

# The Unix Shell

Managing your code: quietly introducing *Git* - a friend for life

Thanks to all contributors:

Alison Pamment, Sam Pepler, Ag Stephens, Stephen Pascoe, Kevin Marsh, Anabelle Guillory, Graham Parton, Esther Conway, Eduardo Damasio Da Costa, Wendy Garland, Alan Iwi, Matt Pritchard and Tommy Godfrey.

# Managing code in the olden days

- Create "*working\_dir*"...add some code
- Write some outputs...change the code
- Publish a paper...change the code
- Copy "*working\_dir*" to "*working\_dir2*"
- Change the code
- Copy a version to a CD

*...now which version is current? Is it "*working\_dir*" or "*working\_dir2*"? And which one relates to that paper?*

# But those days are gone!

- Scientists are typically **required to publish data and code** (by their funders/institutions).
- Collaboration between scientists requires data-sharing; this implicitly relies on **code-sharing**.
- There are **tools that make it easy** to record our changes, document our workflow and "fix" releases of our code at important steps along the way.

So, working on the premise that we accept that we need to know about, and use, version control...



We will use Git and GitHub

# Introducing GitHub

<https://github.com>



The screenshot shows the GitHub homepage with a navigation bar at the top containing links for Personal, Open source, Business, Explore, Pricing, Blog, and Support. A search bar and 'Sign in'/'Sign up' buttons are also present. The main content area features the headline 'How people build software' and a sub-headline 'Millions of developers use GitHub to build personal projects, support their businesses, and work together on open source technologies.' To the left of this text is a small illustration of the GitHub Octocat mascot. On the right, there is a sign-up form with three input fields: 'Pick a username', 'Your email address', and 'Create a password'. Below the password field is a note: 'Use at least one letter, one numeral, and seven characters.' A large green 'Sign up for GitHub' button is positioned below the form. At the bottom of the form, a disclaimer states: 'By clicking "Sign up for GitHub", you agree to our terms of service and privacy policy. We'll occasionally send you account related emails.'

Personal Open source Business Explore Pricing Blog Support Search GitHub Sign in Sign up

## How people build software

Millions of developers use GitHub to build personal projects, support their businesses, and work together on open source technologies.

Pick a username

Your email address

Create a password

Use at least one letter, one numeral, and seven characters.

**Sign up for GitHub**

By clicking "Sign up for GitHub", you agree to our [terms of service](#) and [privacy policy](#). We'll occasionally send you account related emails.



# Let's get started with GitHub

- Anyone can get a free GitHub account - you'll only need to pay if you want *private* repos
- We are going to learn Git and GitHub by using them throughout this course.
- Let's get started... 😊

# Create a GitHub account

Go to: <https://github.com> and sign up:



The screenshot shows the GitHub homepage with a sign-up form. The navigation bar at the top includes links for Personal, Open source, Business, and Explore, along with Pricing, Blog, and Support. A search bar and Sign in/Sign up buttons are also present. The main content area features the text 'How people build software' and a description of GitHub's use by developers. The sign-up form includes fields for 'Pick a username', 'Your email address', and 'Create a password', followed by a 'Sign up for GitHub' button. A small GitHub mascot figure is visible in the bottom left corner of the main content area.

Personal Open source Business Explore Pricing Blog Support Search GitHub Sign in Sign up

## How people build software

Millions of developers use GitHub to build personal projects, support their businesses, and work together on open source technologies.

Pick a username

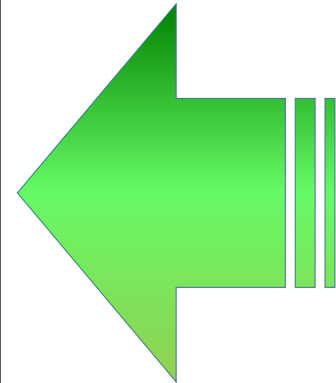
Your email address

Create a password

Use at least one letter, one numeral, and seven characters.

**Sign up for GitHub**

By clicking "Sign up for GitHub", you agree to our [terms of service](#) and [privacy policy](#). We'll occasionally send you account related emails.

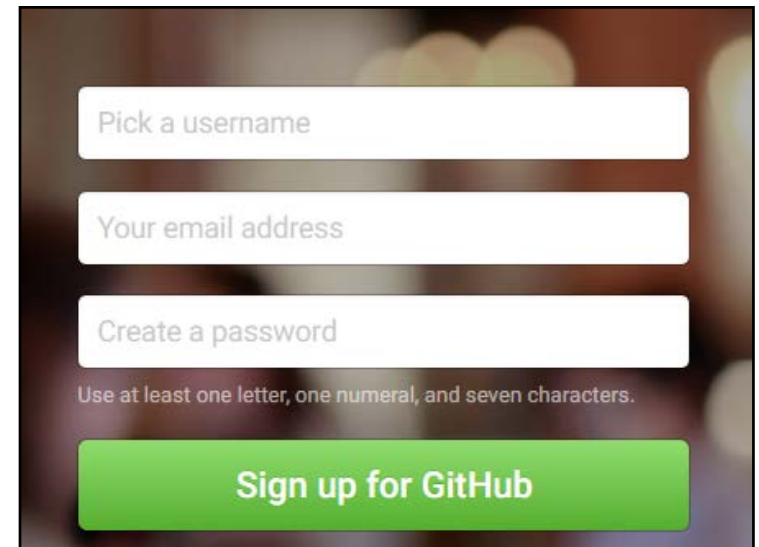




# Authentication

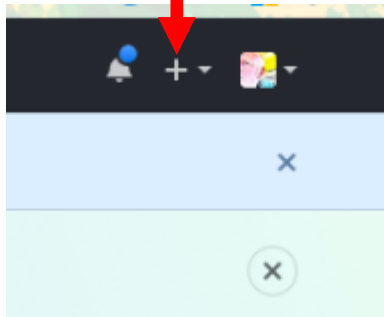
You can use either a username/password **or** SSH key authentication. The latter is more secure but many folks use username/password.

For this course **we will use username/password** for simplicity.

A screenshot of the GitHub sign-up form. It features three white input fields on a dark background: 'Pick a username', 'Your email address', and 'Create a password'. Below the password field is a note: 'Use at least one letter, one numeral, and seven characters.' At the bottom is a large green button with the text 'Sign up for GitHub' in white.

# Make a repo

1) Click to  
add a new  
repo



3) Click the  
add  
README  
box

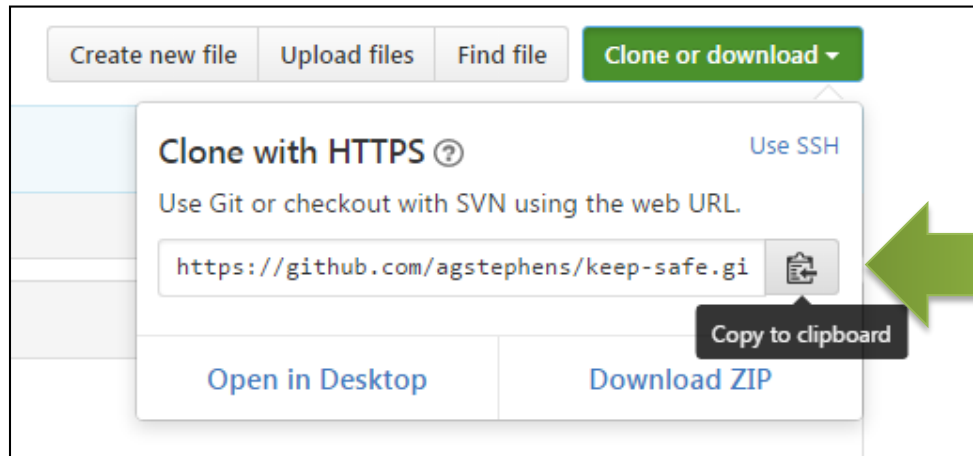
A screenshot of the GitHub 'Create a new repository' page. The page has a dark header with the GitHub logo, a search bar, and links for 'Pull requests', 'Issues', 'Marketplace', and 'Explore'. The main content area is titled 'Create a new repository' with a subtitle 'A repository contains all the files for your project, including the revision history.' Below this, there are two input fields: 'Owner' (set to 'spepler') and 'Repository name' (set to 'my-isc-work' with a green checkmark). A note says 'Great repository names are short and memorable. Need inspiration? How about vigilant-happiness?'. There is a 'Description (optional)' text area. Below these are two radio buttons for 'Public' (selected) and 'Private'. At the bottom, there is a checked checkbox for 'Initialize this repository with a README' with a sub-note. Below the checkbox are two dropdown menus for 'Add .gitignore: None' and 'Add a license: None'. A green 'Create repository' button is at the bottom.

2) Call new repo  
my-isc-work

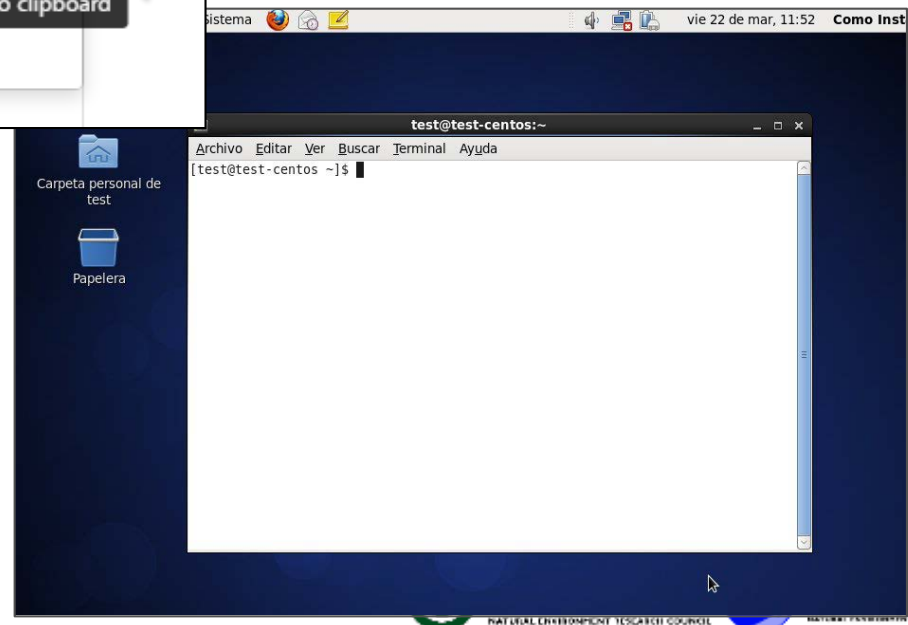


# Copy the *clone* link

1. Click "Clone or download " and copy the link.



2. Go to your Linux Terminal.



# Copy the *clone* link

3. Make sure you are in your home directory:
4. Write the git clone command and add the URL to the repository (which is different for each user):

```
$ cd  
$ git clone  
https://<username>@github.com/<username>/my-isc-work
```

The my-isc-work repo is now on the laptop and I can list the README.md file

```
$ cd my-isc-work/  
$ ls
```

```
README.md
```

Make a blank file "x"

```
$ touch x  
$ ls
```

```
README.md x
```

Use git add and git commit to put the file under version control

```
$ git add x
```

```
$ git commit x -m 'new x file'
```

```
[master 3aefe17] new x file
```

```
1 file changed, 0 insertions(+), 0 deletions(-)
```

```
create mode 100644 x
```

```
$ git push
```

Use git push to update github copy of the repo with the changes (in this case adding the "x" file)

```
Counting objects: 3, done.
```

```
Delta compression using up to 4 threads.
```

```
Compressing objects: 100% (2/2), done.
```

```
Writing objects: 100% (3/3), 272 bytes | 272.00 KiB/s, done.
```

```
Total 3 (delta 0), reused 0 (delta 0)
```

```
To https://github.com/spepler/my-isc-work.git
```

```
183fa53..3aefe17 master -> master
```

# Now it's visible on GitHub

2 commits

1 branch

0 releases

1 contributor

Branch: master ▾


New pull request

Create new file



Upload files

Find file

Clone or download ▾

 spepler new x file

Latest commit 3aefe17 16 minutes ago

 README.md	Initial commit	19 minutes ago
 x	new x file	16 minutes ago

# I'll be back...

- There is already a copy of the course materials on your laptop from a public GitHub repository.
- You are setup on GitHub for use latter in the course.
- We'll add more git stuff as we go.

