From knowing the definition of Linux Kernel to becoming a kernel hacker

Vaishali Thakkar
(vaishalithakkar.in, @kernel_girl)

Who Am I?

- Freelance Linux Kernel Developer
- Co-organizer of RGSoC
- Co-coordinator of Outreachy for Linux Kernel projects
- Linux Kernel and Open source evangelist
- Open learning and open education enthusiast/advocate

What this talk is about

Journey of

What this talk is about

Journey of

What this talk is about

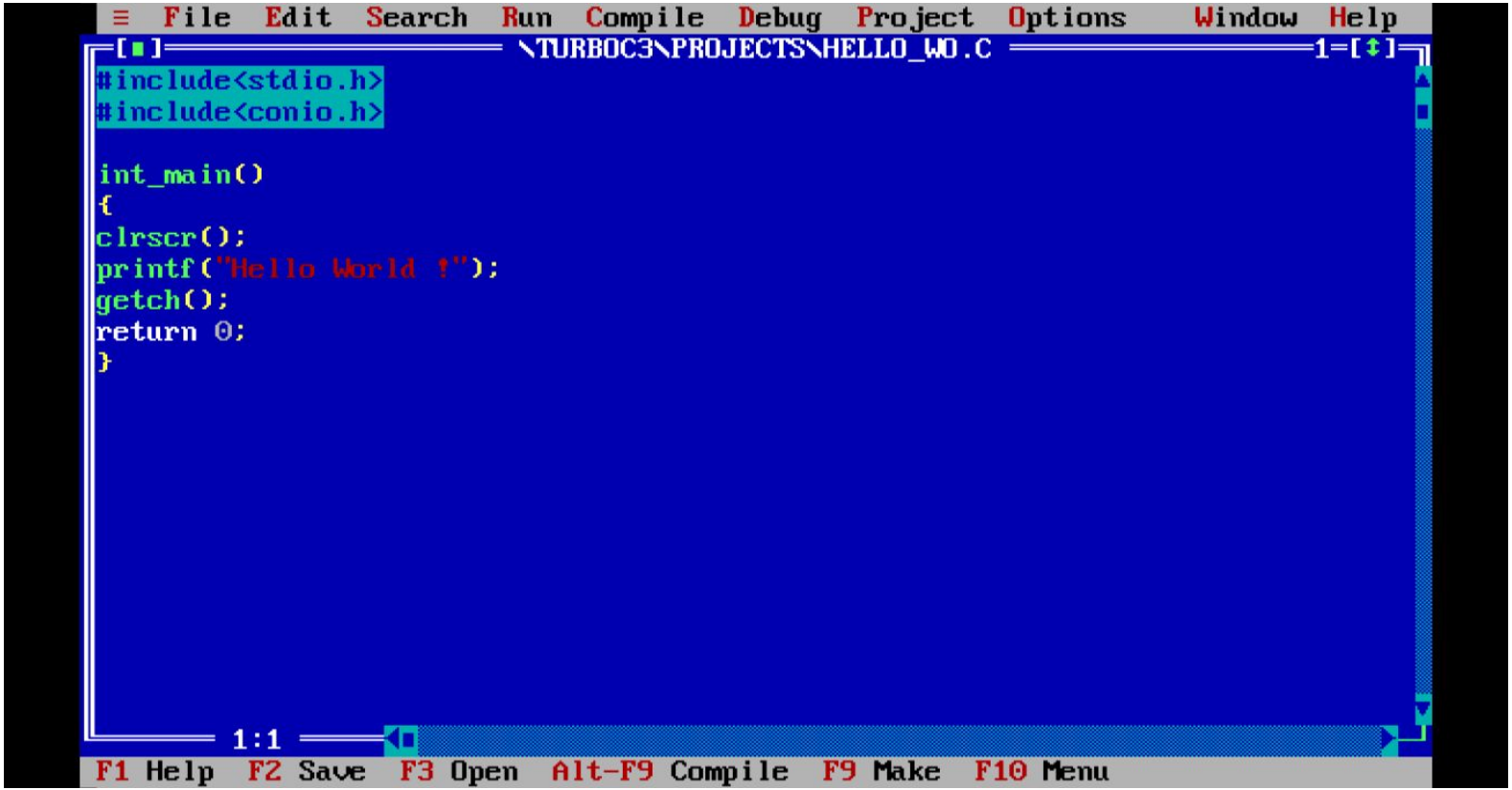
More about

My Idea of Linux Kernel Engineer v/s Me

1

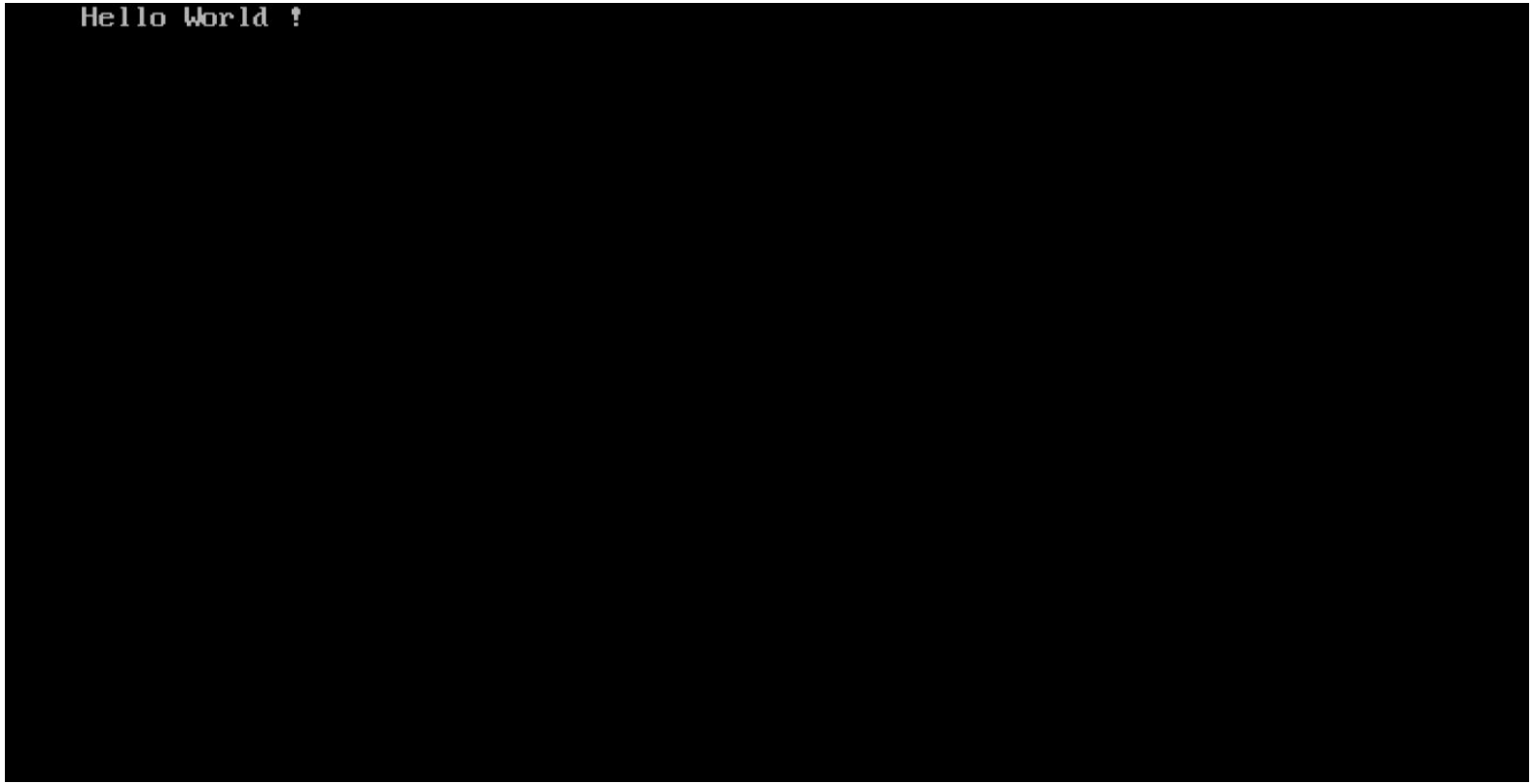
My Idea of Linux Kernel Engineer v/s Me

1



My Idea of Linux Kernel Engineer v/s Me

1



My Idea of Linux Kernel Engineer v/s Me

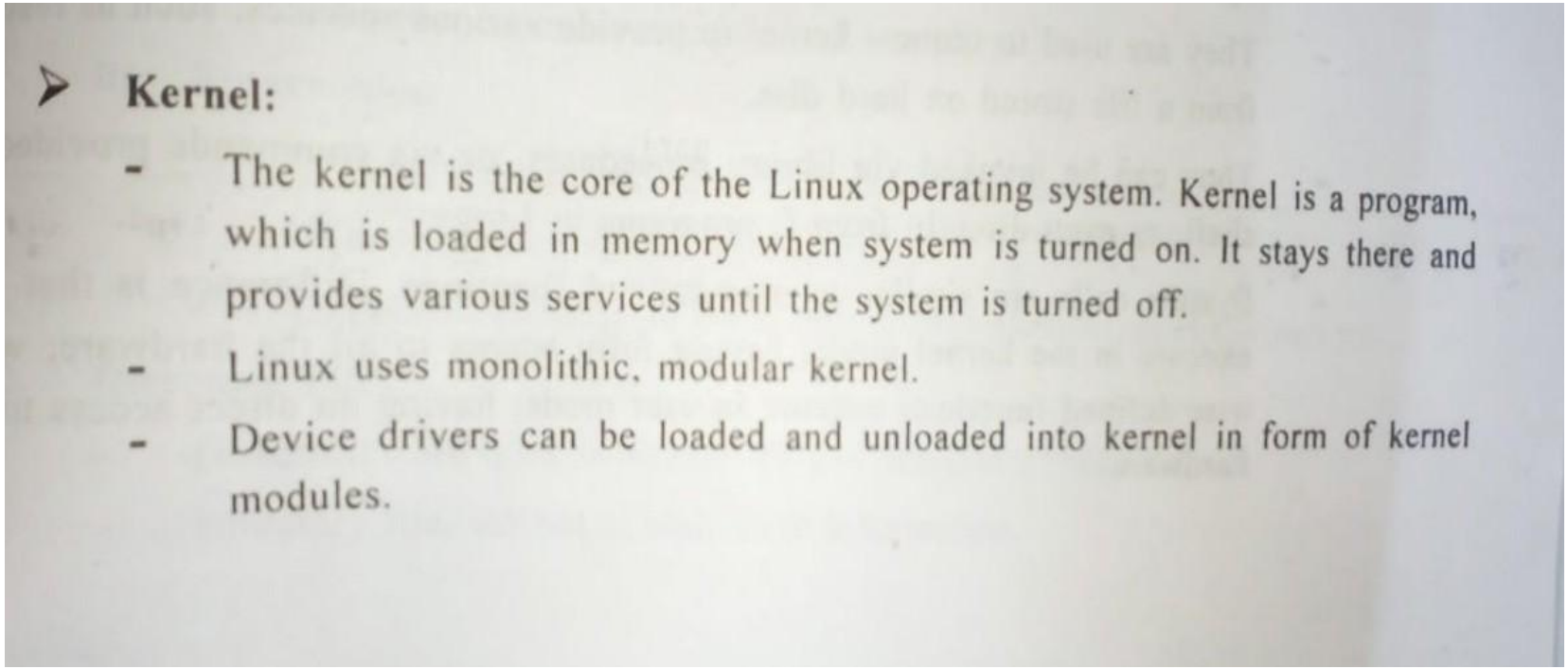
2

My Idea of Linux Kernel Engineer v/s Me

2

My Idea of Linux Kernel Engineer v/s Me

2

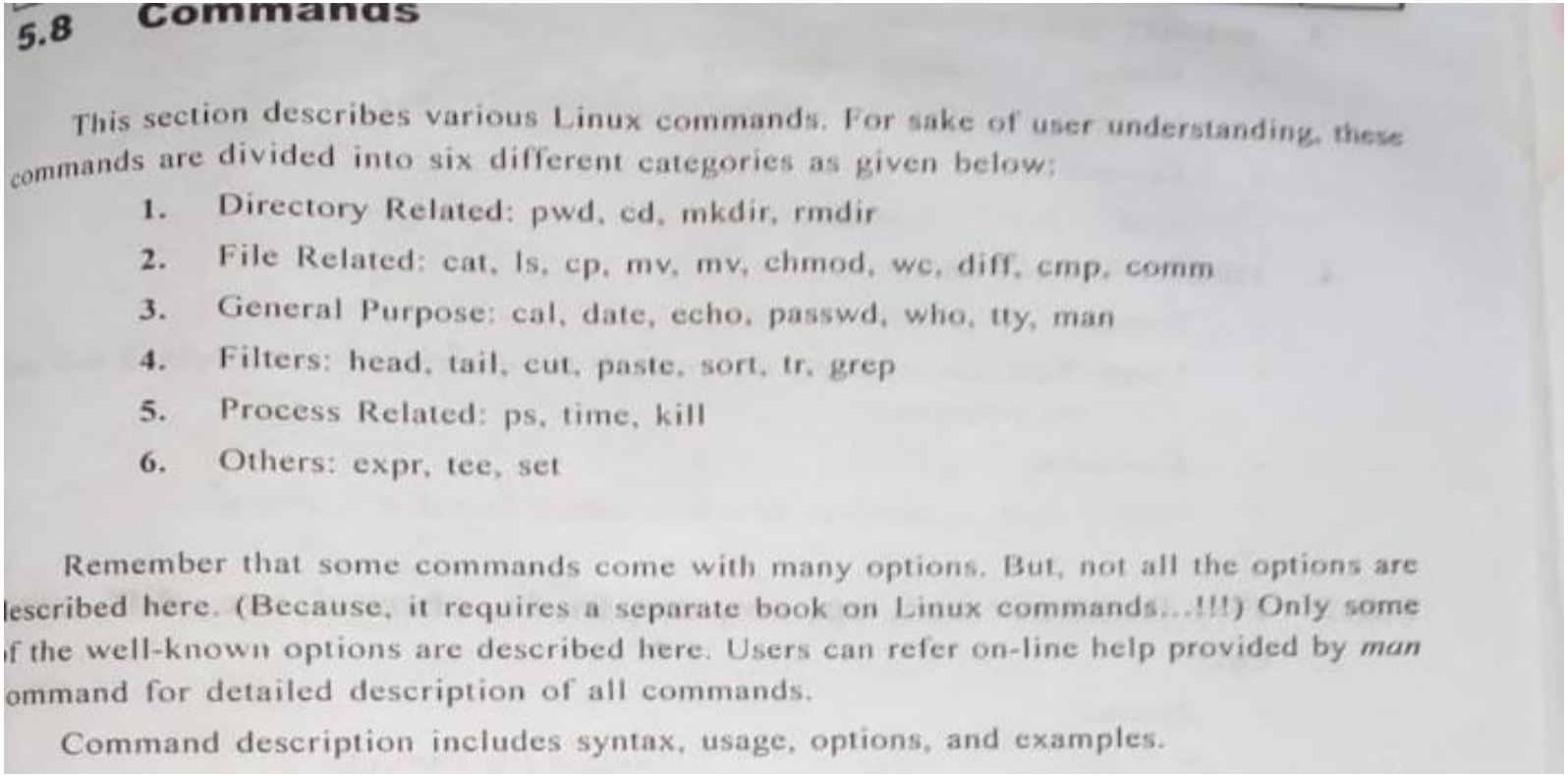


➢ **Kernel:**

- The kernel is the core of the Linux operating system. Kernel is a program, which is loaded in memory when system is turned on. It stays there and provides various services until the system is turned off.
- Linux uses monolithic, modular kernel.
- Device drivers can be loaded and unloaded into kernel in form of kernel modules.

My Idea of Linux Kernel Engineer v/s Me

2



5.8 Commands

This section describes various Linux commands. For sake of user understanding, these commands are divided into six different categories as given below:

1. Directory Related: pwd, cd, mkdir, rmdir
2. File Related: cat, ls, cp, mv, mv, chmod, wc, diff, cmp, comm
3. General Purpose: cal, date, echo, passwd, who, tty, man
4. Filters: head, tail, cut, paste, sort, tr, grep
5. Process Related: ps, time, kill
6. Others: expr, tee, set

Remember that some commands come with many options. But, not all the options are described here. (Because, it requires a separate book on Linux commands...!!!) Only some of the well-known options are described here. Users can refer on-line help provided by *man* command for detailed description of all commands.

Command description includes syntax, usage, options, and examples.

My Idea of Linux Kernel Engineer v/s Me

3

My Idea of Linux Kernel Engineer v/s Me

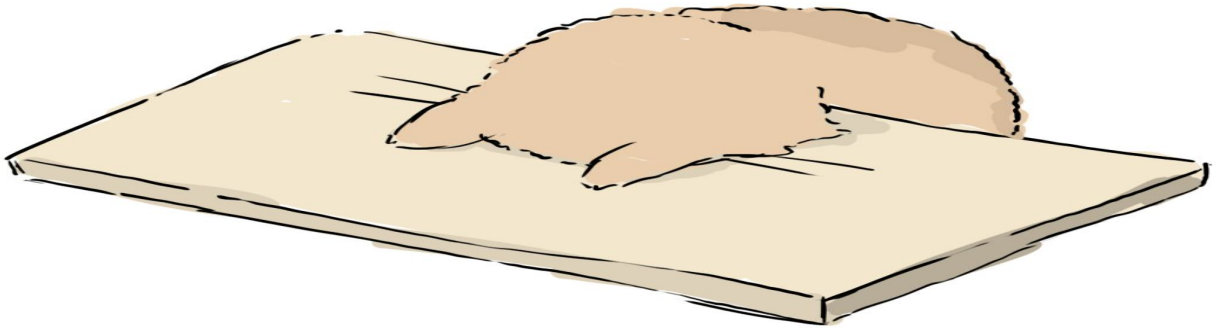
3

My Idea of Linux Kernel Engineer v/s Me

4

My Idea of Linux Kernel Engineer v/s Me

4



Then how did I started hacking on Linux Kernel?

Then how did I started hacking on Linux Kernel?

Then how did I started hacking on Linux Kernel?

Then how did I started hacking on Linux Kernel?

What I did?

- Installing Linux Distro

What I did?

Me after discovering GCC

What I did?

Discovering Outreachy

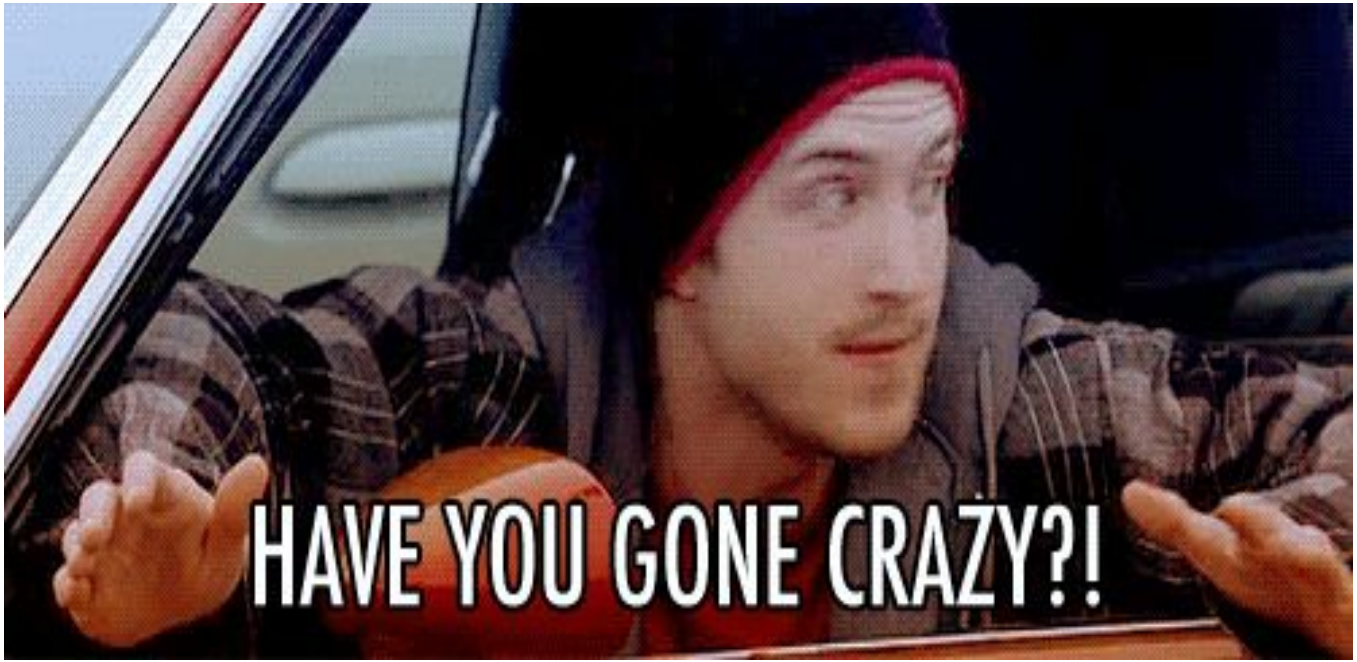
What I did?

"Thanks, your patch is applied"

What I did?

Linux kernel intern -> Linux Kernel Engineer

So you mean one need to get fractured to be a Linux Kernel hacker?



A Growth Mindset

Can positive attitude help you to be a Linux kernel hacker?

Like anything else, learning is a skill as well and you can get better at it when you start putting efforts!

Six self learning skills for Linux Kernel

1. Finding right resources

Six self learning skills for Linux Kernel

1. Finding right resources
    a. Source code and LWN articles

Six self learning skills for Linux Kernel

1. Finding right resources
    a. Source code and LWN articles
    b. Mailing list archives

Six self learning skills for Linux Kernel

1. Finding right resources
    a. Source code and LWN articles
    b. Mailing list archives
    c. Git {grep, blame, log, show} 

Six self learning skills for Linux Kernel

1. Finding right resources
    a. Source code and LWN articles
    b. Mailing list archives
    c. Git {grep, blame, log, show} 
    d. Conference talks and kernel developer blogs

Six self learning skills for Linux Kernel

2. Asking questions

    a. Important to figure out why you don’t know about X or why is it difficult to understand Y topic

Six self learning skills for Linux Kernel

2. Asking questions

    a. Important to figure out why you don’t know about X or why is it difficult to understand Y topic
    b. Good questions v/s bad questions

Six self learning skills for Linux Kernel

2. Asking questions

a. Important to figure out why you don’t know about X or why is it difficult to understand Y topic
b. Good questions v/s bad questions
c. Mailing list etiquettes

Six self learning skills for Linux Kernel

3. Understanding Maintainer’s style

a. Not all subsystem maintainers prefer same style of patch submission

Six self learning skills for Linux Kernel

3. Understanding Maintainer’s style

   a. Not all subsystem maintainers prefer same style of patch submission
   b. Patch series v/s patch per change

Six self learning skills for Linux Kernel

4. Automating the learning

    a. Use existing tools [Git grep tricks, Vim editor tricks]

Six self learning skills for Linux Kernel

4. Automating the learning

   a. Use existing tools [Git grep tricks, Vim editor tricks]
   b. Write your own scripts [bash, coccinelle] by recognizing the repetitive tasks/patterns

Six self learning skills for Linux Kernel

5. Keep updating your knowledge about kernel subsystems

a. New v/s old v/s deprecated APIs/subsystems

Six self learning skills for Linux Kernel

5. Keep updating your knowledge about kernel subsystems

a. New v/s old v/s deprecated APIs/subsystems
b. From Subsystem specific Git trees to stable tree

Six self learning skills for Linux Kernel

6. Improving your craft as a programmer
    a. Recognizing the difference between good/bad/intelligent code

Six self learning skills for Linux Kernel

6. Improving your craft as a programmer
    a. Recognizing the difference between good/bad/intelligent code
    b. Learning from other programmers

Let’s be open about sharing our learning in Linux Kernel and build something new/exciting/impossible together!