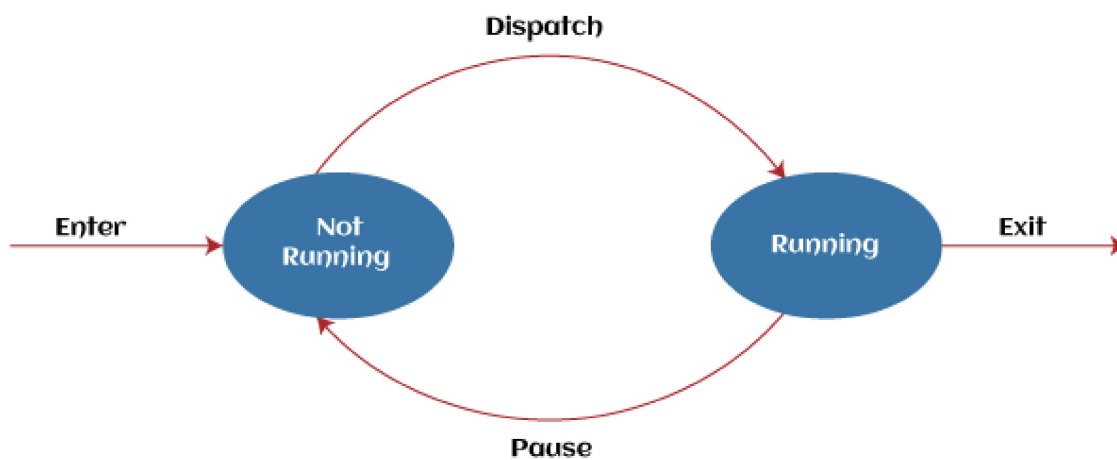


Home	Operating System	C	Java	PHP	HTML	CSS	Bootstrap	JavaScript	jQuery
------	------------------	---	------	-----	------	-----	-----------	------------	--------

Two State Process Model in Operating System

The OS process's passes from many states, beginning from its formation to its completion. A process is made up of program data and its associated data and a PCB. A process can change its state due to the following events such as I/O requests, synchronization of processes, interrupt routines, process scheduling algorithms, etc. The process can run or cannot run, and if the process is running, it has to be supported by the systems for proper progress of the process.

Two-State Process Model

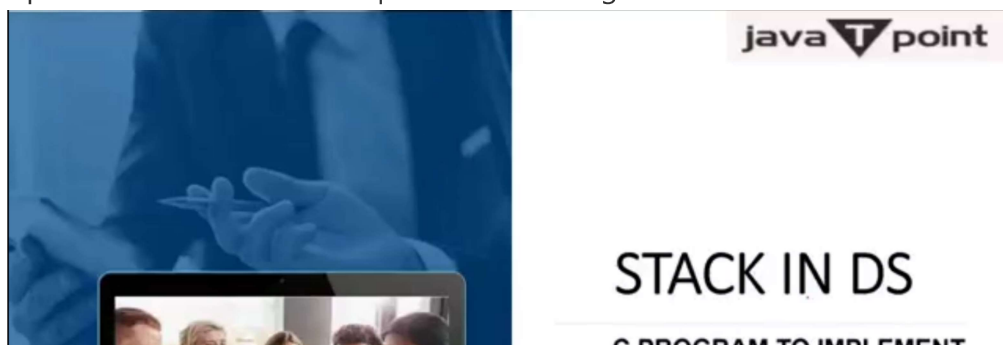


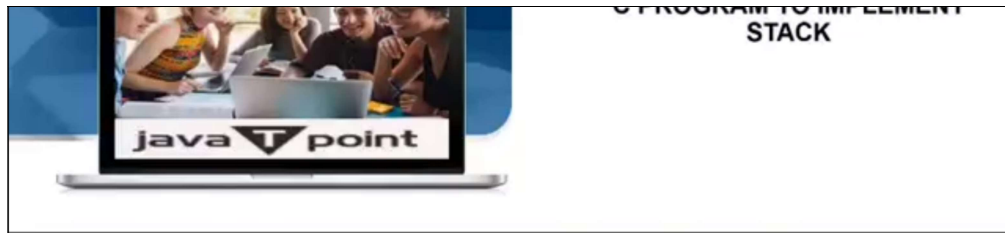
Two State Process Model

It is the simplest process model in the process state model because it has only two states. These states are as follows:

1. Running State

It is the current process state in which the process is running.





2. Not Running State

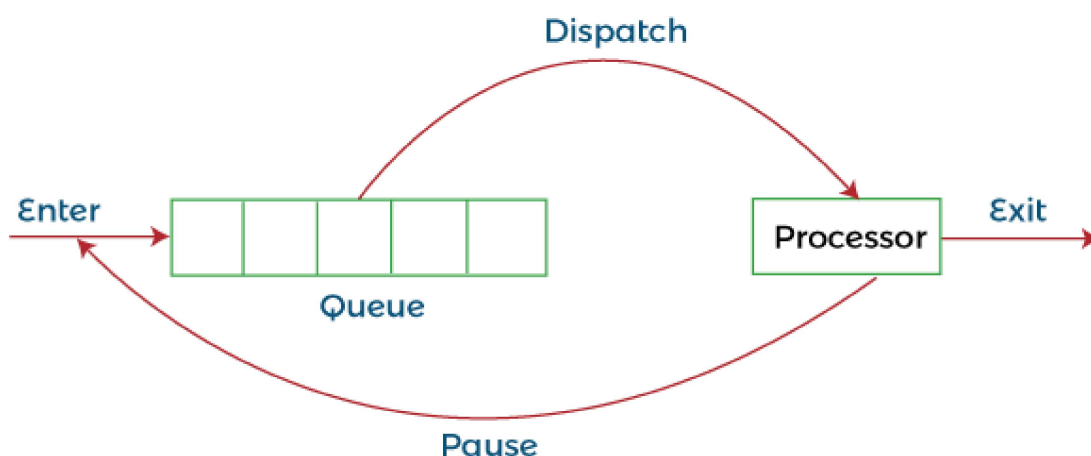
It is the state of a process that is waiting for running.

Execution of Two-State Process Model

This process state may be created anytime, whether a process is running or not.

1. Firstly, when the operating system starts a new process, it also constructs a PCB for it so that it may enter in the system in a non-running condition. When a process exits or leaves the system, the OS is aware about it.
2. The currently running process would be interrupted or broken into from time to time, and the operating system will execute any other process.
3. Now, the former process transitions from the running to the non-running state, while one of the other processes transitions to the running state before exiting the system.


Processes that are not currently running must be placed in a queue and wait for their chance to execute. There is only a single queue, and the entry is a pointer to a PCB. It has a block that stores information in a data structure such as context data, state, program counter, identifier, etc.





Queuing Diagram

You must be aware that a queue may contain linked list blocks representing a single process. If a process is interrupted, it is shifted to the queue waiting process, and if the process is finished its execution, it is terminated. The dispatcher then chooses another process from the queue and runs it. The dispatcher's behavior is shown in the above queuing diagram.

[< Prev](#)[Next >](#)

**100+**
Companies are looking
for Trained **Candidates**
from **CloudxLab**

Get Hired by Top Companies




Data Science Courses @ 50% Off

Unlimited access to 100+ courses, hands-on projects, get job-ready certificate programs,

Cloudxlab - Get Certified

[Apply Now >](#)

For Videos Join Our Youtube Channel: [Join Now](#)

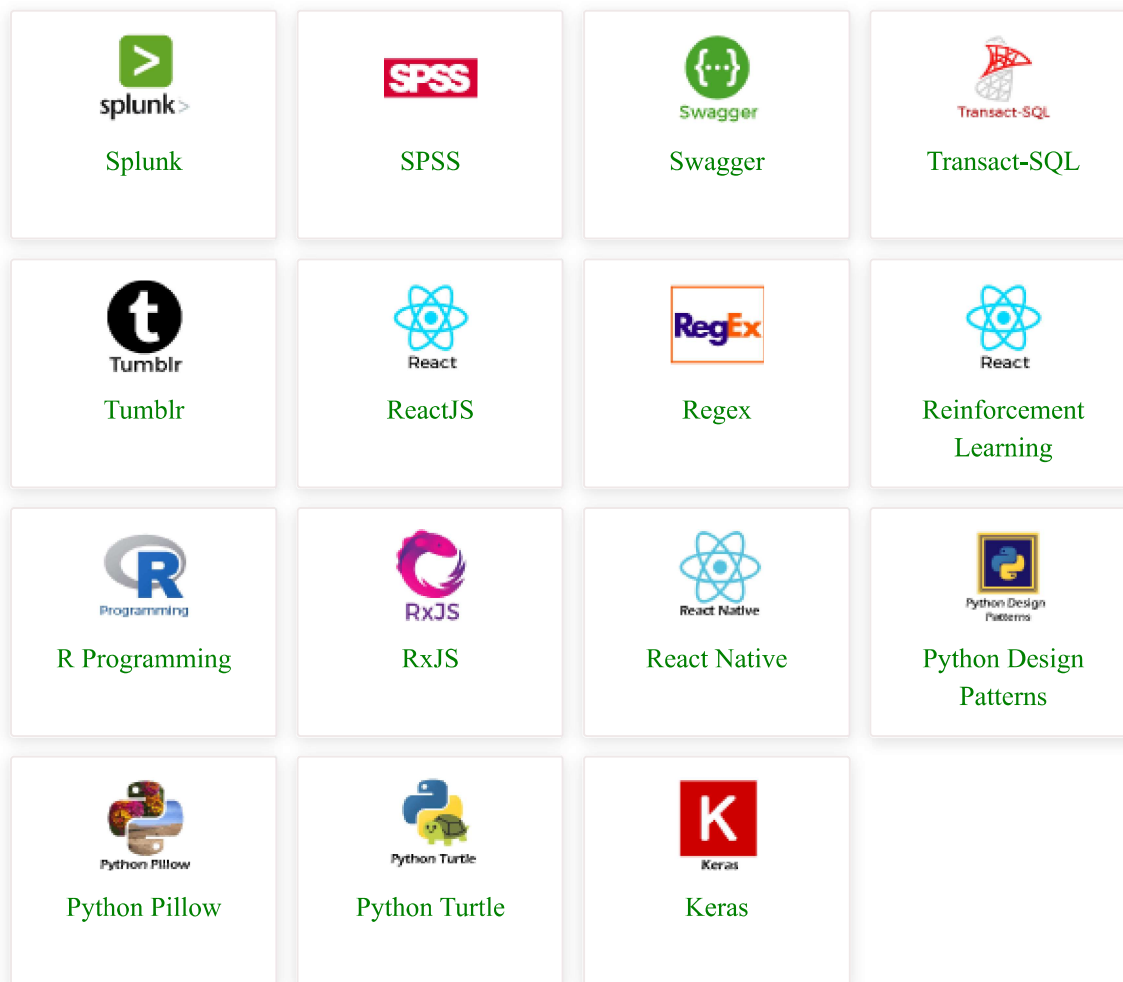
Feedback

- Send your Feedback to feedback@jvatpoint.com

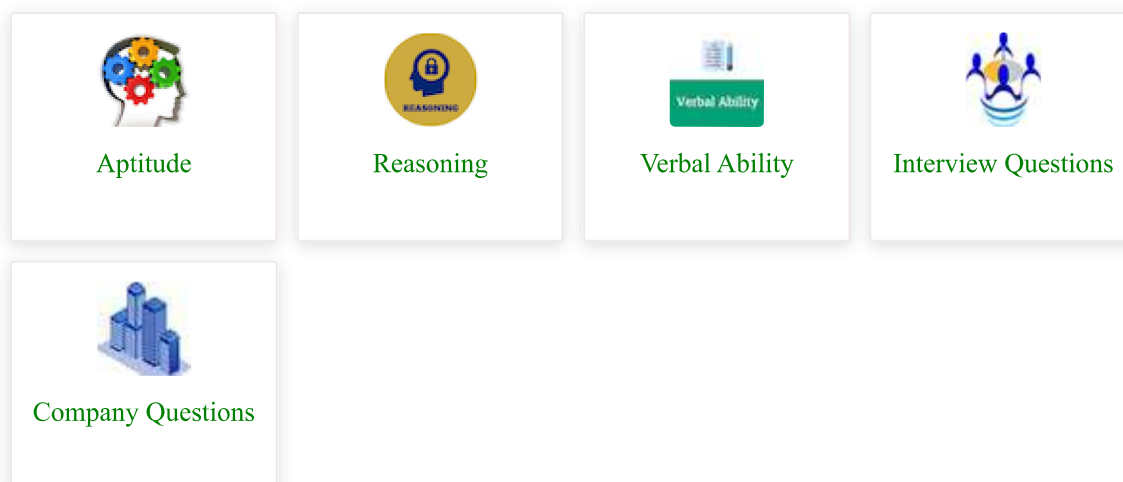
Help Others, Please Share



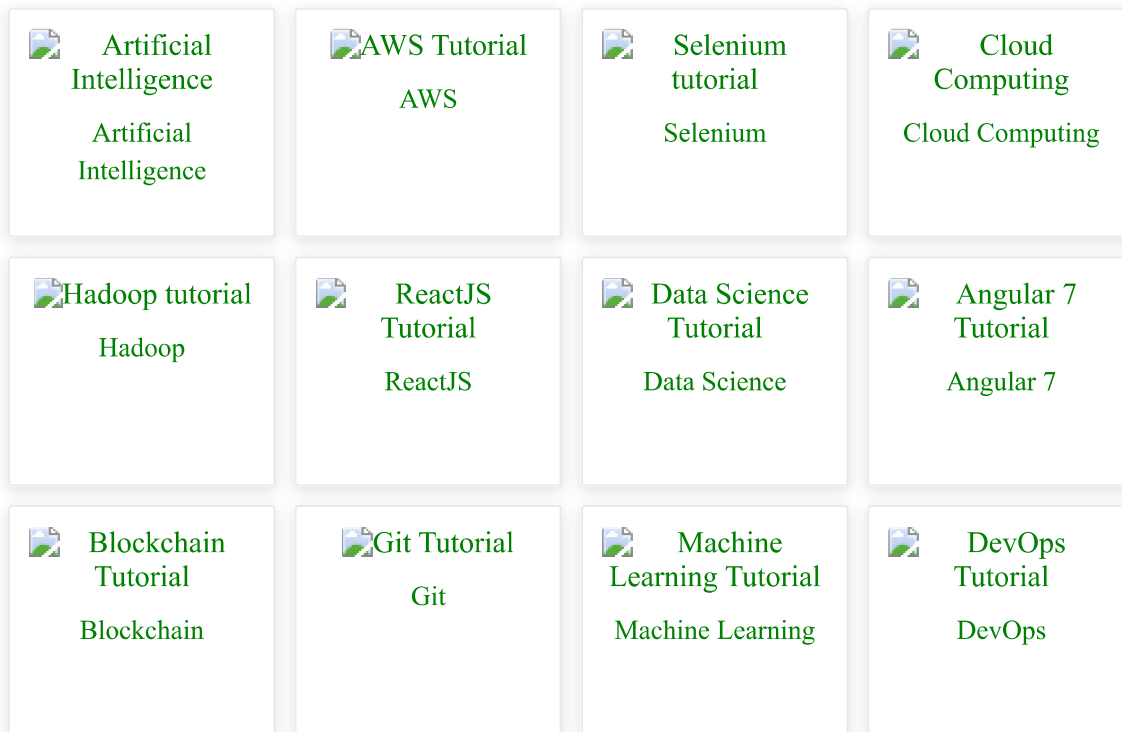
Learn Latest Tutorials



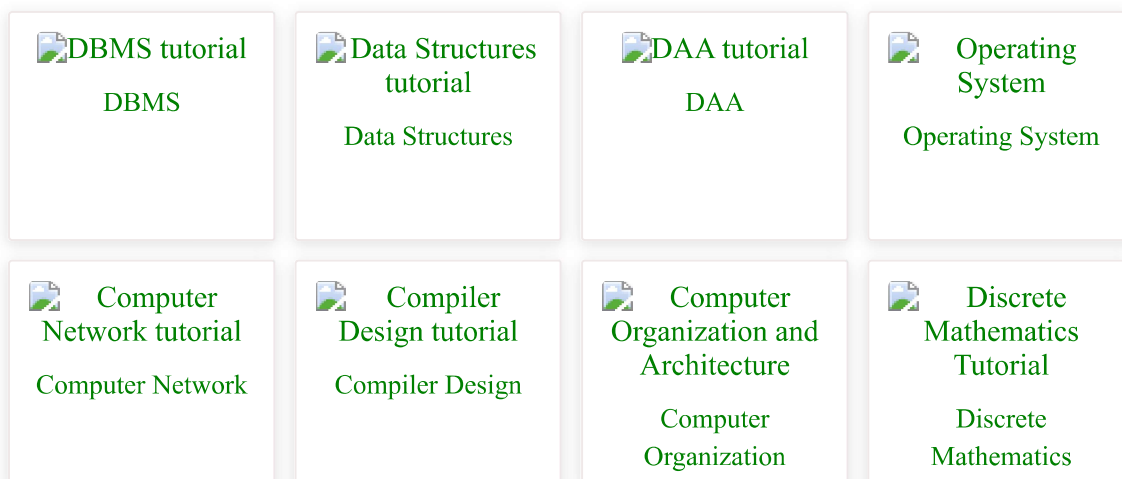
Preparation




Trending Technologies




B.Tech / MCA







Ethical Hacking
Ethical Hacking



Computer
Graphics Tutorial
Computer Graphics



Software
Engineering
Software
Engineering




html tutorial
Web Technology




Cyber Security
tutorial
Cyber Security




Automata
Tutorial
Automata



C Language
tutorial
C Programming




C++ tutorial
C++




Java tutorial
Java




.Net
Framework
tutorial
.Net



Python tutorial
Python



List of
Programs
Programs



Control
Systems tutorial
Control System



Data Mining
Tutorial
Data Mining



Data
Warehouse
Tutorial
Data Warehouse