DevOps is Development and Operations.

Why devops?

As DevOps is collaboration of development and operations. Before evolution of DevOps there are two teams working separately there is no proper communication between them due to this there is a delay in production and deployment. DevOps bridge the gap between the development and operations team by this clients requirements are fulfilled within time and at low budget. The collaboration between the two teams will increase the quality and security of the code.

DevOps is team of Development and Operations where they both work for product lifecycle, Collaborate each other and work efficiently.

DevOps is about removing the barriers between the traditional development and Operations team separately.

Key principles of DevOps:

Automation of software development lifecycle, Collaboration and communication, Continuous integration, Feedback.

Pipeline: It is used to automate the delivery of code instead of using manual work. The components of pipeline is CI/CD, automated testing, reporting.

Traditional IT vs DevOps: Traditional IT is like developers use different versions and different platforms to write code. Due to this production and code debugging might be tough. Coming to DevOps, here all teams members deploy their code into central repository itself. Debugging and production is easy and fast.

DevOps practices:

7c:Continuous Development,Integration,Testing,Deployment,Feedback,monitoring,operations.

**Continuous Integration**: It is a development practice where developers integrate the code in central repository. It is used to find the bugs in code easily and quicker. It refers to a build and unit testing stages of a software release process.

**Continuous Delivery**: It is a software development practice where code changes are automatically prepared for release to production.

**Testing**- It is done to check whether the developed software is bug free and suitable to the requirements or not.

Testing is divided into two steps-Verification and validation.

**Verification**: Verifying that the product which are building is right or not!

**Validation**: Verifying that the right product is been built or not!

The main objective of the testing is to release a quality product to the customer.

Quality product is nothing but a bug free application to the client. This Bugs or errors can be rectified and modified by testing only.

Why testing is important?

Let us take a example a banking application is built and without testing released to production, due to some errors in code clients cant do a transaction online or not showing available balance. This is called errors in application to overcome this testing is done before sent to production.

**Error**: When developer fails to understand the logic or requirement of client. A mistake made in the code .That might be a syntax error, logical error, runtime error, etc..

**Bug:** System failed to perform the required tasks is called as a bug.

**Failure**: If there is a multiple defects that lead to a software failure.

Bugs occur due to lack of communication, specification, code, design.

Software testing is majorly of two types:

1- Manual testing

2-Automation testing.

**Manual Testing**-Here do not use any testing tools, test cases are prepared by humans and test the software manually. This testing works well for Functionalities, user Interface, website behaviour, user acceptance.

**Automation Testing**- Here we use pre-scripted tools to test the software. It finds more bugs compared manual testing, it allows of reuse of tests.