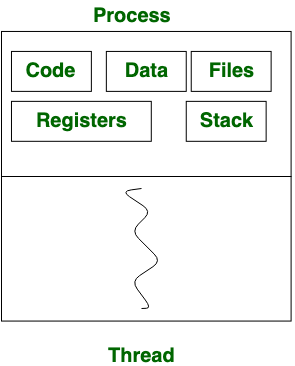
**Difference between Process and Thread**

**Thread:**Thread is the segment of a process which means a process can have multiple threads and these multiple threads are contained within a process. A thread has three states: Running, Ready, and Blocked.

The thread takes less time to terminate as compared to the process but unlike the process, threads do not isolate.



*Process vs Thread*

## **Difference between Process and Thread:**

| S.NO | Process | Thread |
| --- | --- | --- |
| 1. | Process means any program is in execution. | Thread means a segment of a process. |
| 2. | The process takes more time to terminate. | The thread takes less time to terminate. |
| 3. | It takes more time for creation. | It takes less time for creation. |
| 4. | It also takes more time for context switching. | It takes less time for context switching. |
| 5. | The process is less efficient in terms of communication. | Thread is more efficient in terms of communication. |
| 6. | Multiprogramming holds the concepts of multi-process. | We don’t need multi programs in action for multiple threads because a single process consists of multiple threads. |
| 7. | The process is isolated. | Threads share memory. |
| 8. | The process is called the heavyweight process. | A Thread is lightweight as each thread in a process shares code, data, and resources. |
| 9. | Process switching uses an interface in an operating system. | Thread switching does not require calling an operating system and causes an interrupt to the kernel. |
| 10. | If one process is blocked then it will not affect the execution of other processes | If a user-level thread is blocked, then all other user-level threads are blocked. |
| 11. | The process has its own Process Control Block, Stack, and Address Space. | Thread has Parents’ PCB, its own Thread Control Block, and Stack and common Address space. |
| 12. | Changes to the parent process do not affect child processes. | Since all threads of the same process share address space and other resources so any changes to the main thread may affect the behavior of the other threads of the process. |
| 13. | A system call is involved in it. | No system call is involved, it is created using APIs. |
| 14. | The process does not share data with each other. | Threads share data with each other. |

Note: *In*s*ome cases where the thread is processing a bigger workload compared to a process’s workload then the thread may take more time to terminate. But this is an extremely rare situation and has fewer chances to occur.*