What is a Thread?
A thread is a path of execution within a process. A process can contain multiple threads.

Process vs Thread?
The primary difference is that threads within the same process run in a shared memory space, while processes run in separate memory spaces.
Threads are not independent of one another like processes are, and as a result threads share with other threads their code section, data section, and OS resources (like open files and signals). But, like process, a thread has its own program counter (PC), register set, and stack space.

Why Multithreading?
A thread is also known as a lightweight process. The idea is to achieve parallelism by dividing a process into multiple threads. For example, in a browser, multiple tabs can be different threads. MS Word uses multiple threads: one thread to format the text, another thread to process inputs, etc. More advantages of multithreading are discussed below

What is Process Scheduling?
The process scheduling is the activity of the process manager that handles the removal of the running process from the CPU and the selection of another process on the basis of a particular strategy. Process scheduling is an essential part of a Multiprogramming operating system.

What is Arrival Time?
Time at which the process arrives in the ready queue.

What is Burst Time?
Time required by a process for CPU execution.

What is Completion Time?
Time at which process completes its execution.

What is Turn Around Time?
Turn Around Time is the total time taken between the submission of a program/process/thread/task for execution and the return of the complete output to the customer/user.
Formula: TAT = CT - AT

What is Waiting Time?
The time spent by a process waiting in the ready queue for getting the CPU. The time difference b/w Turnaround Time and Burst Time is called Waiting Time.
Formula: WT = TAT - BT

What is Response Time?
Response Time The difference between the arrival time and the time at which the process first gets the CPU is called Response Time.

What is an Operating System?
An operating system (OS) is system software that manages computer hardware, software resources, and provides common services such as recognizing input from the keyboard, sending output to the display screen, keeping track of files and directories on the disk, and controlling peripheral devices such as disk drives and printers.
for computer programs.

What is OS used for?

- In a multitasking operating system where multiple programs can be running at the same time, the operating system determines which applications should run in what order and how much time should be allowed for each application before giving another application a turn.
- It manages the sharing of internal memory among multiple applications.
- It handles input and output to and from attached hardware devices, such as hard disks, printers, and dial-up ports.
- It sends messages to each application or interactive user (or to a system operator) about the status of operation and any errors that may have occurred.
- It can offload the management of what are called batch jobs (for example, printing) so that the initiating application is freed from this work.
- On computers that can provide parallel processing, an operating system can manage how to divide the program so that it runs on more than one processor at a time.

What is a Kernel?
A Kernel is a computer program that is the heart and core of an Operating System. Since the Operating System has control over the system so, the Kernel also has control over everything in the system. It is the most important part of an Operating System.

What are objectives of Kernel :

- To establish communication between user level application and hardware.
- To decide state of incoming processes.

- To control disk management.
- To control memory management.
- To control task management.

What is First Come First Serve?

First-Come-First-Serve algorithm is the simplest scheduling algorithm. It works on the principle of FIFO - First In First Out Processes are dispatched according to their arrival time on the ready queue. Being a non-pre-emptive discipline, once a process has a CPU, it runs to completion. FCFS scheme is not useful in scheduling interactive users because it cannot guarantee good response time. One of the major drawbacks of this scheme is that the average time is often quite long.

What is Shortest Job First?

Shortest-Job-First (SJF) is a non-pre-emptive discipline in which waiting job (or process) with the smallest estimated run-time-to-completion is run next. In other words, when CPU is available, it is assigned to the process that has smallest next CPU.
The SJF scheduling is especially appropriate for batch jobs for which the run times are known in advance. The SJF algorithm favours short jobs at the expense of longer ones.
The problem with the SJF scheme is that it requires precise knowledge of how long a job or process will run, and this information is not usually available.

What is Shortest Time Remaining First?

The preemptive version of SJF is known as the Shortest Remaining Time First (SRTF) Algorithm. It is useful when a multiprogramming environment is used. SRTF is more advantageous because the CPU utilization is more efficient as compared to SJF. Preemptive scheduling is slightly more flexible as compared to SJF. Also, the waiting and response time of SRTF is less as compared to SJF.
The disadvantages of SRT are that it takes slightly longer to suspend a running task and low priority tasks have to wait for a longer time if higher priority tasks arrive continuously.

How does priority work?

Priority can be defined either internally or externally. Internally defined priorities use some measurable quantities or qualities to compute priority of a process.
Examples of Internal priorities are:

- Time limits.
- Memory requirements.
- File requirements, for example, number of open files.
- CPU Vs I/O requirements.

Externally defined priorities are set by criteria that are external to operating system such as:

- The importance of process.

- Type or amount of funds being paid for computer use.
- The department sponsoring the work.
- Politics.

What is Preemptive and Non Preemptive scheduling?
In Preemptive Priority Scheduling, at the time of arrival of a process in the ready queue, its Priority is compared with the priority of the other processes present in the ready queue as well as with the one which is being executed by the CPU at that point of time. The One with the highest priority among all the available processes will be given the CPU next.

In the Non Preemptive Priority scheduling, The Processes are scheduled according to the priority number assigned to them. Once the process gets scheduled, it will run till the completion. Generally, the lower the priority number, the higher is the priority of the process.

What is the problem with priority scheduling?
A major problem with priority scheduling is indefinite blocking or starvation. A solution to the problem of indefinite blockage of the low-priority process is *aging*. Aging is a technique of gradually increasing the priority of processes that wait in the system for a long period of time.

Which algorithm out of non preemptive and preemptive is more advantageous?

CPU utilization is more efficient compared to Non-Preemptive Scheduling.	CPU utilization is less efficient compared to preemptive Scheduling.
Waiting and response time of preemptive Scheduling is less.	Waiting and response time of the non-preemptive Scheduling method is higher.
Preemptive Scheduling is prioritized. The highest priority process is a process that is currently utilized.	When any process enters the state of running, the state of that process is never deleted from the scheduler until it finishes its job.

What is program?
When we execute a program that was just compiled, the OS will generate a process to execute the program. Execution of the program starts via GUI mouse clicks, command line entry of its name, etc. A program is a passive entity as it resides in the secondary memory, such as the contents of a file stored on disk. One program can have several processes.

What is process?
The term process (Job) refers to program code that has been loaded into a computer's memory so that it can be executed by the central processing unit (CPU). A process can be described as an instance of a program running on a computer or as an entity that can be assigned to and executed on a processor. A program becomes a process when loaded into memory and thus is an active entity.

What is Short Term Scheduling?
Short-term scheduling involves selecting one of the processes from the ready queue and scheduling them for execution. This is done by the short-term scheduler. A scheduling algorithm is used to decide which process will be scheduled for execution next by the short-term scheduler.

What is Mid Term Scheduling?
Medium-term scheduling involves swapping out a process from main memory. The process can be swapped in later from the point it stopped executing. This can also be called as suspending and resuming the process and is done by the medium-term scheduler.

What is Long Term Scheduling?
Long-term scheduling involves selecting the processes from the storage pool in the secondary memory and loading them into the ready queue in the main memory for execution. This is handled by the long-term scheduler or job scheduler.

What are the number of states?
1. New: A program which is going to be picked up by the OS into the main memory is called a new process.
2. Ready: Whenever a process is created, it directly enters in the ready state, in which, it waits for the CPU to be assigned. The OS picks the new processes from the secondary memory and put all of them in the main memory. The processes which are ready for the execution and reside in the main memory are called ready state processes. There can be many processes present in the ready state.
3. Running: One of the processes from the ready state will be chosen by the OS depending upon the scheduling algorithm. Hence, if we have only one CPU in our system, the number of running processes for a particular time will always be one. If we have n processors in the system then we can have n processes running simultaneously.

4. Block or Wait: From the Running state, a process can make the transition to the block or wait state depending upon the scheduling algorithm or the intrinsic behavior of the process. When a process waits for a certain resource to be assigned or for the input from the user then the OS moves this process to the block or wait state and assigns the CPU to the other processes.
5. Completion or Termination: When a process finishes its execution, it comes in the termination state. All the context of the process (Process Control Block) will also be deleted and the process will be terminated by the Operating system.

What was there before OS?/What would you do if OS was not there?

Each user had sole use of the machine for a scheduled period of time and would arrive at the computer with a program and data, often on punched paper cards and magnetic or paper tape. The program would be loaded into the machine and the machine would work until the program was complete or crashed. Programs could generally be debugged via a control panel using toggle switches and panel lights.

Where does the process arrive before it arrives in the CPU?
It arrives in the ready state.

Why does process blocking occur?
From the Running state, a process can make the transition to the block or wait state depending upon the scheduling algorithm or the intrinsic behavior of the process. When a process waits for a certain resource to be assigned or for the input from the user then the OS moves this process to the block or wait state and assigns the CPU to the other processes.

Can a thread share 2 different processes?
Yes

What are the essential elements of a process?
Attributes of a process The Attributes of the process are used by the Operating System to create the process control block (PCB) for each of them. This is also called context of the process. Attributes which are stored in the PCB are described below.
1. Process ID: When a process is created, a unique id is assigned to the process which is used for unique identification of the process in the system.
2. Program counter: A program counter stores the address of the last instruction of the process on which the process was suspended. The CPU uses this address when the execution of this process is resumed.
3. Process State: The Process, from its creation to the completion, goes through various states which are new, ready, running and waiting.
4. Priority: Every process has its own priority. The process with the highest priority among the processes gets the CPU first. This is also stored on the process control block.

5. General Purpose Registers: Every process has its own set of registers which are used to hold the data which is generated during the execution of the process.
6. List of open files: During the Execution, Every process uses some files which need to be present in the main memory. OS also maintains a list of open files in the PCB.
7. List of open devices: OS also maintain the list of all open devices which are used during the execution of the process.

Difference between linux and windows

Linux and Windows both are operating systems. Linux is open source and is free to use whereas Windows is a proprietary. Following are the important differences between Linux and Windows.

Sr. No.	Key	Linux	Windows
1	Open Source	Linux is Open Source and is free to use.	Windows is not open source and is not free to use.
2	Case sensitivity	Linux file system is case sensitive.	Windows file system is case insensitive.
3	kernel type	Linux uses monolithic kernel.	Windows uses micro kernel.
4	Efficiency	Linux is more efficient in operations as compared to Windows.	Windows is less efficient in operations.
5	Path Seperator	Linux uses forward slash as path seperator between directorioes.	Windows uses backward slash as a path seperator.
6	Security	Linux is highly secure as compared to Windows.	Windows provides less security as compared to Linux.