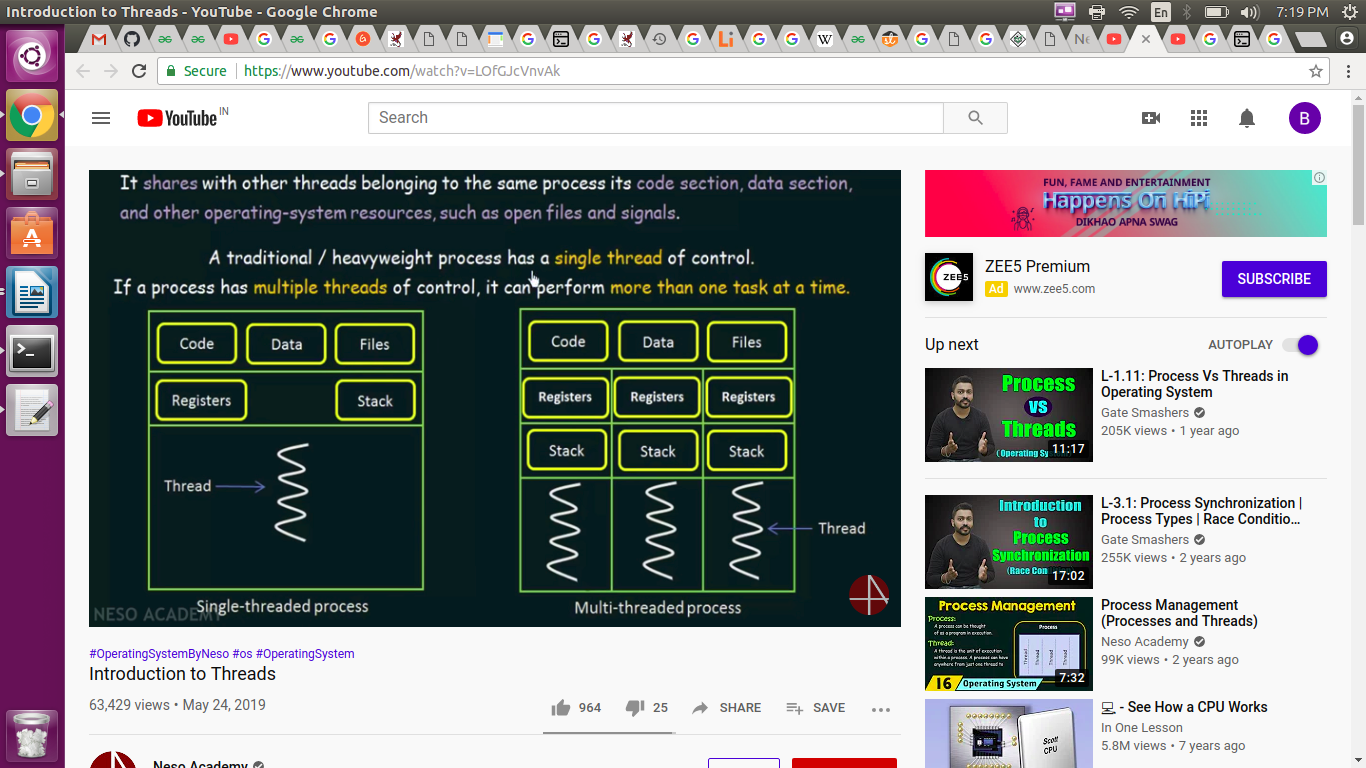
**Treading:**

Program under execution is known as **process** and thread is a basic unit of execution.

Each program may have number of processes associated with it and each process can have number of threads executing in it.

Threads shares with other threads belonging to the same process its code section, data section and other operating system resources. If a process has single thread it will perform single task, if a process has multiple threads of control, it can perform more than one task at a time.



**Single threaded proess:**

The block considered as process. The code, data, files, registers and stack belonging to the single thread process. This perform only one task at a time.

**Multi threaded proess:**

It a single process contain multiple threads and each of thread has own stack and register. Code, data section and files belonging to these process are shared by these threads. Multi thread process much more efficient than single thread process it make computation faster and efficient.

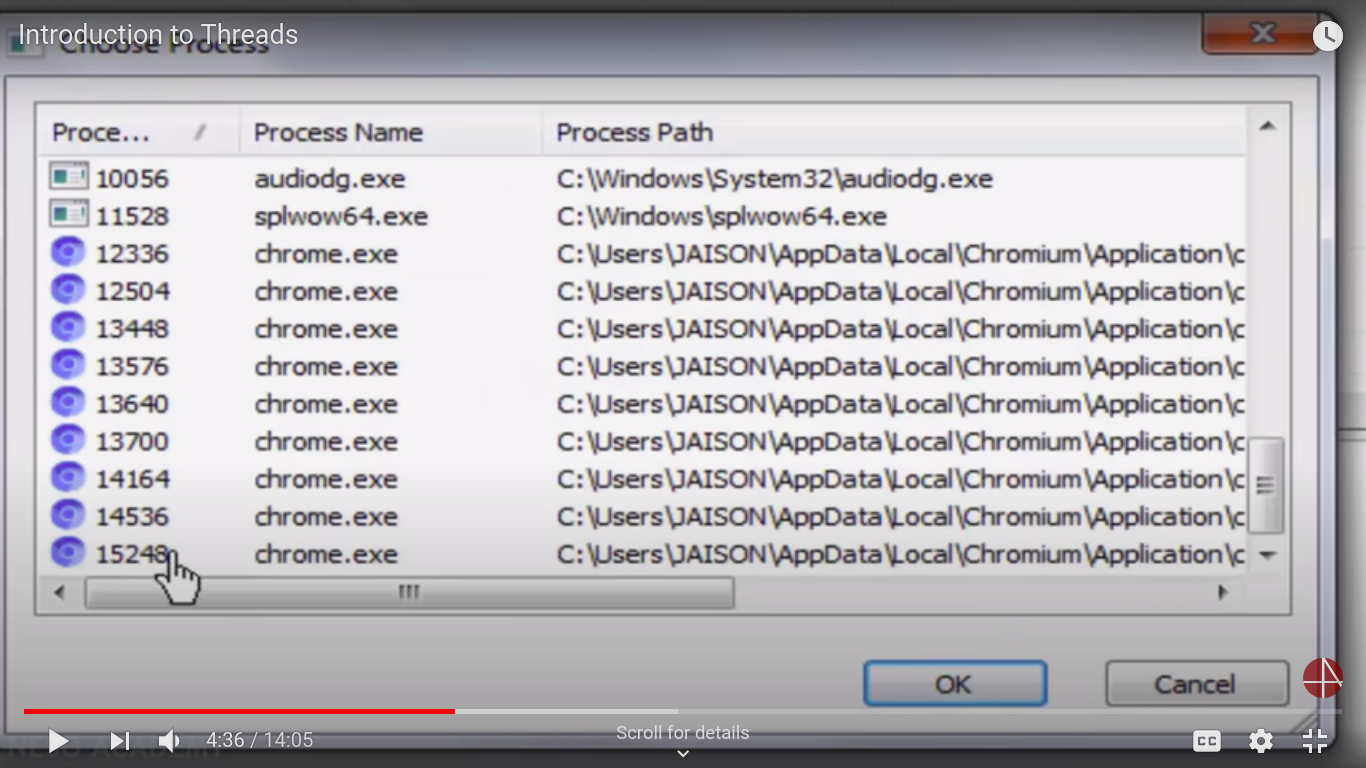
In a multi-threaded process, all of the process threads share the same memory and open files. Within the shared memory, each thread gets its own stack. Each thread has its own instruction pointer and registers. An operating system had to keep track of processes, and stored its per-process information in a data structure called a process control block (PCB).

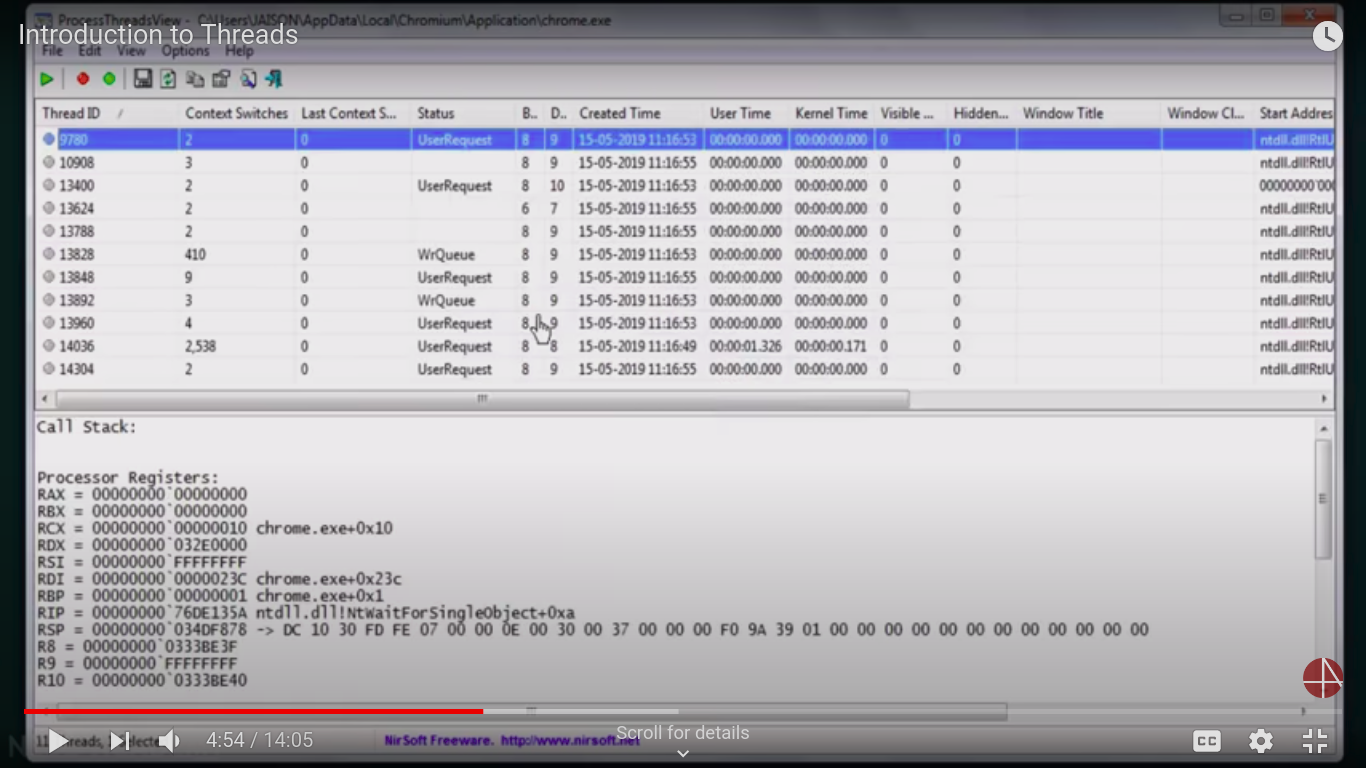
The items that the operating system must store that are unique to each thread are:

* Thread ID
* Saved registers, stack pointer, instruction pointer
* Stack (local variables, temporary variables, return addresses)
* Signal mask
* Priority (scheduling information)

The items that are shared among threads within a process are:

* Text segment (instructions)
* Data segment (static and global data)
* BSS segment (uninitialized data)
* Open file descriptors
* Signals
* Current working directory
* User and group IDs

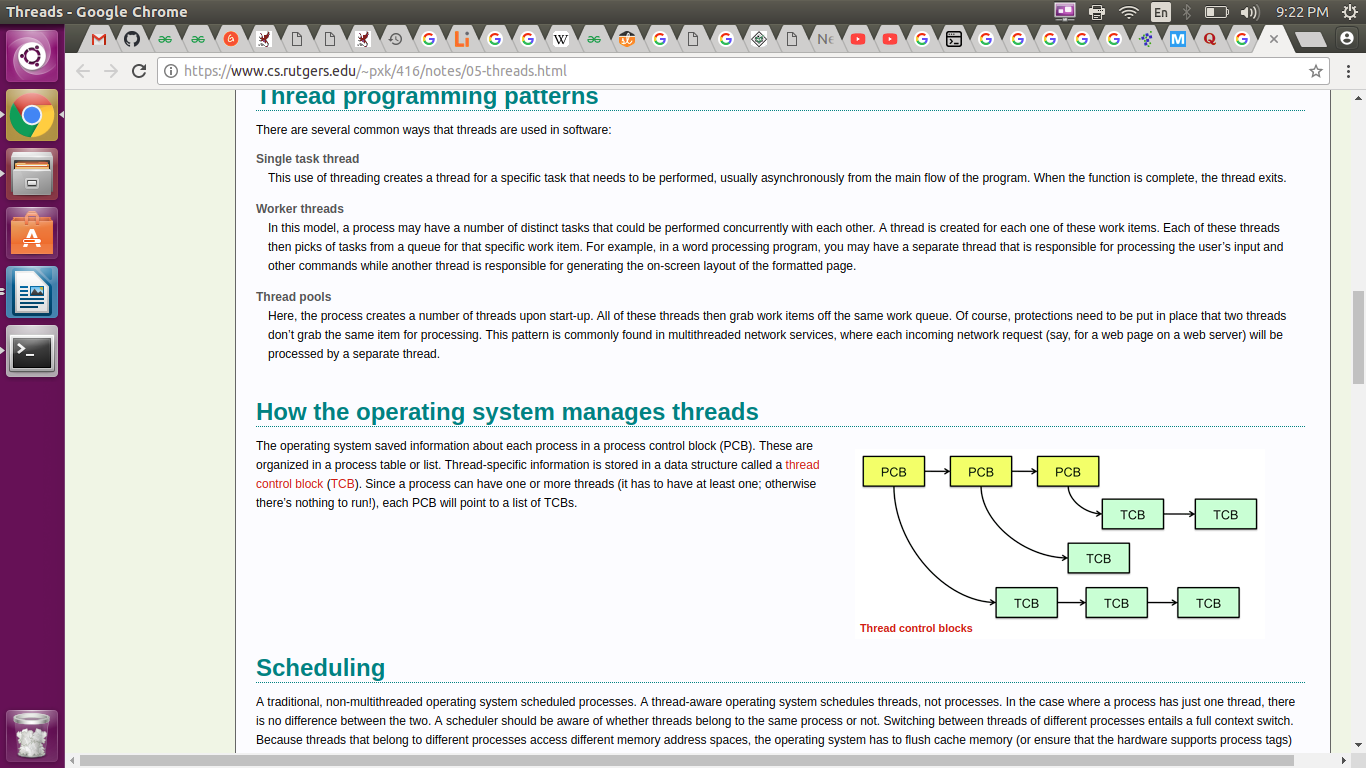
****

****

Ex: One thread using for display window and another thread download from internet.

# **How the operating system manages threads:**

The operating system saved information about each process in a process control block (PCB). These are organized in a process table or list. Thread-specific information is stored in a data structure called a thread control block(TCB), each PCB will point to a list of TCBs.



**Benefits of multi thread :**

**Resource Sharing:**

All the threads of a process share its resources such as memory, data, files etc. A single application can have different threads within the same address space using resource sharing.

**Responsiveness:**

Program responsiveness allows a program to run even if part of it is blocked using multithreading. This can also be done if the process is performing a lengthy operation. For example - A web browser with multithreading can use one thread for user contact and another for image loading at the same time.

**Utilization of Multiprocessor Architecture:**

In a multiprocessor architecture, each thread can run on a different processor in parallel using multithreading. This increases concurrency of the system. This is in direct contrast to a single processor system, where only one process or thread can run on a processor at a time.

**Economy:**

It is more economical to use threads as they share the process resources. Comparatively, it is more expensive and time-consuming to create processes as they require more memory and resources. The overhead for process creation and management is much higher than thread creation and management.