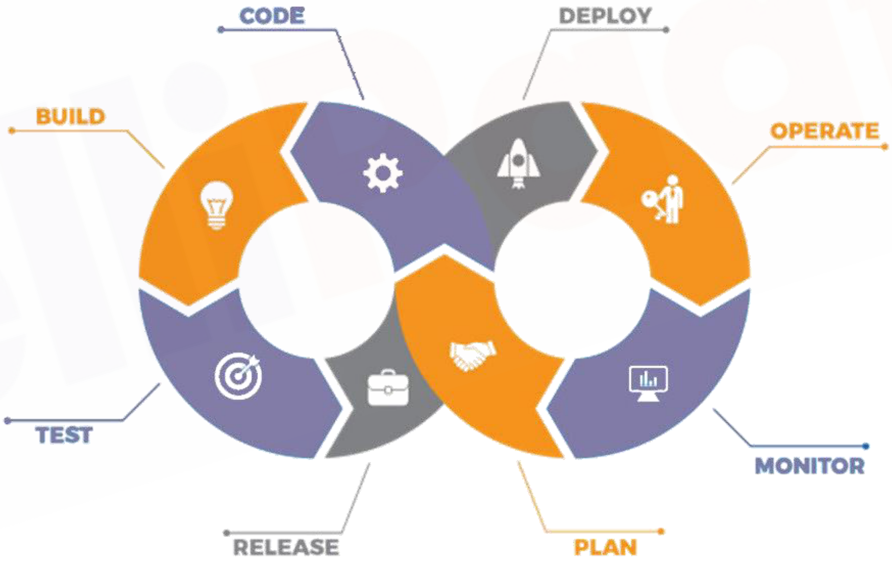
Introduction to Selenium



Agenda

01 Introduction to Software Testing

02 Introduction to Selenium

03 What is Maven?

04 Creating Automated Tests

05 Introduction to TestNG

06 Introduction to Continuous Testing

Introduction to Software Testing

Introduction to Software Testing

Software testing is defined as an activity to check whether the actual results match with the expected results and to ensure that the software system is defect free.

Types of Software Testing

Types of Software Testing

Automated Testing	Manual Testing
Test cases are executed automatically	Tests cases are executed manually
More accurate	Less accurate
More suitable when test cases are run repeatedly	More suitable when test cases are supposed to run only once or twice
Suitable for scenarios when testing is functionality based	More suitable when testing is for user experience

Automated Testing Tools

Types of Testing

Selenium

Appium

Cucumber

Test Studio

Introduction to Selenium

What is Selenium?

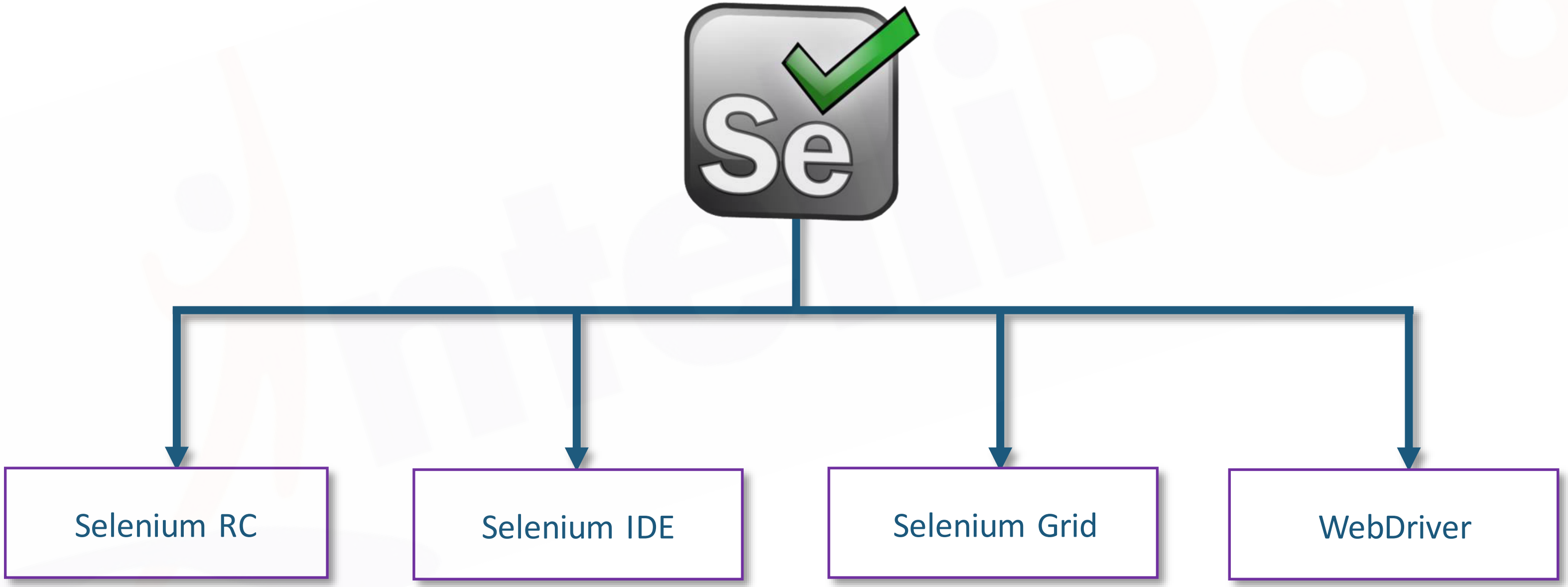
Selenium is a portable framework for testing web applications. Selenium provides a playback tool for authoring functional tests without the need to learn a new test scripting language.

What is Selenium?

- Selenium was first created by **Jason Huggins** in **2004**.
- He was working on a web application which had to be frequently tested.
- Since manual testing was taking a lot of time, he wrote a JavaScript program.
- This program could automatically control the browser's actions.
- He named it as ***JavaScriptTestRunner*** and donated it to the open-source community.
- It was later named as **Selenium Core**.

Components of Selenium

But only the Selenium Core could not suffice all the use cases of testing. Hence, a suite of Selenium components was developed for different purposes.



Components of Selenium

Selenium RC

Selenium IDE

Selenium Grid

WebDriver

- Due to same origin policy, testers had to install Selenium Core and the web server on their local system.
- This was done to keep the domain same for the Selenium Core and the web application to be tested.
- Selenium RC is a web server, which acts as an HTTP proxy.
- It tricks the OS into believing both Selenium Core and the website to be tested are on the same domain.
- This system was also known as Selenium 1.

Components of Selenium

Selenium RC

Selenium IDE

Selenium Grid

WebDriver

- Selenium IDE was originally created as a Firefox extension.
- It could automate tests using the record and playback feature.
- The intention behind creating this component was to increase the speed of creating test cases in Selenium.
- It was created by Shinya Kasatani from Japan.

Components of Selenium

Selenium RC

Selenium IDE

Selenium Grid

WebDriver

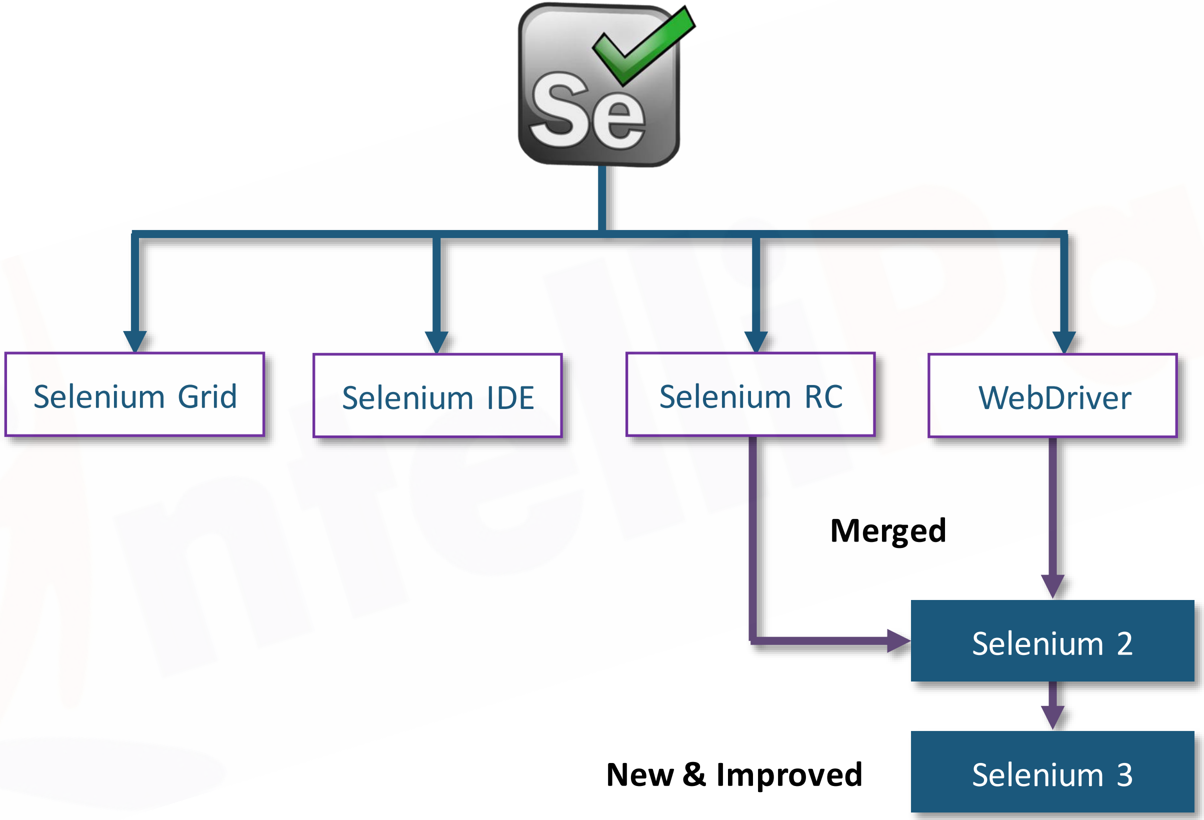
- Selenium Grid enables parallel testing of applications on multiple machines.
- It was primarily created to minimize the time taken in executing test cases.
- It can be used across multiple browsers and OS.
- It can also be used to break down a huge test suite among many computers testing the same application.

Components of Selenium

- Selenium RC
- Selenium IDE
- Selenium Grid
- **WebDriver**

- Selenium WebDriver was the first cross-platform testing framework that could control the browser from the OS level.
- It was developed in 2006, when web applications and browsers were becoming more powerful and restrictive with JavaScript programs like Selenium Core.
- It was better than Selenium IDE and RC.
- It controls the browser by directly communicating with it.

History of Selenium



Selenium Grid

Selenium IDE

Selenium RC

WebDriver

Merged

Selenium 2

New & Improved

Selenium 3

What is Maven?

What is Maven?

Maven is a build automation tool used primarily for Java projects. Maven addresses two aspects of building software: first, it describes how a software is built and, second, it describes its dependencies.

Why do we need Maven?

- Maven is used to download dependencies for a software program.
- Dependencies to be downloaded are included inside a POM file.
- Once the dependencies are added in the POM file, simply save the project and all the dependencies will automatically be downloaded.

Hands-on: Setting up Selenium with Maven

Setting up Selenium with Maven

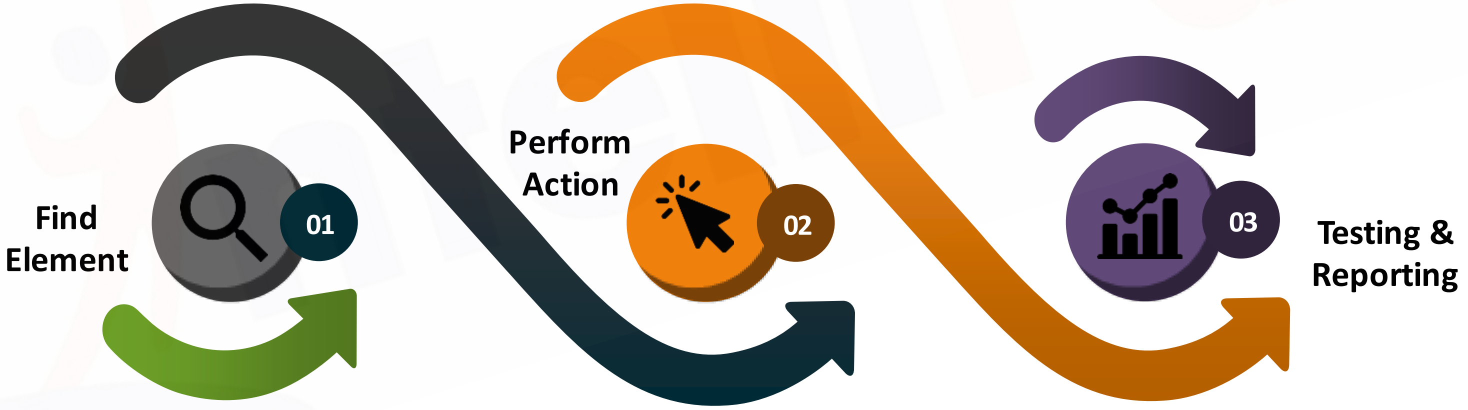
- Install Java
- Download and Install Eclipse for Developers
- Create a Maven Project
- Acquire Maven Dependencies for Selenium

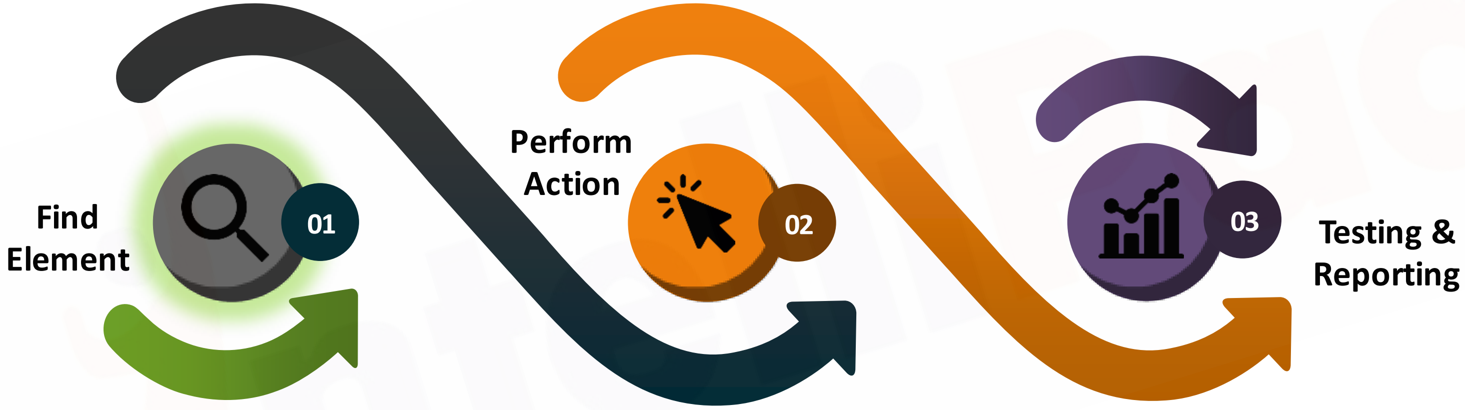
Creating Automated Tests

Creating Automated Tests

There are three steps to executing a web test:

- Find the element on the web browser
- Perform an action on the found element(s)
- Test and create a test report with the results





Find Element

01

Perform Action

02

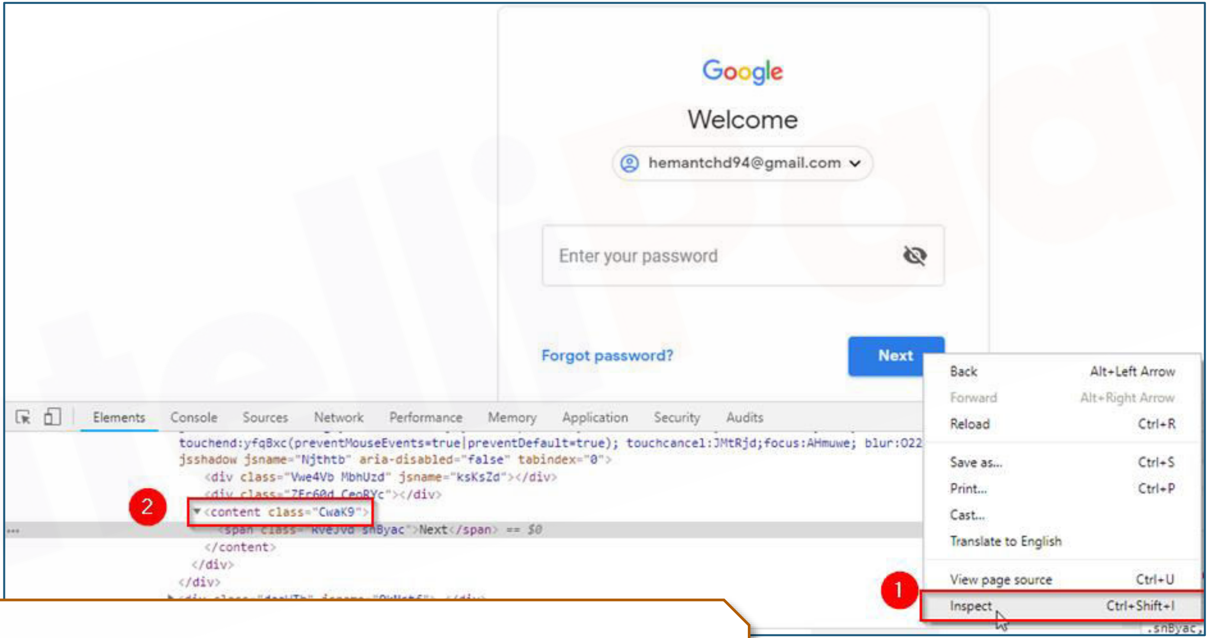
03

Testing & Reporting

Finding Elements in Selenium

An element can be found on a web page using the following selectors:

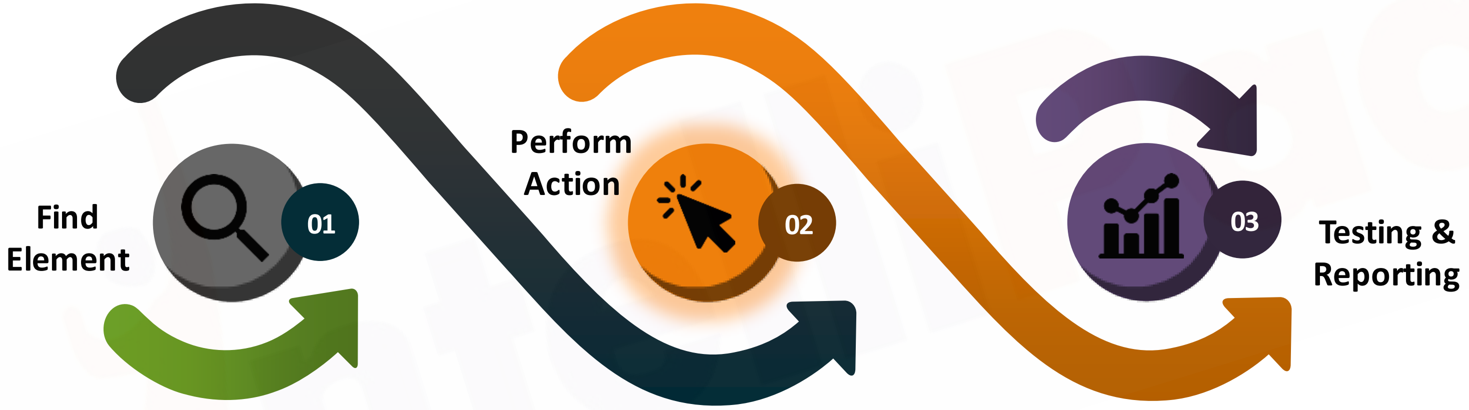
- ID
- Name
- Class Name
- Tag Name
- Link Text
- Partial Link Text
- XPATH



```
touchend:yfqBxc(preventMouseEvents=true|preventDefault=true); touchcancel:JMtRjd;focus:AHmuwe; blur:O22
jsshadow jsname="Njthtb" aria-disabled="false" tabindex="0">
   <div class="Vwe4Vb MbhUzd" jsname="ksKsZd"></div>
   <div class="ZFc60d CeoRYc"></div>
  ▼<content class="CwaK9">
     <span class="RveJvd snByac">Next</span> == $0
   </content>
  </div>
</div>
```

Syntax

WebElement button = driver.findElement(By.class("CwaK9"));



Find Element

01

Perform Action

02

03

Testing & Reporting

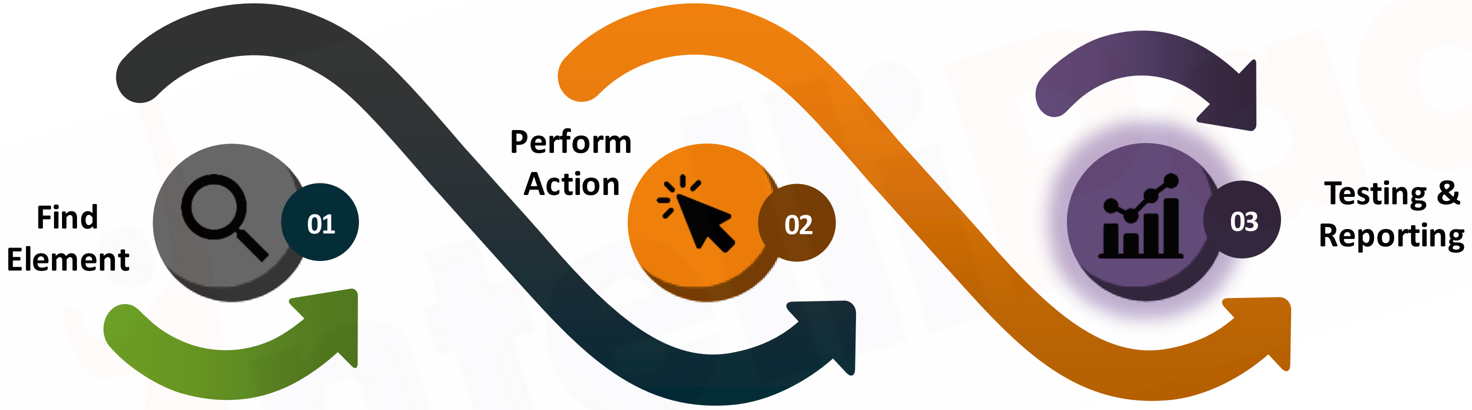
Performing Action on Elements

The next step is taking an action. For that, one can try the following options:

- **Click()**: Used to click on the link and wait for page load to complete before proceeding to the next command
- **sendKeys()**: Used to enter values onto text boxes
- **Clear()**: Used to clear text boxes of its current value
- **Submit()**: WebDriver will automatically trigger the submit function of the form where that element belongs to

Syntax

```
WebElement button = driver.findElement(By.class(“CwaK9")).click();
```


Find Element
01
Perform Action
02
03
Testing & Reporting

Testing & Reporting in Selenium Using TestNG

What is TestNG?

TestNG is an open-source automated testing framework, where **NG** means ”**N**ext **G**eneration”. It is a testing framework inspired from JUnit and NUnit, but introducing some new functionalities makes it more powerful and easier to use.

Features of TestNG

- TestNG annotations are easy to create test cases.
- Test cases can be grouped and prioritized more easily.
- It supports parameterization.
- It supports data-driven testing using DataProviders.
- It generates HTML reports.
- Parallel test execution is possible.
- It readily supports integration with other tools and plug-ins like Eclipse IDE, build tools Ant, Maven etc.

Hands-on: Setting up TestNG

Annotations in TestNG

Annotations in TestNG

Annotations in TestNG are used to decide the flow of the program. There are a lot of annotations in TestNG, we will focus on the most used ones and following are the same:

@BeforeSuite: The annotated method will be run only once before all tests in this suite have run.

@BeforeTest: The annotated method will be run before any test method belonging to the classes inside the <test> tag is run.

@BeforeClass: The annotated method will be run only once before the first test method in the current class is invoked.

@BeforeMethod: The annotated method will be run before each test method.

@Test: This marks a class or a method as a part of the test.

Hands-on: Working with Annotations in TestNG

Hands-on: Creating Automated Test with TestNG

Hands-on: Creating Our First Test Case

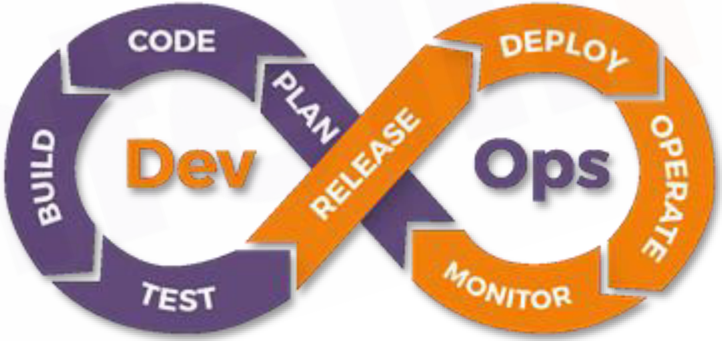
- Open the Intellipaat website, by visiting www.intellipaat.com
- Enter “DevOps” term and click search
- On the search page, check for “DevOps Certification Course”. If it exists, click on the course
- On the course page, verify if in the page “DevOps Certification Training” is present as the header

Hands-on: Running a Headless Test in Selenium

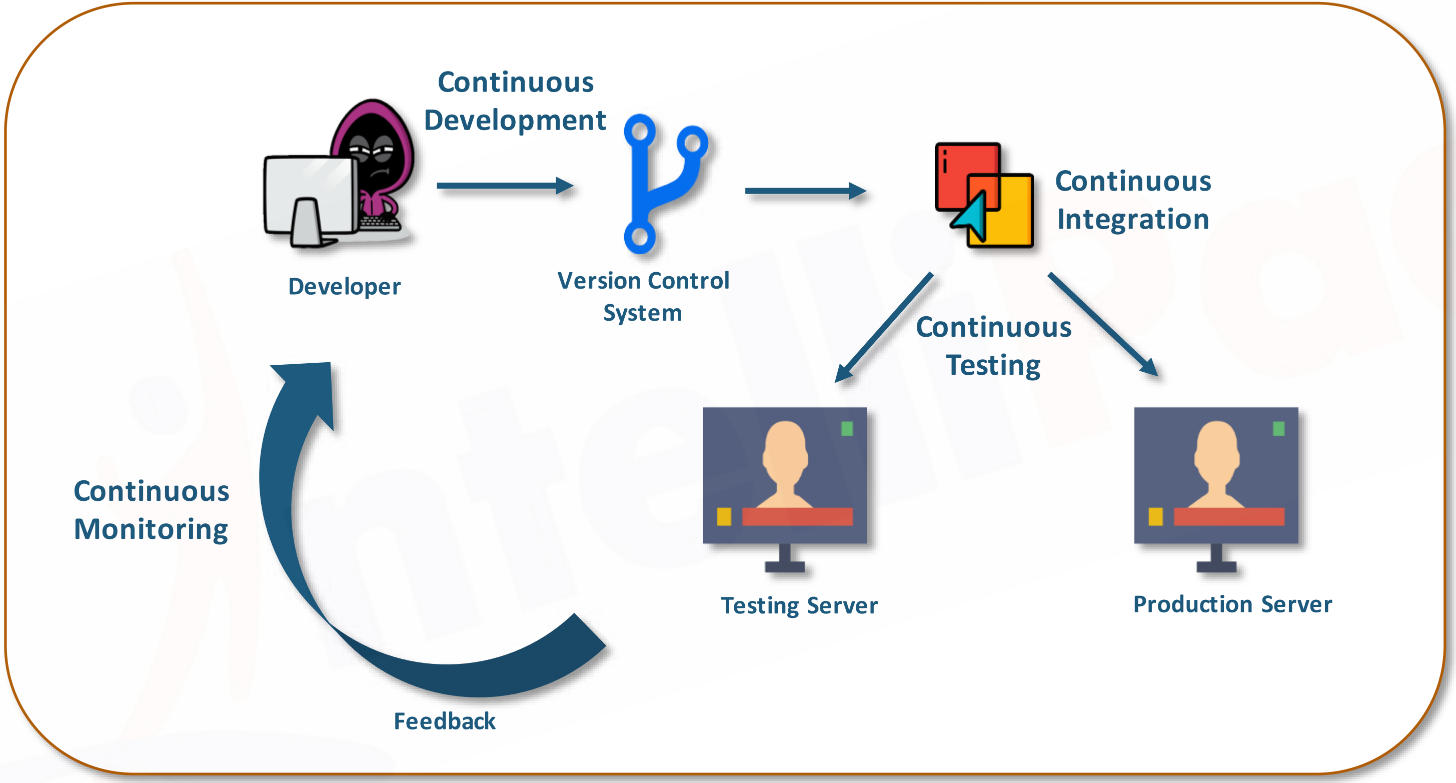
Introduction to Continuous Testing

Introduction to Continuous Testing

Continuous Testing is the process of executing **automated tests** as part of the software delivery pipeline in order to obtain feedback on the business risks associated with a software release candidate as rapidly as possible.



Introduction to Continuous Testing



Software Delivery Pipeline

Introduction to Continuous Testing

Products are built with their respective test suites while the features are being developed.

Production Server

Product Features

Feature A

Testing Server

Test Suite

Test suite for Feature A

Introduction to Continuous Testing

Products are built with their respective test suites while the features are being developed.

Production Server

Product Features

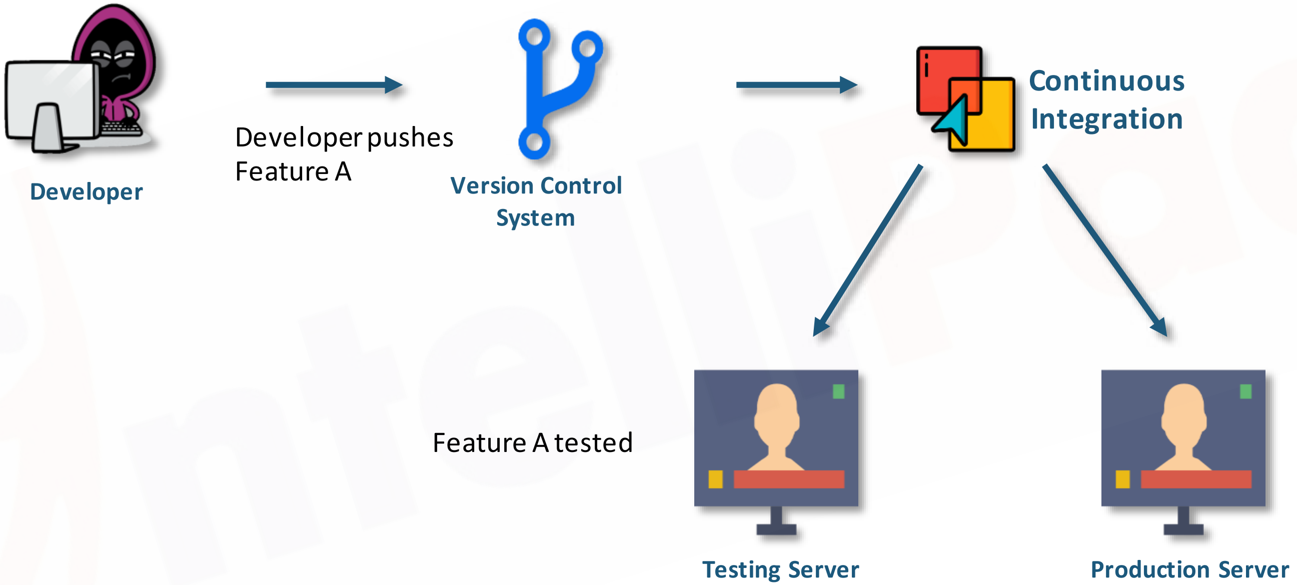
Feature A
Feature B

Testing Server

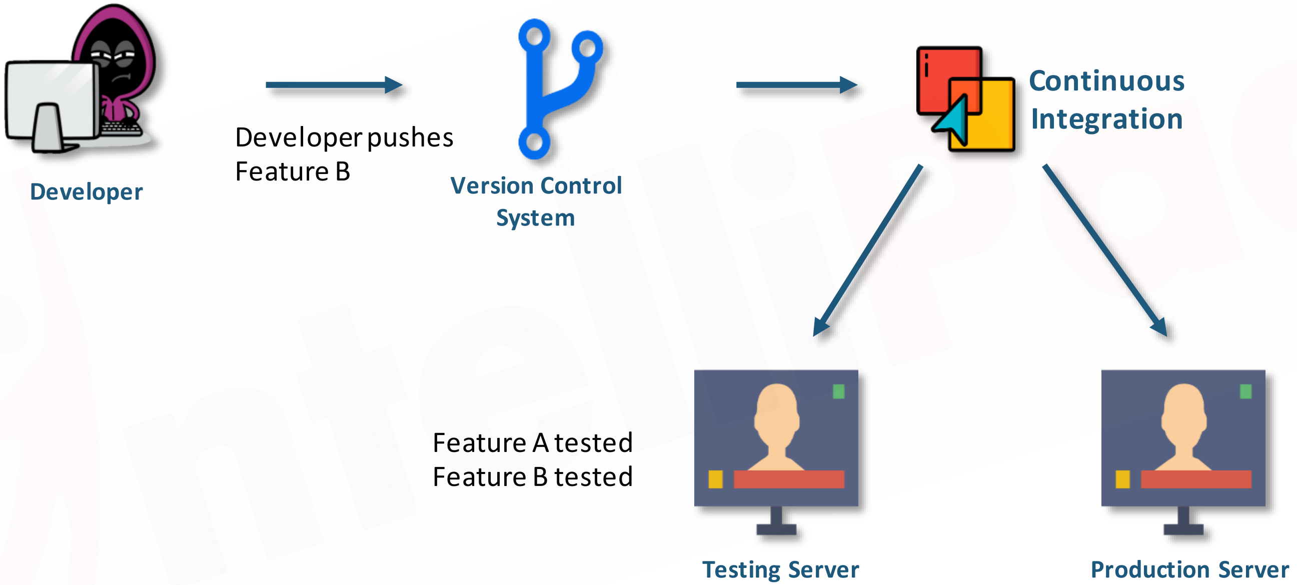
Test Suite

Test suite for Feature A
Test suite for Feature B

Introduction to Continuous Testing



Introduction to Continuous Testing



Developer

Developer pushes Feature B

Version Control System

Continuous Integration

Feature A tested
Feature B tested

Testing Server

Production Server

Quiz

Quiz

1. Can Selenium automate Desktop Applications?

A. No

B. Yes

C. Only Microsoft Applications

D. None of these

Quiz

1. Can Selenium automate Desktop Applications?

[✓ A. No]

B. Yes

C. Only Microsoft Applications

D. None of these

Quiz

2. What does Maven do?

A. Creates Test Reports

B. Dependency Management

C. Helps in Testing Applications

D. None of these

Quiz

2. What does Maven do?

A. Creates Test Reports

[✓ **B. Dependency Management**]

C. Helps in Testing Applications

D. None of these

Quiz

3. Which component in Selenium helps in parallel test execution?

A. Selenium RC

B. Selenium Grid

C. Selenium WebDriver

D. None of these

Quiz

3. Which component in Selenium helps in parallel test execution?

A. Selenium RC

[✓ **B. Selenium Grid**]

C. Selenium WebDriver

D. None of these

Quiz

4. Annotations in Selenium help us _________.

A. Follow a sequence in Selenium code

B. Comment code

C. Create reports

D. None of these

Quiz

4. Annotations in Selenium help us ________.

[✓ A. Follow a sequence in Selenium code]

B. Comment code

C. Create reports

D. None of these

Quiz

5. What is Headless Mode?

A. Runs Selenium tests in Headless Mode

B. Runs Selenium tests in foreground

C. Runs Selenium tests in background

D. None of these

Quiz

5. What is Headless Mode?

A. Runs Selenium tests in Headless Mode

B. Runs Selenium tests in foreground

[✓ **C. Runs Selenium tests in background**]

D. None of these

IntelliPaat

India: +91-7847955955

US: 1-800-216-8930 (TOLL FREE)

support@intellipaat.com

24/7 Chat with Our Course Advisor