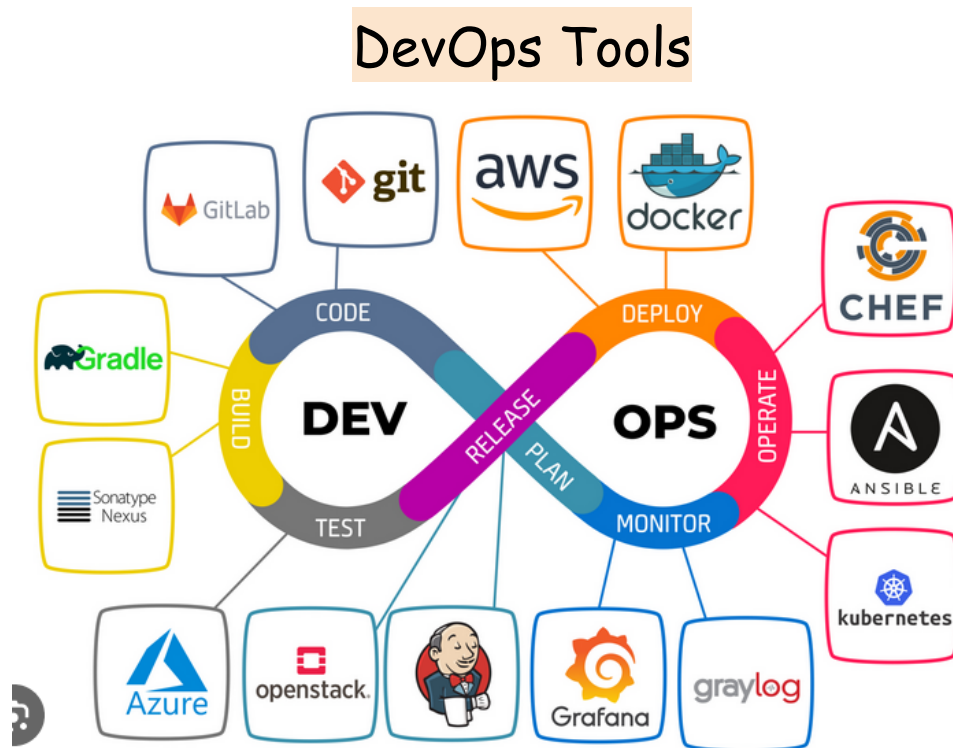


Agile vs DevOps

No.	Agile	DevOps
1	Agile is a method for creating software.	It is not related to software development. Instead, the software that is used by DevOps is pre-built, dependable, and simple to deploy.
2	The agile handle centers on consistent changes.	DevOps centers on steady testing and conveyance.
3	A few of the finest steps embraced in Agile are recorded underneath: 1. Backlog Building 2.Sprint advancement	DevOps to have a few best hones that ease the method : 1. Focus on specialized greatness. 2. Collaborate straightforwardly with clients and join their feedback.
4	Agile relates generally to the way advancement is carried of, any division of the company can be spry in its hones. This may be accomplished through preparation.	DevOps centers more on program arrangement choosing the foremost dependable and most secure course.
5	All the group individuals working in a spry hone have a wide assortment of comparable ability sets. This is often one of the points of interest of having such a group since within the time of requirement any of the group	DevOps features a diverse approach and is very viable, most of the time it takes after "Divide and Conquer". Work partitioned among the improvement and operation groups.

	individuals can loan help instead of holding up for the group leads or any pro impedances.	
6	Spry accepts "smaller and concise". Littler the group superior it would be to convey with fewer complexities.	DevOps, on the other hand, accepts that "bigger is better".
7	Since Agile groups are brief, a foreordained sum of time is there which are sprints. Tough, it happens that a sprint has endured longer than a month but regularly a week long.	DevOps, on the other hand, prioritizes reliabilities. It is since of this behavior that they can center on a long-term plan that minimizes commerce's unsettling influences.
8	A big team for your project is not required.	It demands collaboration among different teams for the completion of work.
9	Some of the Tools: <ul style="list-style-type: none"> • Bugzilla • JIRA • Kanboard and more. 	Some of the Tools: <ul style="list-style-type: none"> • Puppet • Ansible • AWS • Chef • team City OpenStack and more.
10	It is suitable for managing complex projects in any department.	It centers on the complete engineering process.
11	It does not focus on automation.	It focusses on automation.
12	Working system gets more significance in Agile than	The process documentation is significant in DevOps.

	documentation.	
--	----------------	--



No.	Tool	Example
1	Version Control Tool	<p>Git: is perhaps the best and most widely used version control tool in a development era characterized by dynamism and collaboration. Version control provides developers with a means by which they can keep track of all the changes and updates in their codes such that in the event of a mishap, it is quite easy to return to and use the previous versions of the code and Git happens to be the best for many reasons.</p> <ul style="list-style-type: none"> - Git features three storage tools including, GitHub and GitLab cloud-hosted code repository services as well as BitBucket

		the source code hosting service.
2	Build Tool	<p>Maven: is one of the important DevOps tools for building projects. Apache Maven is more than just an automation build framework. It is also designed to manage reporting, documentation, distribution, releases, and dependencies processes. Written in Java language, Maven can build and manage projects written in Java or C#, Ruby, Scala, and other languages.</p> <ul style="list-style-type: none"> - Maven offers a host of benefits to its users. It eases the build and monitoring process through automation and maintains a uniform build process allowing for consistency and efficiency. - It has a rich repository of plugins to enhance the build process and wide compatibility with IDEs like Eclipse, JBuilder, MyEclipse, NetBeans, IntelliJ IDEA, and others.
3	Continuous Integration Tool	<p>Jenkins: is an integration DevOps tool. For continuous integration (CI), Jenkins is an open-source Java-based automation CI server that is supported by multiple operating systems including Windows, macOS, and other Unix OSs. Jenkins can also be deployed on cloud-based platforms.</p> <ul style="list-style-type: none"> - A valuable automation CI tool, Jenkins is pretty easy to install and configure. It is designed to support distributed workflows for accelerated and transparent builds, tests, and deployments across platforms.

4	Configuration Management Tool	<p>Chef, Puppet, and Ansible are handy Configuration management automation frameworks. While Chef and Puppet are Ruby-based frameworks, Ansible is a Python-based framework.</p> <ul style="list-style-type: none"> - Configuration management (CM) refers to the maintenance and control of the components of large complex systems in a known, consistent, and determined state throughout the DevOps life cycle. Components of an IT system may include servers, networks, storage, and applications. - For this reason, configuration management is critical to any system as it is the process by which changes in the system are tracked, properly implemented, and controlled. Further, if not automated, CM can be laborious, resource-draining, and prone to costly errors. It implements configuration tools for such repetitive administrative tasks as version management, regulatory compliance, feature releases, and processes automation, among others.
5	Container Platforms	<p>Docker and Kubernetes as the most widely used container technologies.</p> <ul style="list-style-type: none"> - Container platforms are application solutions that allow developers to build, test, and ship applications in resource-independent environments. Each container comprises a complete runtime environment including the specific

		<p>application, its libraries, source code, configurations, and all its dependencies. Container platforms offer orchestration, automation, security, governance, and other capabilities.</p> <p>Docker: is open-source and compatible with cloud services like AWS, GCP, and Azure Cloud. Docker also runs on Windows and Linux operating systems.</p>
10	Communication and Collaboration	<p>Slack: One of the most popular communication and collaboration tools.</p> <ul style="list-style-type: none"> - Workplace communication and collaboration technologies are as numerous and as diverse as can be imagined. And when it comes to deciding which tools best suit specific business requirements, several factors go into consideration such as integration and automation capabilities, security, user experience, as well as whether to develop, buy or rent. <p>Slack:</p> <ul style="list-style-type: none"> • Powerful search capabilities with well-designed search modifiers to ease document tracking, management, and file sharing. • A friendly project management architecture integrates with project management tools like Twitter, Google Hangouts, Trello, and more. • Powerful collaboration and communication capabilities via shared channels, direct

		chat, voice, and video conferencing.
--	--	--------------------------------------