DASA DevOps Principles

The DevOps Agile Skills Association defines certain principles for DevOps

1. **Customer-centric Action**
   Companies must constantly innovate and invest in goods and services that would delight customers to the greatest extent possible.

2. **Create with end in mind**
   The principle focus on identifying customer's true needs and working towards creating products and services that address those demands.

3. **End-to-End Responsibilities**
   The term End-to-End refers that the team is responsible for the quality and quantity of services it provides to its clients.

4. **Cross-functional autonomous teams**
   The team must be completely self-contained throughout the product life cycle
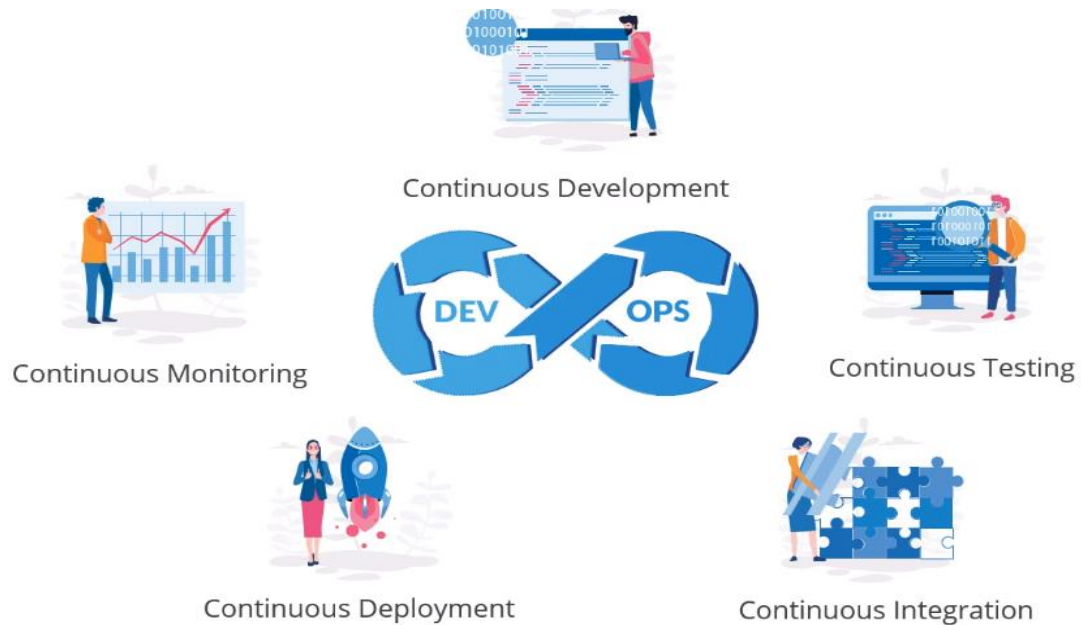
5. **Continuous Improvements**
   Continuous improvements enhances the products and services offered to customers

6. **Automate everything you can**
   Automation is associated with the goal of redefining how the team delivers its services.

DevOps Lifecycle

1. **Continuous development** manifest planning and coding. The scope of the project is determined during the planning and developers build the code. DevOps tools are not required for planning

2. **Continuous Testing** is the phase in which the application is tested for bugs. Automation Test tools like TestNG, Junit, Selenium and others are used for continuous testing.
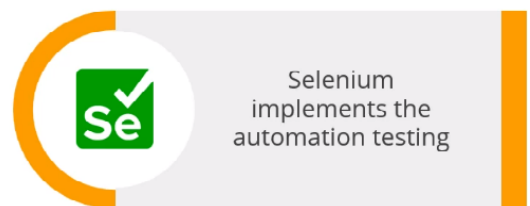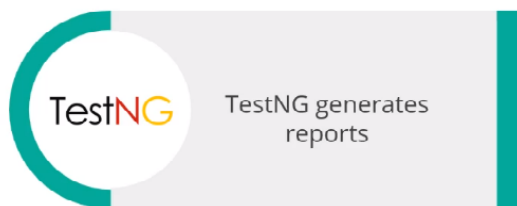
   ✓ Docker Container simulate the test environment in continuous testing
   ✓ Jenkins assist in conducting automation of the test procedure
   ✓ Automation testing saves a lot of time and work, unlike manual testing

3. **Continuous Integration** is an essentials phase of DevOps which deals with software development

    Frequent changes are made to the source code

    Building code is not only involved in compilation but also involves

    - Unit Testing
    - Integration Testing
    - Code review
    - Code Packaging

    The existing code is continuously and seamlessly put together with the new code which leads to development of software.
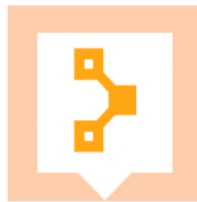
4. **Continuous Deployment** means that the implementation code is deployed across all production server.

    New code is continuously deployed and configured tools are required to manage it effectively.

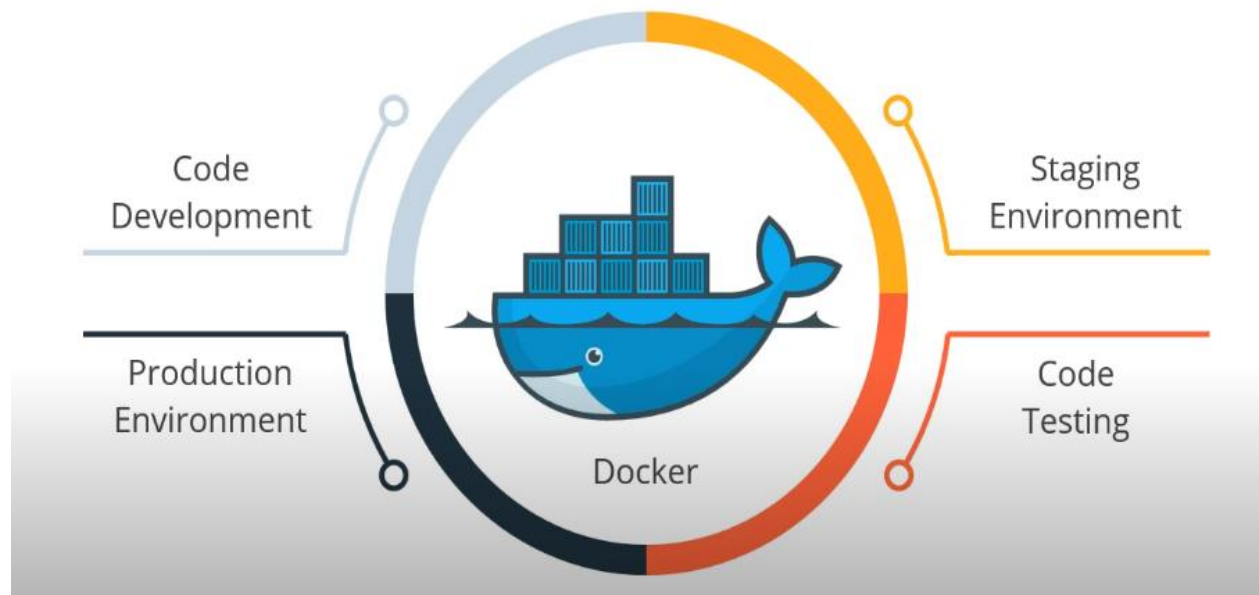   Popular tools are used in this phase

Chef　　　Puppet　　　Ansible　　　SaltStack

Containerization plays a pivotal role in the deployment process.

5. **Continuous Monitoring** includes all the operational aspects where critical information about the software use is recorded and analyzed to derive analytics

Popular tools are