04. Installing Linux, Syntax, Comments, Style and Formatting

CPSC 120: Introduction to Programming
Pratishtha Soni~ CSU Fullerton

Agenda

0. Sign-in sheet
1. Technical Q&A
2. Installing Linux
3. Syntax and Comments
4. Code Style
5. Formatting and Diff Output

1. Technical Q&A

Technical Q&A

Let’s hear your noted questions about...

- This week’s Lab
- Linux
- Any other technical issues

Reminder: write these questions in your notebook during lab

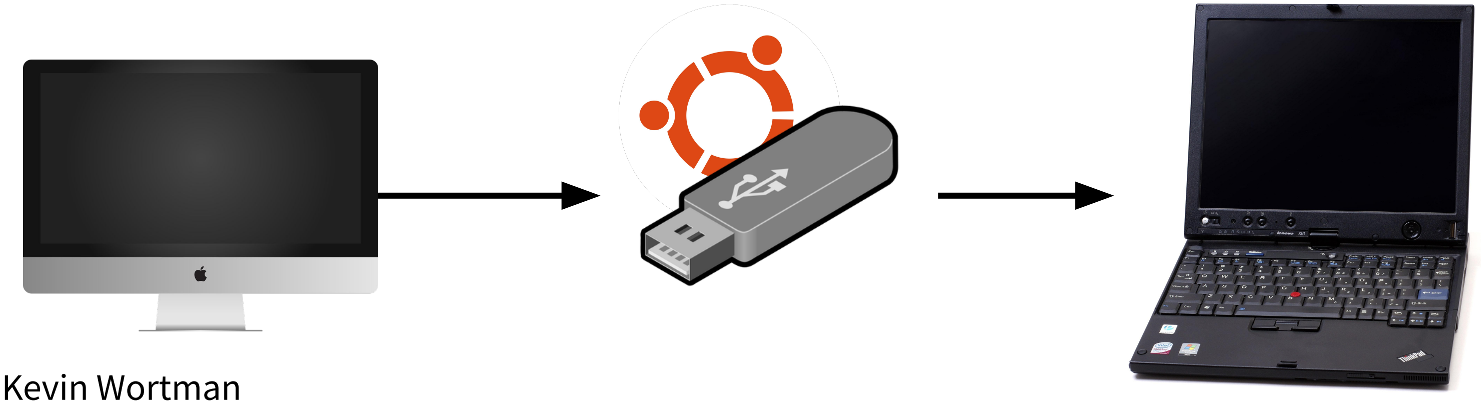
2. Installing Linux

Week 3 Lab

- Survey
- Installing Linux
- Bring your laptop
- Bring USB if possible
    - Some will be provided

Overview

- **Operating system**: software that manages hardware and provides platform for other software
  - macOS, Windows, Linux, ...
- Computer runs one operating system at a time
- Install Linux: copy Linux OS software to computer storage
  - Replaces existing OS



Download .iso

- .iso: “image” of contents of USB
- ubuntu-20.04.5-desktop-amd64.iso (3.4 GB)
- Use any computer to download
- Large
- USB must be 4 GB or larger (common)

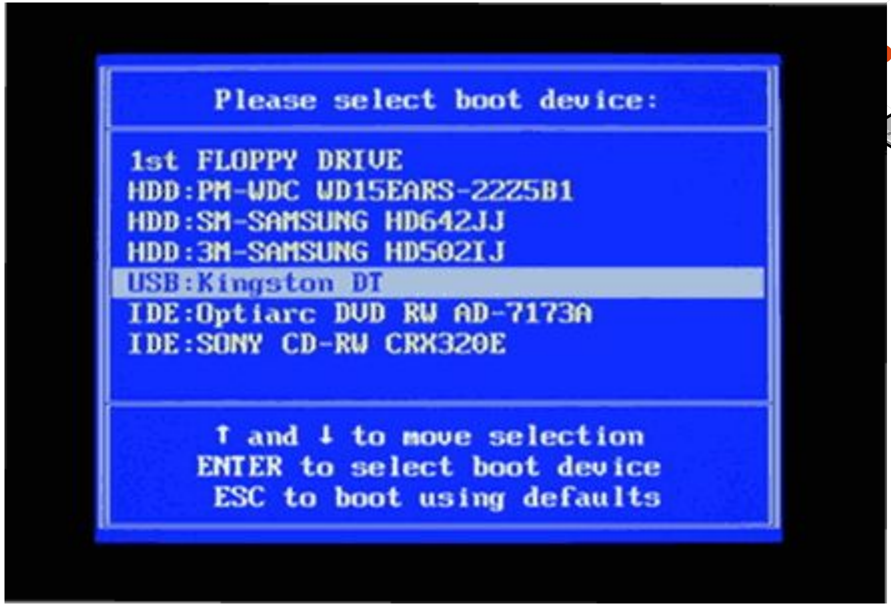
Create USB

- Need to write ubuntu-20.04.5-desktop-amd64.iso to USB
- Erases USB contents
  - You can reformat after install
- Need to use image-writing software
  - balenaEtcher (macOS, Windows, Linux)
  - Startup Disk Creator (Ubuntu)

Boot from USB

- Insert USB into computer for install
- Restart
- Wait for Power On Self Test (POST) = logo appears
- Press button for **Boot Menu**
- Possible boot menu keys:
  - **F12 (Lenovo, Dell)**
  - Escape
  - F2
  - F10
- In doubt: Google “*manufacturer* boot menu key”
  - Ex: “lenovo boot menu key”
  - (hardest part)

Boot Menu - Choose USB

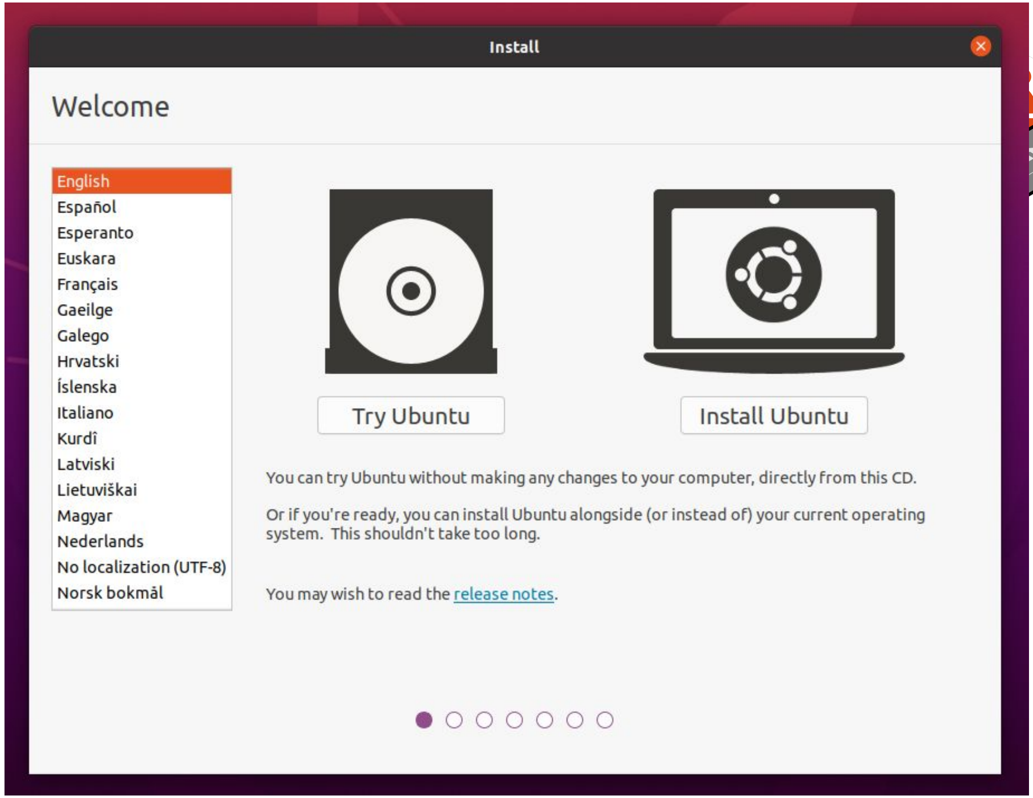


```
Please select boot device:

1st FLOPPY DRIVE
HDD:PM-WDC WD15EARS-22Z5B1
HDD:SM-SAMSUNG HD642JJ
HDD:3M-SAMSUNG HD502IJ
USB:Kingston DT
IDE:Optiarc DVD RW AD-7173A
IDE:SONY CD-RW CRX320E

↑ and ↓ to move selection
ENTER to select boot device
ESC to boot using defaults
```

Ubuntu Setup: Install



Install

Welcome

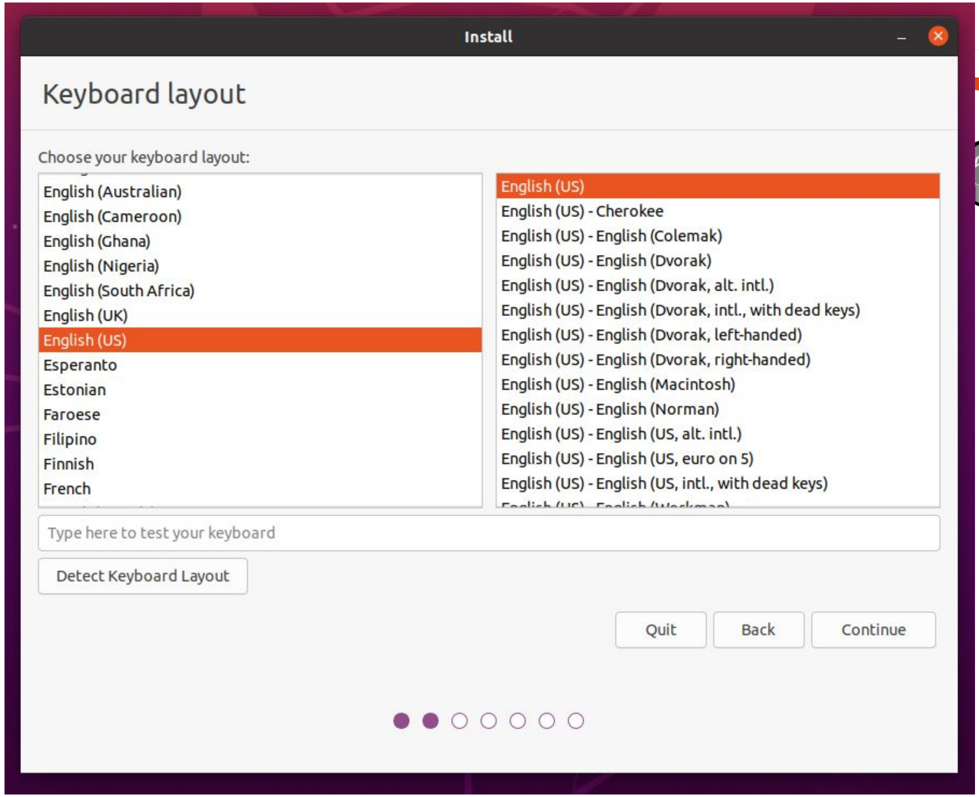
English
Español
Esperanto
Euskara
Français
Gaeilge
Galego
Hrvatski
Íslenska
Italiano
Kurdî
Latviski
Lietuviškai
Magyar
Nederlands
No localization (UTF-8)
Norsk bokmål

You can try Ubuntu without making any changes to your computer, directly from this CD.

Or if you're ready, you can install Ubuntu alongside (or instead of) your current operating system. This shouldn't take too long.

You may wish to read the release notes.

Ubuntu Setup: Language


Install

Keyboard layout

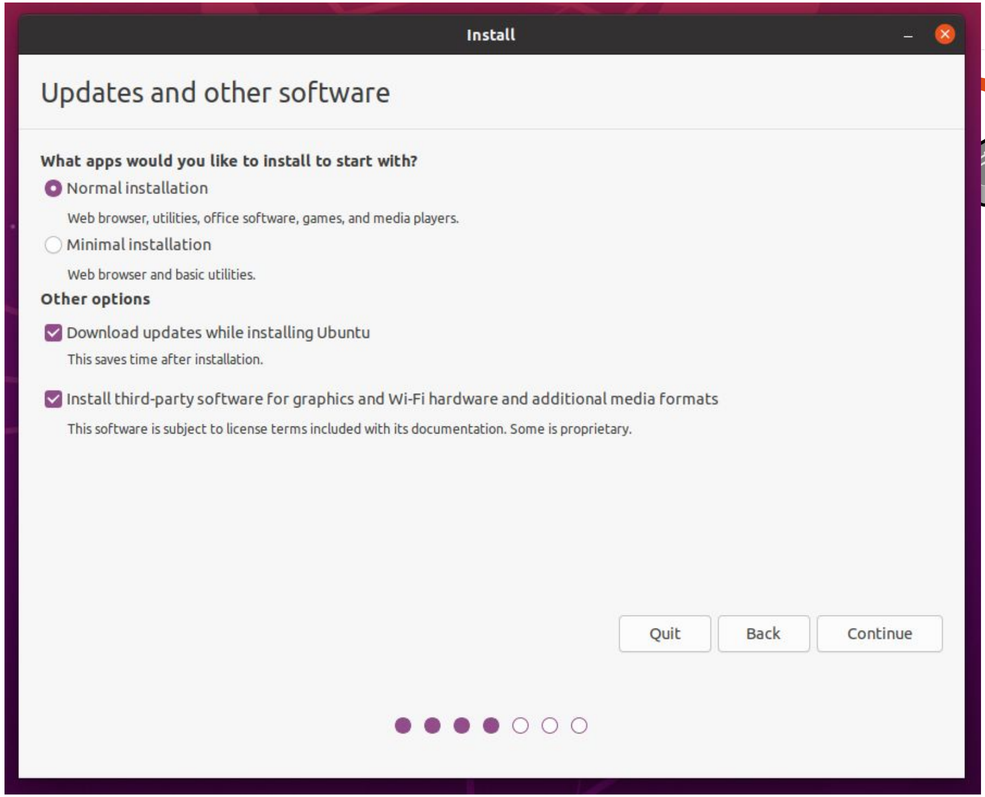
Choose your keyboard layout:

English (Australian)
English (Cameroon)
English (Ghana)
English (Nigeria)
English (South Africa)
English (UK)
English (US)
Esperanto
Estonian
Faroese
Filipino
Finnish
French

English (US)
English (US) - Cherokee
English (US) - English (Colemak)
English (US) - English (Dvorak)
English (US) - English (Dvorak, alt. intl.)
English (US) - English (Dvorak, intl., with dead keys)
English (US) - English (Dvorak, left-handed)
English (US) - English (Dvorak, right-handed)
English (US) - English (Macintosh)
English (US) - English (Norman)
English (US) - English (US, alt. intl.)
English (US) - English (US, euro on 5)
English (US) - English (US, intl., with dead keys)

Type here to test your keyboard

Ubuntu Setup: Updates



Updates and other software

What apps would you like to install to start with?

[✓ Normal installation]
Web browser, utilities, office software, games, and media players.

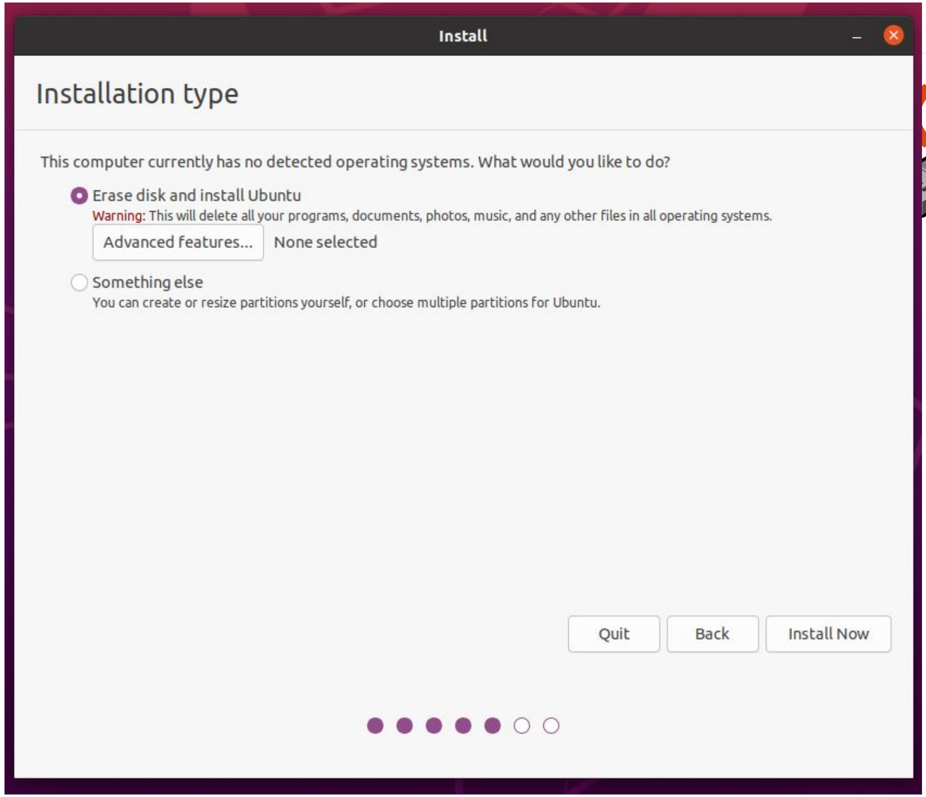
Minimal installation
Web browser and basic utilities.

Other options

[✓ Download updates while installing Ubuntu]
This saves time after installation.

[✓ Install third-party software for graphics and Wi-Fi hardware and additional media formats]
This software is subject to license terms included with its documentation. Some is proprietary.

Ubuntu Setup: Disk



Installation type

This computer currently has no detected operating systems. What would you like to do?

[✓ Erase disk and install Ubuntu]

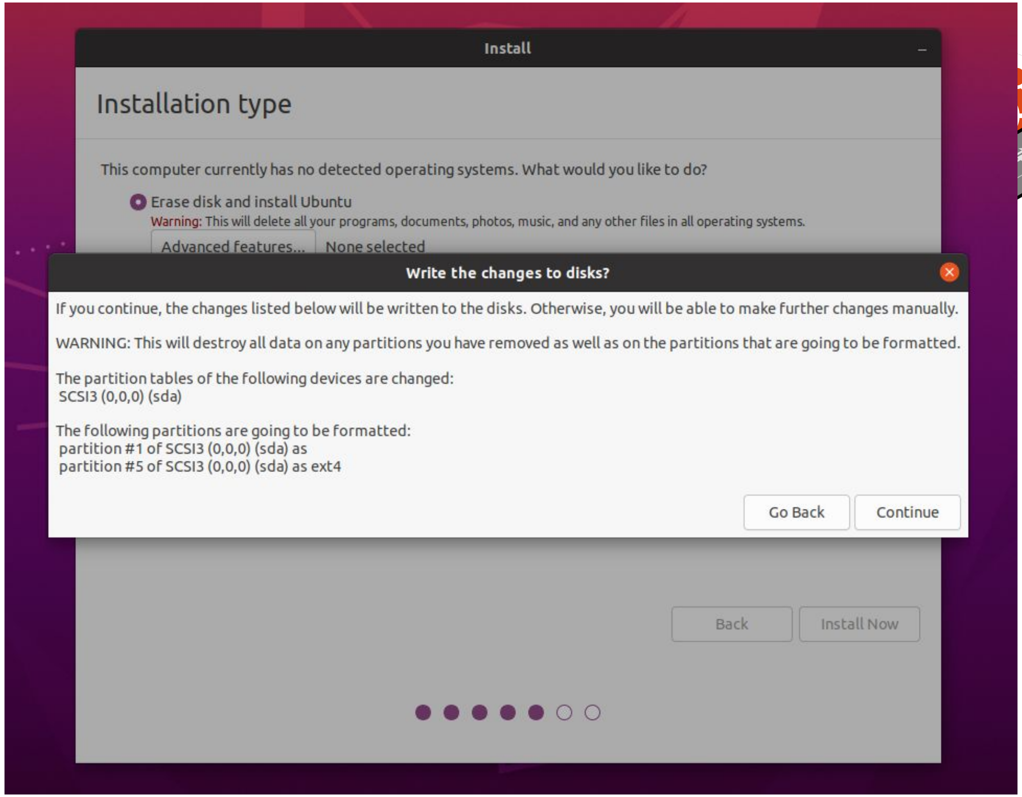
Warning: This will delete all your programs, documents, photos, music, and any other files in all operating systems.

Advanced features... None selected

Something else

You can create or resize partitions yourself, or choose multiple partitions for Ubuntu.

Ubuntu Setup: Confirm - Point of No Return


Install

Installation type

This computer currently has no detected operating systems. What would you like to do?

Erase disk and install Ubuntu

Warning: This will delete all your programs, documents, photos, music, and any other files in all operating systems.

Advanced features... None selected

Write the changes to disks?

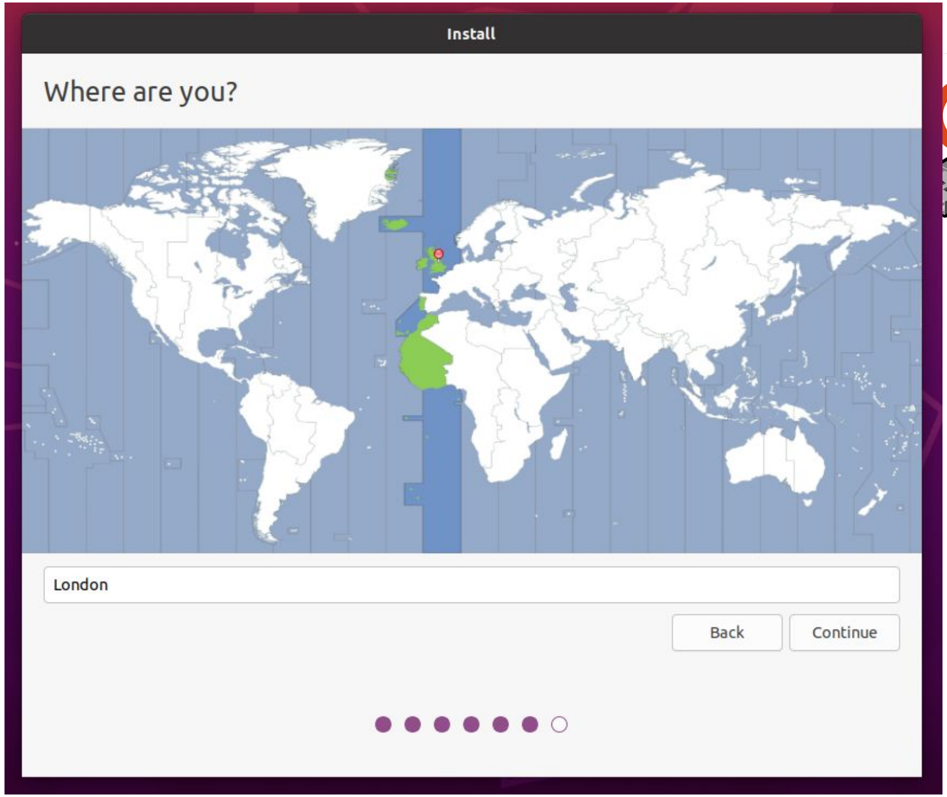
If you continue, the changes listed below will be written to the disks. Otherwise, you will be able to make further changes manually.

WARNING: This will destroy all data on any partitions you have removed as well as on the partitions that are going to be formatted.

The partition tables of the following devices are changed:
SCSI3 (0,0,0) (sda)

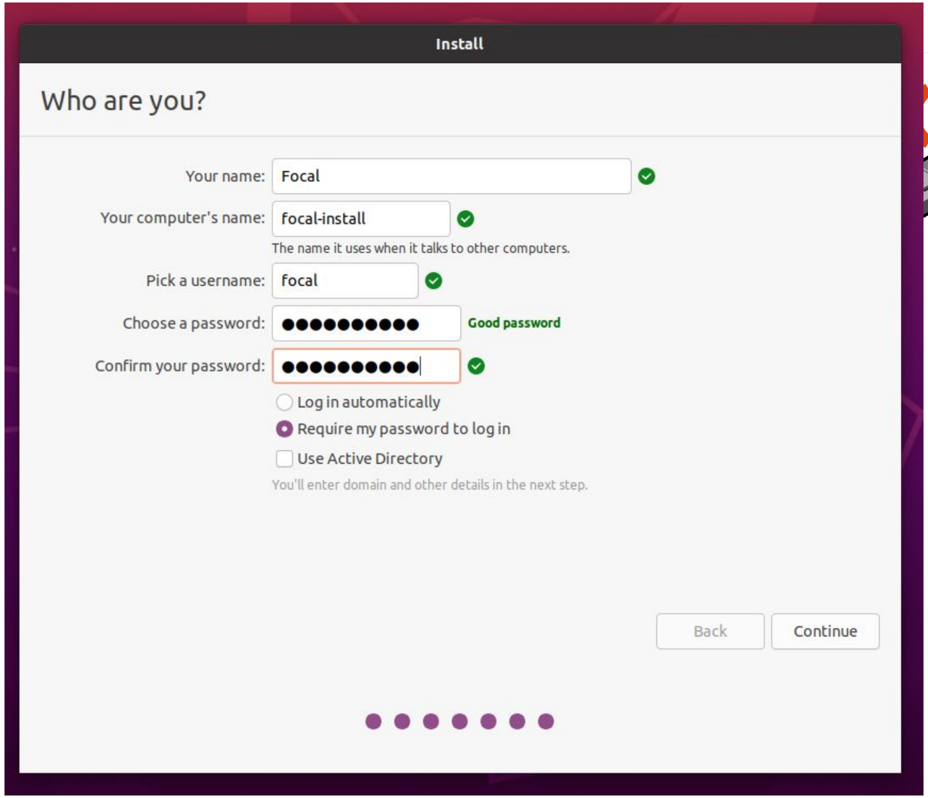
The following partitions are going to be formatted:
partition #1 of SCSI3 (0,0,0) (sda) as
partition #5 of SCSI3 (0,0,0) (sda) as ext4

Ubuntu Setup: Location



@copyright Kevin Wortman

Ubuntu Setup: Username/Password



Who are you?

Your name: Focal

Your computer's name: focal-install

The name it uses when it talks to other computers.

Pick a username: focal

Choose a password: ●●●●●●●●●●● Good password

Confirm your password: ●●●●●●●●●●●

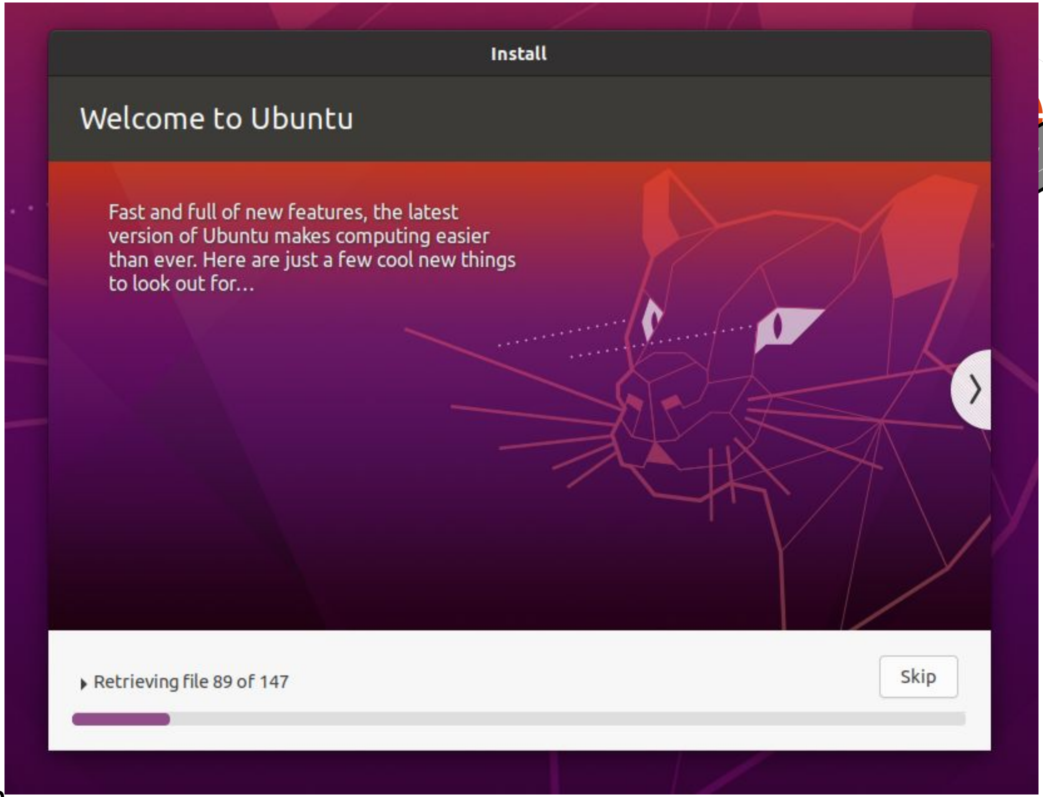
( ) Log in automatically

(●) Require my password to log in

[ ] Use Active Directory

You'll enter domain and other details in the next step.

Ubuntu Setup: Copying

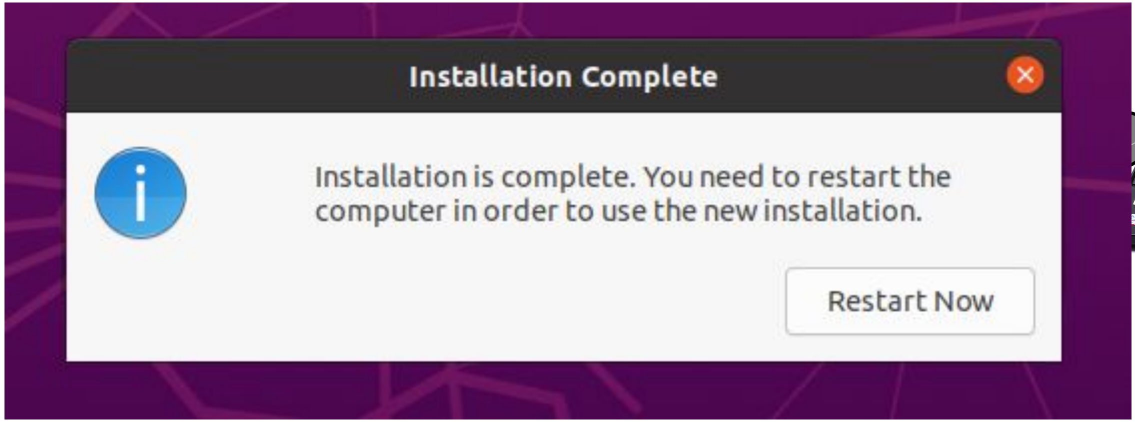


Welcome to Ubuntu

Fast and full of new features, the latest version of Ubuntu makes computing easier than ever. Here are just a few cool new things to look out for…

Retrieving file 89 of 147

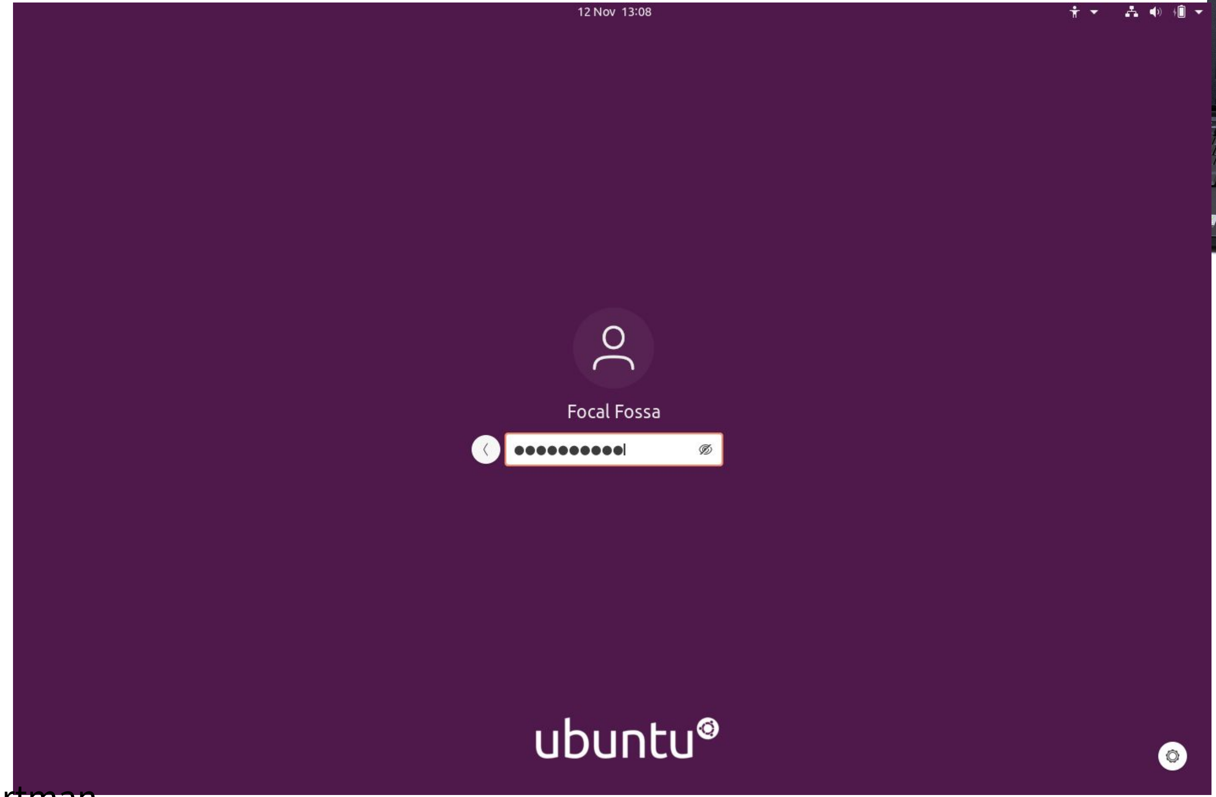
Ubuntu Setup: Reboot


Installation Complete

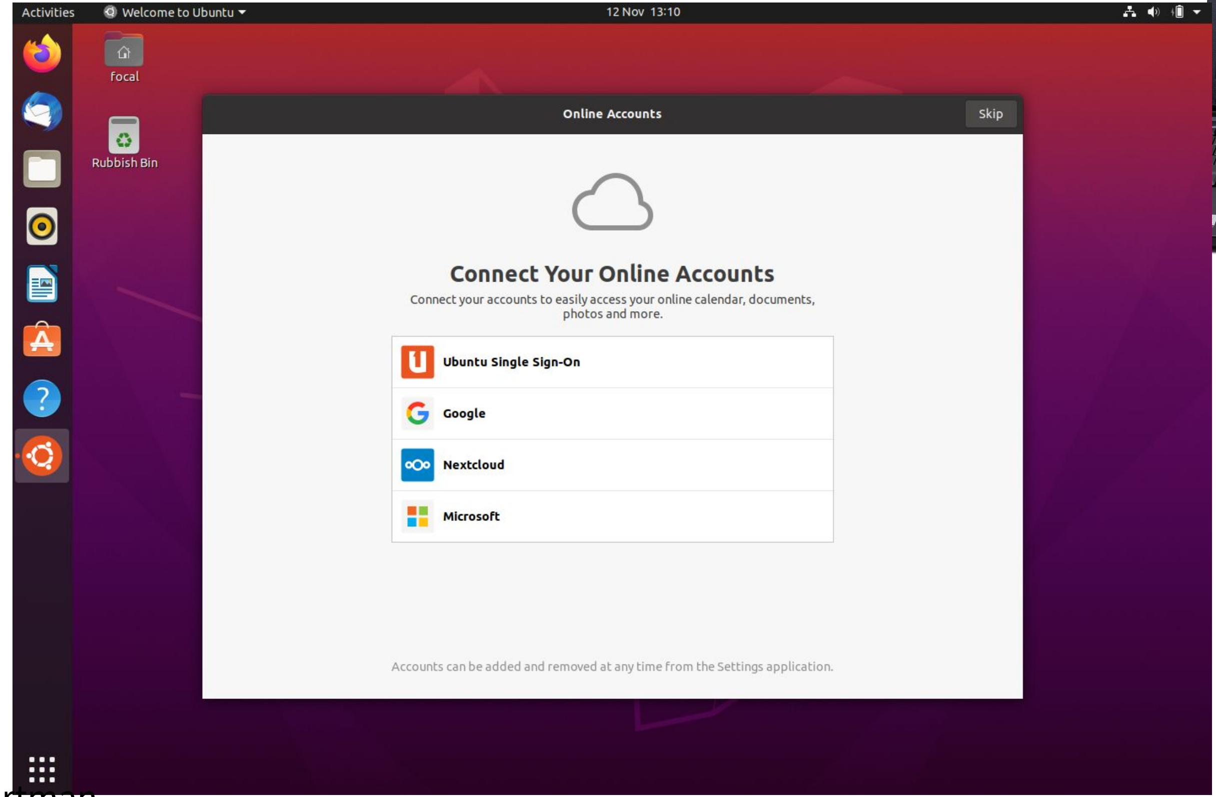
Installation is complete. You need to restart the computer in order to use the new installation.

Remove the USB drive when it asks.

Ubuntu Setup: Login



Ubuntu Setup: Login



Online Accounts

Connect Your Online Accounts

Connect your accounts to easily access your online calendar, documents, photos and more.

- Ubuntu Single Sign-On
- Google
- Nextcloud
- Microsoft

Accounts can be added and removed at any time from the Settings application.

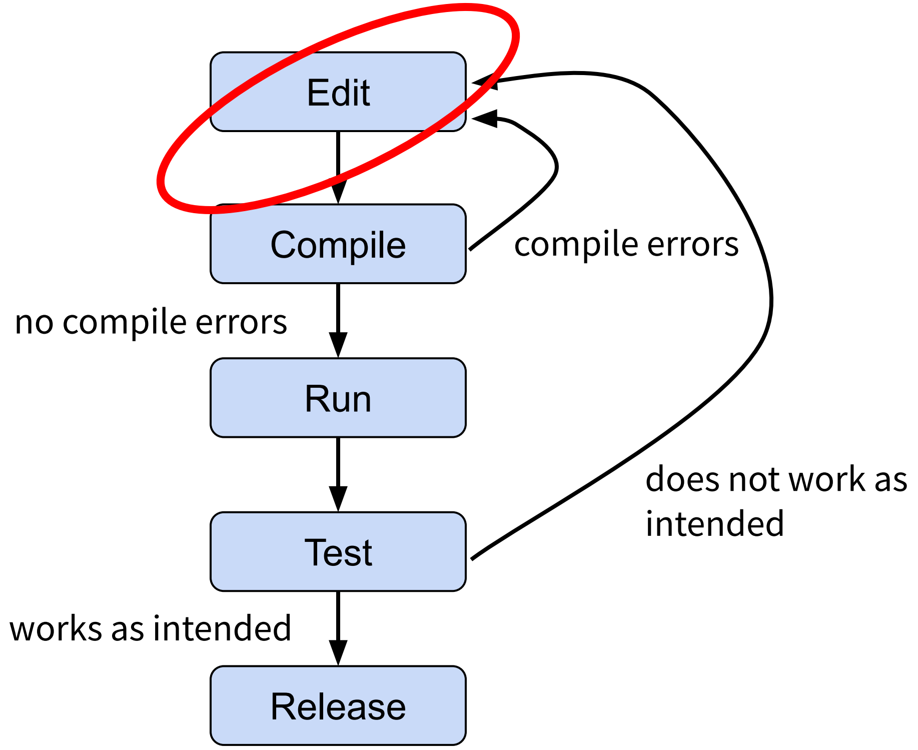
Development Tools (clang++, VS Code)

As described in Linux & Tools:

```
$ wget -q https://raw.githubusercontent.com/mshafae/tusk/main/quickinstall.sh -O- | sh
```

3. Syntax and Comments

The Edit-Compile-Run Cycle



Syntax and Semantics

	Syntax	Semantics
In general	source code **structure**	source code **meaning**
`cout << “Hello world!”;`	cout, then <<, then “Hello world!”, then ;	print out Hello World

Syntax Patterns

Same as cppreference.com:

Syntax is...	Notation
verbatim (write exactly as-is)	**bold**
fill-in-the-blank	*italics*
optional	has(optional)
may be repeated	ellipsis...

Pattern of a Source File

source-file:

directive, declaration, or definition...

Semantics:

- The compiler processes each *directive, declaration,* or *definition* in top-to-bottom order.

Hello World

```
// our hello world program
#include <iostream>

int main(int argc, char* argv[]) {
  std::cout << "Hello World!" << std::endl;
  return 0;
}
```

directive

definition

Pattern for a Program

program:

a program is one or more *source files* that contains exactly one *main function definition*

Semantics:

- The program starts by executing **main**
- The return value of **main** is the exit code of the program

Hello World

```
// our hello world program
#include <iostream>

int main(int argc, char* argv[]) {
  std::cout << "Hello World!" << std::endl;
  return 0;
}
```

definition

return value = exit code

Syntax Categories

Category	Semantics	Example
directive	orders the compiler to compile in a certain way	`#include <iostream>`
declaration	introduce the name of a variable, function, or data type	`int increase(int value);`

definition	declaration that also includes the body of a function or data type	`int decrease(int value) {`
		`  return value - 1;`
		`}`

statement	perform one step of an algorithm inside a function body	`cout << “Hello world”;`
expression	inside a statement, use operators to calculate a value	`(price + tax)`

Pattern for Main Function Definition

definition:

> **int main(int argc, char* argv[]) {**
>
>   *statement...*
>
> **}**

Semantics:

- Execute *statement...* in **top-to-bottom order**
- **bold** syntax is boilerplate

Fill-in-the-Blanks are Interchangeable

- You can fill a blank with **any** syntax of the matching type
- In

  directive, declaration, or definition...

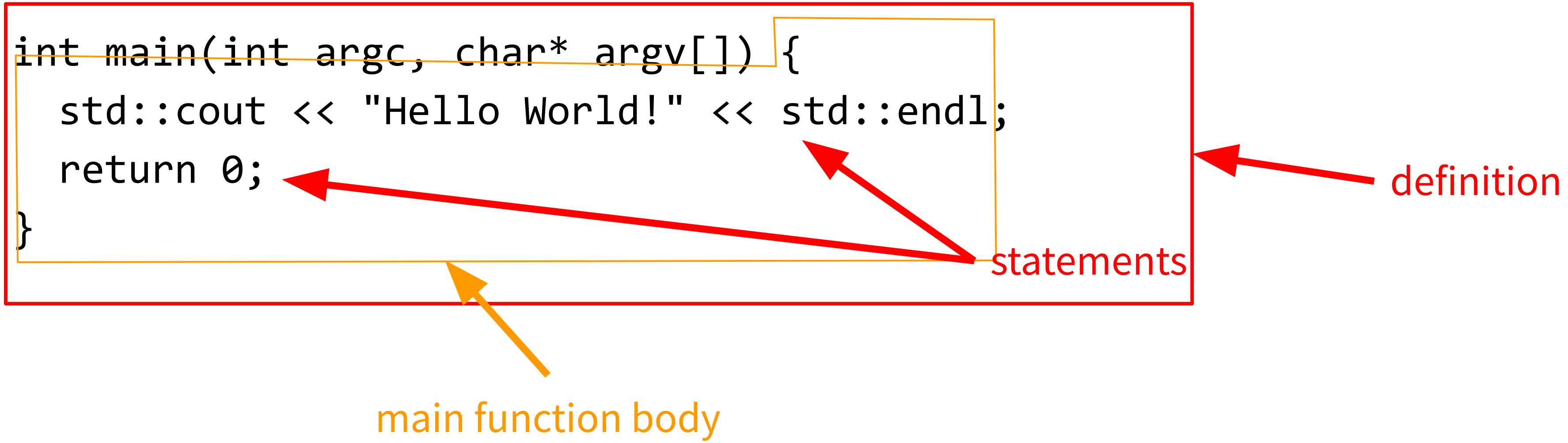
  you can fill in any kind of *directive* or *declaration* or *definition*
- In

  statement...

  you can fill in any kind of *statement*

Hello World

```
// our hello world program
#include <iostream>
```



```
int main(int argc, char* argv[]) {
  std::cout << "Hello World!" << std::endl;
  return 0;
}
```

Whitespace

- **Whitespace**: invisible formatting (space, tab, newline)
- Ignored by compiler
- Can go in between other syntax

Comments

- **Comment**: text in source code that is ignored by the compiler
- Purpose: notes, rationale, authorship, copyright
- Audience: other programmers, your future self
- Like whitespace, is allowed anywhere

comment:

// text...

Semantics:

- Compiler ignores // and *text...*

4. Code Style

Clean Code

"Clean code is code that is easy to understand and easy to change." --Carl Vuorinen

- Source code is for **human** consumption
- Code lifetime
    - **write once**
    - **read many times**
- Clarity matters
    - Hard for you to debug unclear code
    - Coworkers
    - Future self
- Valued in job market

Clean versus Unclean Whitespace

```
int main(int argc, char* argv[]) {
  std::cout << "Hello World!" << std::endl;
  return 0;
}

int main(int argc,char* argv[]){std::cout<<"Hello
World!"<<std::endl;return 0;}
```

Style Guide

- **Style guide**: defines clean/unclean code
- Living document
- We use Google C++ Style Guide
- Common issues in lab 2:
    - Horizontal Whitespace
    - Vertical Whitespace
    - Function Declarations and Definitions (curly brace { placement)

5. Formatting and Diff Output

Ideal Division of Labor

- **Business Logic:** the human meaning of algorithm data
- Programs
  - **Cannot** understand business logic or design algorithms
  - Can perform tedious, repetitive work flawlessly, quickly, cheaply
- Humans
  - **Can** understand business logic and design algorithms
  - Busy-work is tedious, error-prone, expensive
- Division of Labor Best Practice
  - Humans think about business logic and algorithms
  - Computer programs do repetitive work

Automating Clean Code

- **Focus of lab 2**
- Program (not person) checks code
- Corresponds to Google C++ Style Guide
- clang-format: checks syntax
  - whitespace, variable names, ...
- **linter** (clang-tidy): checks logic errors
  - covered soon

No Format Errors

```
$ ./check_formatting
2023-02-03 17:24:13,465 - INFO - Checking format for file:
/home/csuftitan/cpsc-120-solution-lab-02/part-1/fahrenheit_to_celsius.cc
2023-02-03 17:24:15,422 - INFO - 😀 Formatting looks pretty good! 🥳
2023-02-03 17:24:15,422 - INFO - This is not an auto-grader.
2023-02-03 17:24:15,422 - INFO - Make sure you followed all the instructions and
requirements.
```

Format Errors

```
  int main(int argc, char const *argv[]) {
! std::cout << "Hello World!";
    return 0;
  }

--- 16,22 ----

  using namespace std;

  int main(int argc, char const *argv[]) {
!   std::cout << "Hello World!";
    return 0;
  }

2023-02-03 17:36:54,726 - ERROR - 🤯😳😤😫🤬
2023-02-03 17:36:54,726 - ERROR - Your formatting doesn't conform to the Google C++ style.
2023-02-03 17:36:54,726 - ERROR - Use the output from this program to help guide you.
2023-02-03 17:36:54,726 - ERROR - If you get stuck, ask your instructor for help.
2023-02-03 17:36:54,726 - ERROR - Remember, you can find the Google C++ style online at
https://google.github.io/styleguide/cppguide.html.
```

Contextual Diff

- GNU Diffutils: programs for identifying differences between files
- Contextual Diff: prints differences **with surrounding context**
- Compares **unclean source** to hypothetical **cleaned source**
- **Hunk** of differences: area that differs

Contextual Diff Format

```
***************
*** first-unclean-line, last-unclean-line ****
  unclean-line…
--- first-clean-line, last-clean-line ----
  clean-line…
```

Left column:

!	lines differ
+	line added to unclean
-	line deleted from unclean

Example: Contextual Diff Output

```
2023-02-03 17:36:54,718 - ERROR - Error: Formatting needs improvement.
2023-02-03 17:36:54,726 - WARNING - Contextual Diff
*** Student Submission (Yours)

--- Correct Format

***************

*** 16,22 ****

  using namespace std;

  int main(int argc, char const *argv[]) {
! std::cout << "Hello World!";
      return 0;
  }

--- 16,22 ----

  using namespace std;

  int main(int argc, char const *argv[]) {
!   std::cout << "Hello World!";
      return 0;
```

Debugging Format Errors

1. Run format check
2. Identify lines with differences; left column is one of ! + -
3. Identify difference between unclean(top) and clean (bottom) source
4. Edit source code to match clean
5. Save, go back to step 1

No Lint Errors

```
$ ./check_for_lint
2023-02-03 17:50:07,249 - INFO - Linting file:
/home/csuftitan/cpsc-120-solution-lab-02/part-2/quadratic_formula.cc
2023-02-03 17:50:23,639 - INFO - 😀 Linting passed 🥳
2023-02-03 17:50:23,641 - INFO - This is not an auto-grader.
2023-02-03 17:50:23,643 - INFO - Make sure you followed all the instructions and
requirements.
```

Lint Errors

```
2023-02-03 17:51:55,222 - INFO - Linting file:
/home/csuftitan/cpsc-120-solution-lab-02/part-2/quadratic_formula.cc
2023-02-03 17:51:57,236 - INFO - stderr:
/home/csuftitan/cpsc-120-solution-lab-02/part-2/quadratic_formula.cc:19:10: warning: unused
variable 'total' [-Wunused-variable]
  double total;
         ^
1 warning generated.
2023-02-03 17:52:01,691 - ERROR - Linter found improvements.
2023-02-03 17:52:01,691 - WARNING -
/home/csuftitan/cpsc-120-solution-lab-02/part-2/quadratic_formula.cc:19:10: warning: variable
'total' is not initialized [cppcoreguidelines-init-variables]
  double total;
         ^
                = NAN
2023-02-03 17:52:01,691 - ERROR - 🤯😳😤😫🤬
2023-02-03 17:52:01,691 - ERROR - Use the output from this program to help guide you.
2023-02-03 17:52:01,691 - ERROR - If you get stuck, ask your instructor for help.
2023-02-03 17:52:01,691 - ERROR - Remember, you can find the Google C++ style online at
https://google.github.io/styleguide/cppguide.html.
```

Debugging Lint Errors

1. Run lint check
2. Identify error, line number, message
3. Edit source code to solve problem
4. Save, go back to step 1