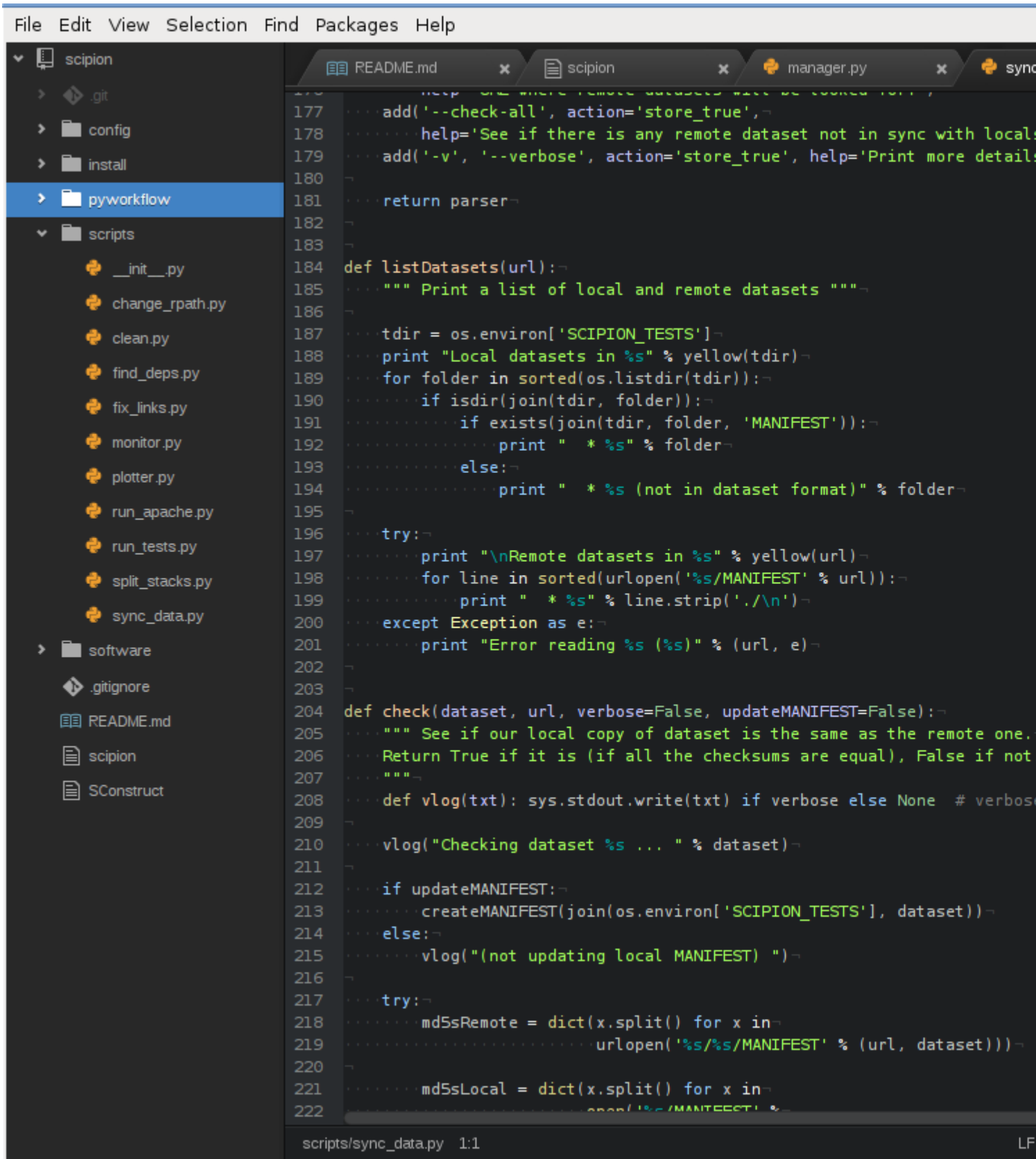| Version	Release Date
0.192.0	2015-04-15
0.191.0	2015-04-14
0.190.0	2015-04-07
0.189.0	2015-04-01
0.188.0	2015-03-24
0.187.0	2015-03-09

Examples

What is Atom?

Atom is a hackable text editor created by GitHub and developed on top of the Electron desktop application platform.

This means it can be used as a text editor for basic programming up to a full-sized IDE. It is also extremely customisable, it provides thousands of community-made packages (syntax highlighting, improved UX, etc.) and themes to suit everyone's needs. It is also available on Windows, MacOS, and Linux.

Here is an example:

Atom provides other helpful features including:

- Opening directories
- Multiple editing tabs
- Side-by-side editing panes
- Multiple editing cursors

- Line switching
- File and directory tree management

Running a "Hello, World!" program in Python using Atom from scratch

Atom is versatile and flexible text editor and has hundreds of community-made, open-source packages that can compile and run source files, for many languages. This guide will show how easy it is to code Python with the Atom editor.

This guide assumes you do not have Python nor Atom installed in your system.

Step 1: Installing Python

Python can be installed from the either the official website, or if you're using Linux, through package managers (however Python usually comes pre-installed anyways).

If you're a Windows user, do not forget to set `python.exe` to your `%PATH%`.

Step 2: Installing Atom

You can install the Atom editor from the official website or through package managers.

Step 3: Configuring Atom

For more information about installing packages, and themes, read this dedicated topic.

In order to compile and run programs, the Atom community provides packages to fill that need. For this example, we will be using `script` to run our program.

Go to File > Settings > Install.

Type `script` in the search bar and install it. When it is installed, it should be listed in "Packages" in the Settings bar. It should be noted that `script` is not capable of user input.

If you're using MacOS or Linux, you can use the `apm` package manager to install packages.

Step 4: Programming and executing

Pick a directory where you would like to store your PY source file.

1. Make sure you can see the Tree View pane; if you cannot see this pane, you can toggle it by going to View > Toggle Tree View.

2. Go to File > Add Project Folder and select a directory which will be set as your root directory for a project.

3. Right-click the folder and click New File, then enter in `hello-world.py` and type in the

following code:

```
print("Hello, World!")
```

4. Press CTRL+SHIFT+B to run the script. Alternatively, you can go to View > Toggle Command Palette and enter Script: Run.

The script should return:

```
Hello, World!
[Finished in 0.125s]
```

Read Getting started with atom-editor online: https://riptutorial.com/atom-editor/topic/8684/getting-started-with-atom-editor

---

Chapter 2: Basic Editing With Atom

Remarks

Note, the icons used at the end of the **Opening Files and Directories** example are not part of Atom's standard styling, but are the result of the file-icons styling package.

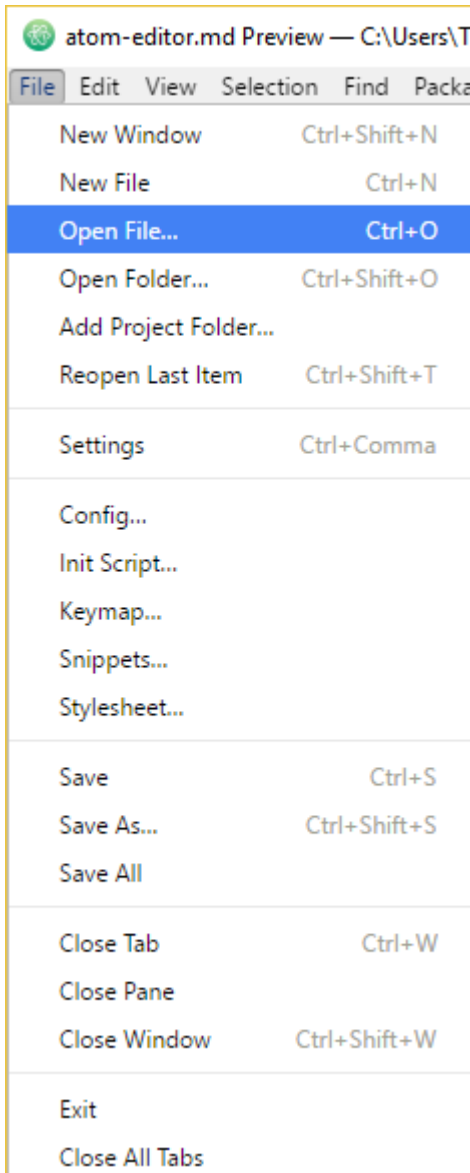## Examples

Opening Files and Directories

Along with other more advanced text editors, Atom allows developers to open a single file or a directory.
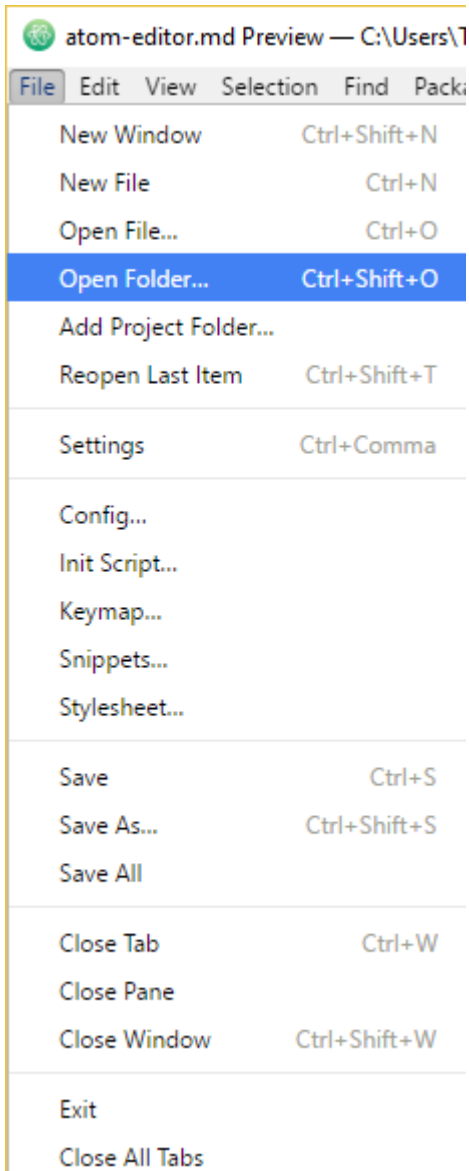
Opening Files

To open files with Atom, either use **File > Open File...** in the menu as show below:
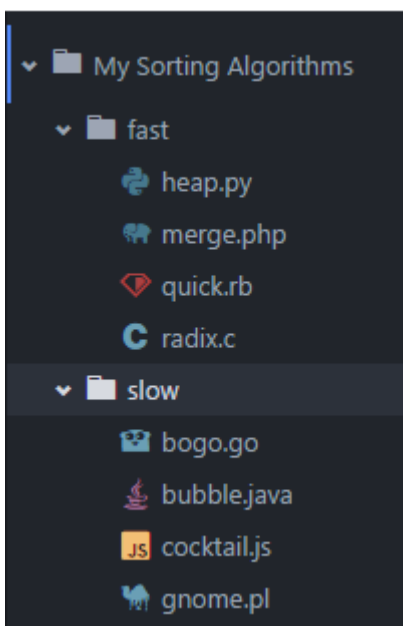
or use the faster keyboard shortcut: `Ctrl`+`O` (For Mac OS: `⌘`+`O`). This will open a file explorer (Finder for Mac OS) from which you can select a file to open, or to select multiple files, use the `Ctrl` (For Mac `⌘`) key while clicking on other files or hold the `Shift` key while selecting other files to select a range. When you have selected the files you wish to open, press the `Open` button on the file explorer. Atom, as a text editor, only elects to handle files under 2 megabytes.

Opening Directories

Especially for projects, Atom's directory opening feature can be quite useful. To do so, you may either use the option in Atom's file menu:
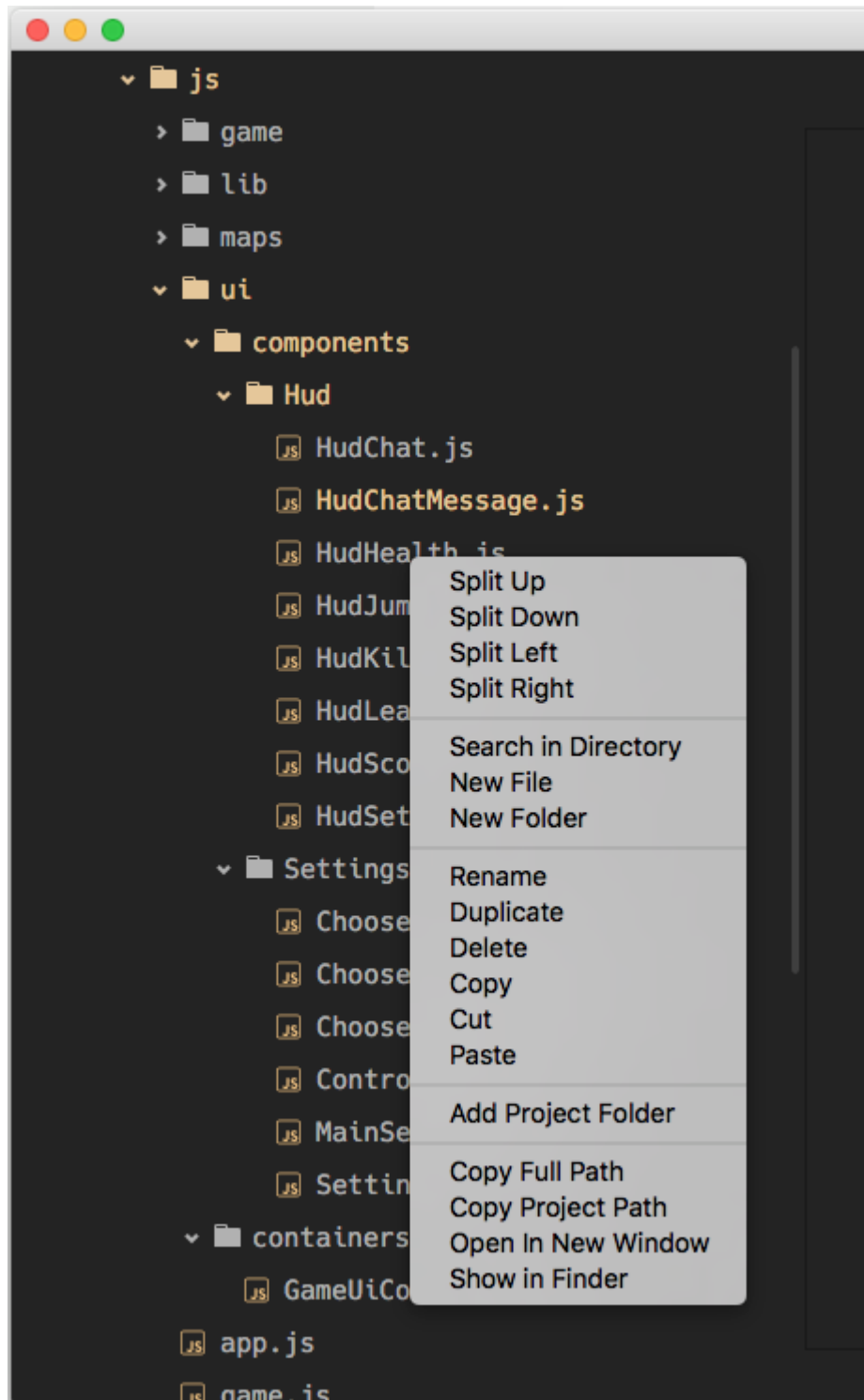
or use the keyboard shortcut `Ctrl`+`Shift`+`o` (For Mac OS: ⌘+`Shift`+`o`). Opening directories will allow you to access other directories and files below the root directory:

Interactive File Tree

In order to keep track of your projects' file structure, Atom, like many text editors and IDEs, uses a file tree model. These trees show the locations and names of your files and directory. To toggle the tree between visible and hidden, the keys `Ctrl`+`\` may be used (⌘+`\` for Mac OS). This tree also includes many operations for both files and directories as shown below:

Operation	Description
Split Up	Splits the editor into two panes with the selected file on the top

Operation	Description
Split Down	Splits the editor into two panes with the selected file on the bottom
Split Left	Splits the editor into two panes with the selected file on the left
Split Right	Splits the editor into two panes with the selected file on the
Search in Directory	Opens the find and replace tool to search the selected file or directory
New File	Creates a new file in the scope of the directory where the click occurred
New Folder	Creates a new folder in the scope of the directory where the click occurred
Rename	Changes the name of the file or directory
Duplicate	Creates an exact copy of the file or directory
Delete	Removes the file or directory
Copy	Copies the file or directory to the clipboard
Paste	Pastes a copied file or directory from the clipboard
Add Project Folder	Allows you to select another directory to be included at the root of the tree
Copy Full Path	Copies the full system path to the selected file or directory onto the clipboard
Copy Project Path	Copies the path of the selected item relative to the project root to the clipboard
Open In New Window	Opens the file or directory as the root in a new window
Show in Finder	Opens the default file explorer of the OS to the selected file or directory

Hidden files will (unless set otherwise in Atom's settings) show up with shaded filenames. A common example is GitHub's repository configuration data in the `.git` directory.

Find and Replace

The find and replace feature in Atom works in two ways, one operates locally only on the file you are in, and the other on a set of files or directories.

To open the find and replace tool for a single file, press `Ctrl`+`F` (For Mac OS use `⌘`+`F`). Enter in the

first blank the string you are searching for. Press the `Enter` key to find all instances of the string. To the right of the **Find** button are the regex, case sensitive, in selection, and whole word option buttons. The **Use Regex** button allows you to search for regex characters such as `\n`, `\t`, `\r` and regex statements `/^[a-z0-9_-]{3,16}$/`. The **Case Sensitive** button - when active - will only find strings with the same case (capitalizations). The **Only in Selection** option will only find instances of the string in highlighted sections of the file. The **Whole Word** option will only find delimited instances, not when the string is part of a larger portion. Clicking the **Replace** button will take the first instance found with the **Find** method and replace them with the contents of the replace field (even if it is empty). Clicking the **Replace All** button will replace all instances found with the **Find** method and replace them all at once with the contents of the replace field.

Read Basic Editing With Atom online: https://riptutorial.com/atom-editor/topic/8717/basic-editing-with-atom

Chapter 3: Installation and Setup

Remarks

To troubleshoot errors that occur with building from source, please view the build documents.

Examples

Installing Atom on Windows

Using the official installer

Download the installer from the official website. It will automatically add `atom` and `apm` (Atom Package Manager) to your `%PATH%` variable.

Building from source

Requirements:

- Node.js 4.4.x or later
- Python 2.7.x
- 7zip
- Visual Studio (One of the versions below)
  - Visual C++ Build Tools 2015
  - Visual Studio 2013 Update 5 (Express Edition or better)
  - Visual Studio 2015 (Community Edition or better)
- Git

Run the following commmands into Command Prompt:

```
cd C:/
git clone https://github.com/atom/atom.git
cd atom
script/build
```

Installing Atom on Mac

Installing from a zip

1. Download the `atom-mac.zip` zip file from the Atom GitHub repository here
2. Unzip the file by double clicking on it in Finder
3. Drag the `Atom` application into your "Applications" folder

---

4. Run the Atom application.

Building from Source

Requirements:

- macOS 10.8 or higher
- Node.js 4.4x or later
- npm 3.10.x or later
- Xcode

Installation:

```
git clone https://github.com/atom/atom.git
cd atom
script/build
```

After building, install with `script/build --install`

Installing Atom on Linux

Installing from a package

Debian, Ubuntu, etc.

```
$ sudo dpkg -i atom-amd64.deb
$ sudo apt-get -f install
```

RedHat Enterprise, CentOS, Oracle Linux, Scientific Linux, etc.

```
$ sudo yum install -y atom.x86_64.rpm
```

Fedora (DNF package manager)

```
$ sudo dnf install -y atom.x86_64.rpm
```

SUSE (Zypp package manager)

```
$ sudo zypper in -y atom.x86_64.rpm
```

Building from Source

Requirements:

- OS with 64 or 32 bit architecture
- C++ 11 toolchain
- Git
- Node.js 4.4x or later
- npm 3.10.x or later
- GNOME Keyring Development headers

Run the following commands:

```
git clone https://github.com/atom/atom.git
cd atom
script/build
```

For specific instructions related to a single Linux distro, read these instructions.

Read Installation and Setup online: https://riptutorial.com/atom-editor/topic/8686/installation-and-setup

Chapter 4: Themes and Packages

Introduction

Atom's packages allow you to customize the editor to your needs. This topic will explains how packages and themes are created, published, and installed.
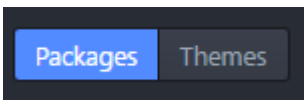
Examples

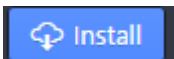### Downloading and Installing Packages and Themes

To view your installed packages or themes, open settings with `Ctrl`+, and select either the "Packages" or "Themes" tab in the left-hand navigation bar. Note, the packages or themes you installed from other publishers will show up under the "Community Themes" section and themes pre-installed with Atom will show up under the "Core Themes" section.

Packages

1. Press `Ctrl`+, to open the settings tab
2. Select the "Install" item on the left navigation pane
3. Ensure the "Packages" button is selected in the top right

4. Use the search bar at the top to find a package (ex. `file icons`)
5. Click the Install button to download and install the package

To view information on packages and their settings, click the package name.

Browse Atom packages online here.

Themes

Downloading and installing themes follows a similar process to that of packages.

1. Press `Ctrl`+, to open the settings tab
2. Select the "Install" item on the left navigation pane
3. Ensure the "Themes" option is selected by the search bar.
4. Search for a theme (ex. `atom-sublime-monokai-syntax`)
5. Click the install button to download and install

To view information on themes and their settings, click the theme name.

Browse Atom themes online here.

Use Atom Package Manager

apm is Atom's native package manager. It allows the user to manage packages and themes without having to initialise Atom itself. apm comes with the official installation and is automatically added to %PATH% if you're on Windows.

To use apm, go to Command Prompt and type

```
$ apm <command>
```

Here is the list of what you can do with this package manager.

```
clean, config, dedupe, deinstall, delete, dev, develop, disable, docs,
enable, erase, featured, home, i, init, install, link, linked, links, list,
ln, lns, login, ls, open, outdated, publish, rebuild, rebuild-module-cache,
remove, rm, search, show, star, starred, stars, test, uninstall, unlink,
unpublish, unstar, update, upgrade, view.
```

For example, if you want to do upgrade all packages from atom:

```
apm upgrade --confirm false
```

Or if you want to install a specific package:

```
apm install <package_name>
```

Read Themes and Packages online: https://riptutorial.com/atom-editor/topic/8687/themes-and-packages

Credits

S. No	Chapters	Contributors
1	Getting started with atom-editor	Community, Pradana Aumars, Thomas Gerot
2	Basic Editing With Atom	Thomas Gerot
3	Installation and Setup	Matthias Pfefferle, Pradana Aumars, Thomas Gerot
4	Themes and Packages	Pradana Aumars, rscata, Thomas Gerot