objects, polymorphism and inheritance.

Both processes and threads are independent sequences of execution. The typical difference is that threads (of the same process) run in a shared memory space, while processes run in separate memory spaces.

**Process**  
Each process provides the resources needed to execute a program. A process has a virtual address space, executable code, open handles to system objects, a security context, a unique process identifier, environment variables, a priority class, minimum and maximum working set sizes, and at least one thread of execution. Each process is started with a single thread, often called the primary thread, but can create additional threads from any of its threads.

**Thread**  
A thread is the entity within a process that can be scheduled for execution. All threads of a process share its virtual address space and system resources. In addition, each thread maintains exception handlers, a scheduling priority, thread local storage, a unique thread identifier, and a set of structures the system will use to save the thread context until it is scheduled. The thread context includes the thread's set of machine registers, the kernel stack, a thread environment block, and a user stack in the address space of the thread's process. Threads can also have their own security context, which can be used for impersonating clients.

**Process:**

* An executing instance of a program is called a process.
* Some operating systems use the term ‘task‘ to refer to a program that is being executed.
* A process is always stored in the main memory also termed as the primary memory or random access memory.
* Therefore, a process is termed as an active entity. It disappears if the machine is rebooted.
* Several process may be associated with a same program.
* On a multiprocessor system, multiple processes can be executed in parallel.
* On a uni-processor system, though true parallelism is not achieved, a process scheduling algorithm is applied and the processor is scheduled to execute each process one at a time yielding an illusion of concurrency.
* **Example:** Executing multiple instances of the ‘Calculator’ program. Each of the instances are termed as a process.

**Thread:**

* A thread is a subset of the process.
* It is termed as a ‘lightweight process’, since it is similar to a real process but executes within the context of a process and shares the same resources allotted to the process by the kernel.
* Usually, a process has only one thread of control – one set of machine instructions executing at a time.
* A process may also be made up of multiple threads of execution that execute instructions concurrently.
* Multiple threads of control can exploit the true parallelism possible on multiprocessor systems.
* On a uni-processor system, a thread scheduling algorithm is applied and the processor is scheduled to run each thread one at a time.
* All the threads running within a process share the same address space, file descriptors, stack and other process related attributes.
* Since the threads of a process share the same memory, synchronizing the access to the shared data withing the process gains unprecedented importance.

In [computing](https://en.wikipedia.org/wiki/Computing), **virtual memory** is a [memory management](https://en.wikipedia.org/wiki/Memory_management) technique that is implemented using both hardware and software. It maps [memory addresses](https://en.wikipedia.org/wiki/Memory_address) used by a program, called [*virtual addresses*](https://en.wikipedia.org/wiki/Virtual_address_space), into *physical addresses* in computer memory. [Main storage](https://en.wikipedia.org/wiki/Main_storage#Primary_storage) as seen by a process or task appears as a contiguous [address space](https://en.wikipedia.org/wiki/Address_space) or collection of contiguous [segments](https://en.wikipedia.org/wiki/Memory_segmentation). The operating system manages virtual address spaces and the assignment of real memory to virtual memory. Address translation hardware in the CPU, often referred to as a [memory management unit](https://en.wikipedia.org/wiki/Memory_management_unit) or *MMU*, automatically translates virtual addresses to physical addresses. Software within the operating system may extend these capabilities to provide a virtual address space that can exceed the capacity of real memory and thus reference more memory than is physically present in the computer.