

DevOps Assignment 2

Q1:- Why is DevOps a major requirement in today's scenario?

⇒ In a nutshell the DevOps model allows companies to create viable application and programmes within a much shorter time frame, thus accelerating the speed of innovation. It is a major requirement because it is a software development and operational approach that enables faster developments of new products & easy maintenance of existing deployments.

Some important benefits of DevOps as a major requirement:

- * faster solution
- * Increased efficiency
- * Improved customer experience
- * Faster ROI
- * Improved performance
- * Continuous Improvements
- * Reduce failures & Roll Back

→ Greater stability of IT Software applications. As it brings various departments such as IT, product, engineering, Cybersecurity, Operations & more & unites them in common objectives of achieving business targets.

→ In this approach the software is seen as a tool to improve organizational efficiency & security by automating several key processes.

Name \Rightarrow Abhilash Bharti

Section \Rightarrow F

Univ. Roll No \Rightarrow 2018007.

Date / /

Page No. 2

Q2. Explain all DevOps tools in detail.

\Rightarrow Since no single tool work across all areas of developments & delivery. The need is to first understand the process and accordingly map the tool to be successfully establish DevOps culture in Organization.

i> Jenkins \Rightarrow An excellent DevOps automation tool being adopted by increased number of software development teams, it is essentially an open source CI/CD server that helps in automating the different stages of delivering pipeline.

- ★ Allows us to setup and customise CD pipeline as per individual needs
- ★ Runs on Linux, Windows & Mac OS.
- ★ Jenkins allows you to iterate & display new code with greater speed.

ii> Git \Rightarrow Widely used across software industries; Git is an distributed SCM (Source Code Management) DevOps tool. It allows you to easily track the process of your development work, where you can save different versions of source code return to previous one as when required.

- ★ A free & open-source tool that supports most of the version control features of check-in, merging, labels, commit, branches etc.

Name \Rightarrow Abhilash Bharti

Section \Rightarrow F

Univ. Roll No. \Rightarrow 2018007

Date / /

Page No. 3

★ Requires a hosted repository such as github or bitbucket that offers unlimited private repositories.

★ Easy to learn & maintain with separate branches of source-code that can be merged through git.

iii> Nagios \Rightarrow One of the most popular free & open-source DevOps monitoring tool; Nagios allows you to monitor your infrastructure real-time so that identifying security threats, detection of outages & error becomes easier.

★ Facilitates two methods of server monitoring \rightarrow agent based or agentless.

★ Allow for monitoring of Windows, UNIX, Linux, & web apps as well.

★ free open source with various add-ons available.

different versions of Nagios are:-

- 1) Nagios Core \rightarrow Command Line Tool
- 2) Nagios XI \rightarrow Web based GUI
- 3) Log Server \rightarrow Searches log data with automatic alerts.
- 4) Nagios Fusion \rightarrow for simultaneous multiple network monitoring.

Name \Rightarrow Abhilash Bharti

Section \Rightarrow F

Univ. Roll No \Rightarrow 2018007

Date / /

Page No. 4

iv) Docker \Rightarrow It is one of the widely used development tool of DevOps & is known to provide platform independent integrated contains security & agile operations for cloud native & legacy applications.

- ★ Easily automates apps deployment & make distributed development easy.

- ★ Docker container support virtual machine environments & platform independent.

- ★ Build-in support for docker available for both Google Cloud & AWS.

v) Kubernetes \Rightarrow ideal for large teams, this DevOps tool is built on what docker started in the field of containerization. It is a powerful tool that can group containers by logical categorization.

- ★ It can be deployed to multiple computers through automated distribution.

- ★ Kubernetes is the first container Orchestration tool.

- ★ Extremely useful in the streamlining complex projects across large teams.

vi) Ansible \Rightarrow It is primarily a design management and organization DevOps tool. It is written in simple programming language YAML. It makes easier for DevOps teams to scale the process of automation & speed up productivity.

Name ⇒ Abhilash Bharti

Section ⇒ F

Univ. Roll No ⇒ 2018007

Date

Page No. 5

- ★ Based on master slave architecture.
- ★ It is an ideal DevOps tool to manage complex deployments & speed up the process of development.

vii> Chef ⇒ This DevOps tool is mainly used for checking the configurations & it is helpful in automating the infrastructure.

- ★ Assist in standardizing and enforcing the configurations continuously.
- ★ Chef automates the whole process and make sure that the systems are correctly configured.
- ★ Chef helps you ensure that the configuration policies remain completely flexible, readable & testable.

viii> Puppet ⇒ It is an open source configuration management tool that is used for deploying, configuring & managing servers.

- ★ Offers master-slave architecture.
- ★ Puppet works smoothly for hybrid infrastructure & applications.
- ★ Compatible with Linux, Windows & Unix OS.

(ix) Splunk ⇒ It is designed to make machine data usable as well as accessible to everyone by delivering operational intelligence to DevOps teams.

Name \Rightarrow Abhilash Bharti

Section \Rightarrow F

Univ. Roll No \Rightarrow 2018007

Date / /

Page No. 6

It is an excellent choice of tool that make companies more secure, productive & competitive.

* Splunk delivers a more central and collective view of IT services.

* Easily detects ~~a more central~~ patterns, highlights, anomalies and areas of impact.

x> Gradle \Rightarrow An extremely versatile DevOps tool, Gradle allows you to write your code in various languages including C++, Java, Python among others.

It is supported by various IDE's such as net beans, Eclipse & IntelliJ IDEA.

* The core model of gradle is based on tasks-action, inputs & outputs.

* The incremental builds of Gradle allows you to save a substantial amount of compile time.