Ansible Vs Shell Scripting

Feature	Ansible	Shell Scripting
Purpose	Configuration Management and Automation	Automation and Task Management
Language	YAML	Shell Scripting Languages (bash, sh, etc.)
Platform support	Cross-platform	Depends on the underlying shells
Agent requirement	Not required	Not required but can be used if needed
Remote execution support	Yes, using SSH	Yes, using SSH
Idempotency*	Built-in	Can be achieved with extra effort
Parallelism	Built-in	Can be achieved with extra effort
Error handling	Built-in	Can be achieved with extra effort
Configuration files	YAML files	Text files
Learning curve	Steep (Relatively hard)	Relatively easy
Community support	Strong	Strong

*Idempotent: An action which, when performed multiple times, has no further effect on its subject after the first time it is performed. In simple words, it means that if any module runs the first time, then it should not run again. This will save our CPU time. So, we need to have every module idempotent in nature.

In summary, Ansible is a powerful automation and configuration management tool that is cross-platform, idempotent, and built-in with many features that are useful for complex tasks. However, it has a steep learning curve and requires some knowledge of YAML. On the other hand, Shell scripting is a more general-purpose automation tool that is relatively easy to learn and has been around for a long time. It can be used for a wide variety of tasks and is flexible but requires more effort to achieve features like idempotency, parallelism, and error handling.