ECS 198F: Introduction

Divyansh Rajesh Jain

UCDAVIS

Introduction

- Divyansh Rajesh Jain
- 4rd year CS and Managerial Economics Major
- Advised by Prof. Siena Saltzen

UCDAVIS

Objectives

- Bridge the gap between school curriculum and industry
- Teach about industry tools and practices
  - Unix and Developer Tools
  - Docker
  - Testing (QA)
  - CI/CD & DevOps
  - Intro to AI Agents
- Hands-on activities to reinforce lecture
- Group Final Project → Apply Course Material

UCDAVIS

Course Information

- Time: M / W 12:10 - 1:00 PM
- Location: Hart Hall 1316
- Lectures will be recorded (if no tech issues)
- Units: 2 (Change in Schedule Builder)
- Office Hours: TBD

UCDAVIS

Syllabus

- Grade Breakdown
  - 60% - 3 HW Assignments (20% each)
  - 40% - Final Project (Group Project in groups of 3 or 4)
- 5 penalty-free late days for any HW assignment
- Final Project **must be submitted by the deadline**
- https://divteaching.github.io/ecs198f/winter2026/

UCDAVIS

Getting to Know You!

- Name, Major?
- What do you hope to learn from this course?
- One fun fact about you.

What is Unix?

- OS made by Bell Lab researchers
- It was distributed → many copies were made
- Working Definition → Any System that supports the GNU toolset
- Linux, MacOS are all examples of Unix-like OS

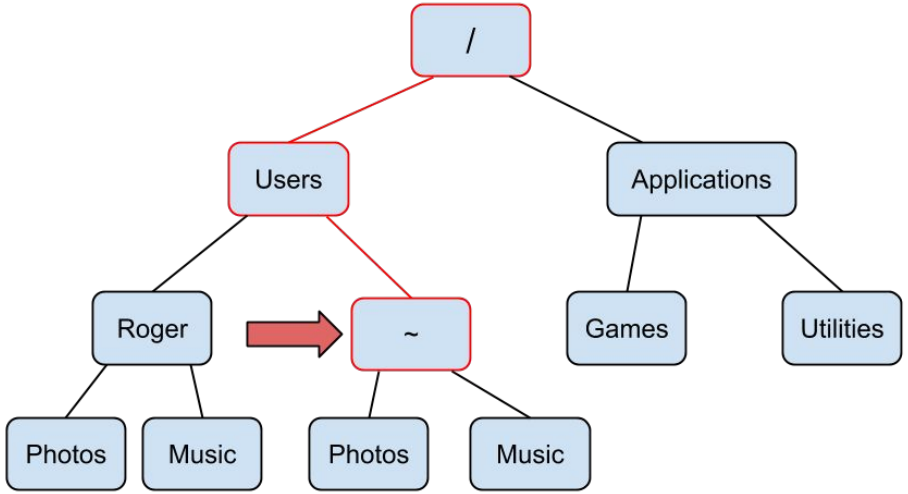
Why the Command Line?

- What is a Shell (Command Line)?
- Gives more control over your computer
- Allows programmers to work more efficiently
- Useful when no GUI → Connect to Remote Server

File Structure

- Tree-like Structure that represents the hierarchy of

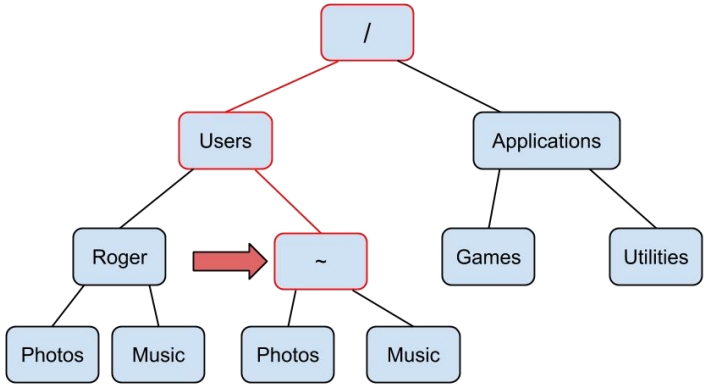
  files stored in computer



UCDAVIS

Navigating in the Terminal

- Paths
  - Absolute Path - The exact path to get to a folder/file
  - Relative Path - The path to get to folder/file from current location (./ = Current Directory, ../ = Previous Directory)



UCDAVIS

Navigating in the Terminal

- pwd command (Print Working Directory)
  - Show the absolute path of current location
- ls command (List)
  - Listing all files/directories in current location
- cd command (Change Directory)
  - cd <path> → navigate to new folder

Navigating in the Terminal

- mkdir command (Make Directory)
  - mkdir <folder> → This is to make a new folder
- mv command (Move)
  - mv <source> <destination> → Move files/folders
- touch command (Create Files)
  - touch <file name> → Create new file

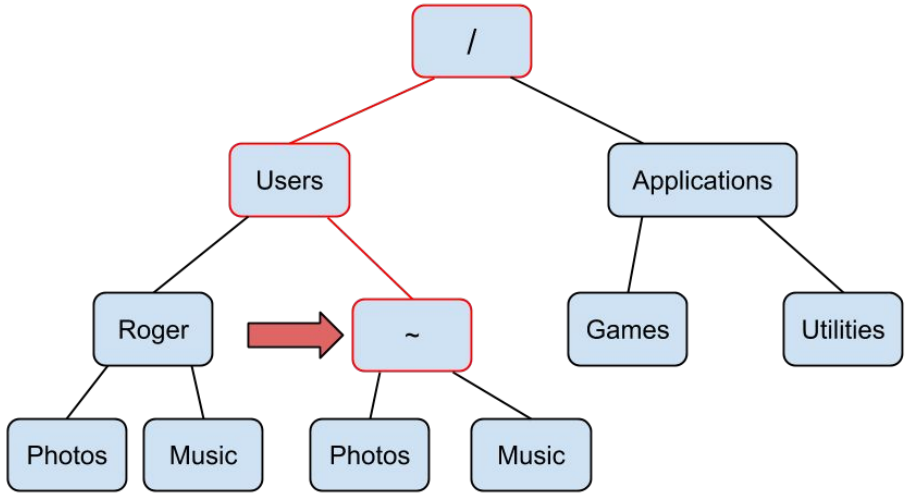
Navigating in the Terminal

- rm command (Remove)
  - rm [-r] <file> → Delete file/folder permanently
- cp command (Copy)
  - cp [-r] <source> <destination> → Copy file/folder
- cat command (Concatenate)
  - cat <file1> <file2> … → Concatenate Files into 1 output

Exercise

1. How to get the current directory?
2. How to navigate to the Games folder?
3. How to delete the Games folder?
4. How to copy the Utilities folder to

   /Applications/NewUtilities/



UCDAVIS

Terminal Text Editing

- Vim → vimtutor (VSCode extension too!)
- Nano → GNU Text Editor (Simple)

Remote Connections

- ssh command (Secure shell)
  - ssh <username>@<hostname>
- Allows to connect to another machine remote
- CSIF
  - username - UC Davis Kerberos Username
  - hostname - pcxx.cs.ucdavis.edu (pc01 - pc43)

UCDAVIS

Remote Connections

- Authentication Method
  - Password (Insecure) - Because passwords are limited in complexity
  - SSH Key (Secure) - Uses Cryptographic Methods to authenticate connection

SSH Keys

- Public-Private Key Structure
  - Public - Share with others (Encrypt Data)
  - Private - Don’t share with others (Decrypt Data)
- SSH Connections using SSH Keys
  - Put Public Key on Server → Now when SSH, the utility should use private key to authenticate with the server

Remote Connections

- scp command (Secure Copy)
    - scp [-r] <source> <destination> → To copy file/folder to or from remote machine

scp [-r] <local_file/folder> <username>@<hostname>:<remote_destination> →

Copy file from local to remote

Bash Scripting

- Instead of typing commands into terminal → write programs that execute in shell to automate repetitive tasks
- Very useful → For example, used in Autograders (Gradescope)

Bash Scripting Basics

- Variables → Can hold integer, string, array, and dictionary values
  - var_name=<value>
    - String value: “string_value”
    - Integer Value: 100
    - Array Value: (item1 item2 … itemN)
    - Dictionary Value: declare -A <dictionary_name>
      - Insert: <dictionary_name>[<key>] = <value>

UCDAVIS

Bash Scripting Basics

- Reference Variables: $var_name
- For example, to print value of TEST_INT
  - echo “This is TEST_INT: $TEST_INT”
- Note: echo in bash prints values to the console

Bash Scripting Basics

- For Loop in Bash → Many different variants

for <iter_var> in <looping statement>; do

    <for_loop_body>

done

Bash Scripting Basics

- For-loop looping statements
  - <directory> → Iterates through that directory
  - {start..stop..step} → Range based iteration
  - “$(array_name[@])” → Iterate through an array

Bash Scripting Example

Cheat Sheet: https://devhints.io/bash

Introduction to Git

- Version Control Software Developed by Linus Torvalds (creator of Linux)
- Used to keep track of changes made to a codebase
- It is a Diff based Version Control System

Intro to Basic Git Concepts

- Repo → Codebase that Git is managing
- Staging → Prepare changes before commiting
- Commit → Save a snapshot of repository at a certain time
- Branch → Separate lines of Development
- Merge → Used to merge code between branches

Common Git CLI Commands

- git init → Initialize Git Repository (Repo)
- git add <file1> … → Add files to staging area
- git commit [-m <message>] → After staging files, commit those files
- git log → View all the commits in Repo

Common Git CLI Commands

- git branch <branch-name> → Create new branch
- git checkout <branch-name> → Switch Branch
- git merge <source branch-name> → Merge source branch into currently checked out branch

Thank you!

Next Time: More on Git & Remote Git Repositories

UCDAVIS