Prepared by

Ahmet Bayram

WHAT IS NEW IN SELENIUM 4

> The rest of the slides are about Selenium 4 new features

WHAT IS SOFTWARE TESTING?

> It is a process of checking if **EXPECTED RESULT** equals **ACTUAL RESULT**

> > If Expected result = Actual result, then PASS

> > If Expected result !=Actual result, then FAIL.

> **Be Positive(HAPPY PATH)** and **Negative(negativity works in testing!)**

> Testing is done to meet the customer/business/client needs.

> To test an application, we use **user stories** and **acceptance criterias**.

WHY TESTING IS IMPORTANT?

- We need testing to have **BUG/DEFECT(UNEXPECTED RESULT) FREE** applications.

- Whenever software is being created, testing is a must.

- Testing results in reputations/reliability etc. of company.

- Simply testing saves money and lives(Think of Boeing crash and Boeing stock chart)

MANUAL TESTING?

❯ Manual testing is to test the application manually without any automation tool.

❯ Manual testers uses limited technology(exel,etc.) for documentation, but not automation tools or language.

❯ In manual testing, human error can happen.

❯ All Automation testers must first do manual testing to understand the application

❯ A good automation tester is also a good manual tester.

AUTOMATION TESTING

❯ Testing a system with the help of automation tools is called automation testing.

❯ Automation testers are more technical

❯ Coding Language

❯ Other Automation Tools

  ❯ Selenium

  ❯ Maven

  ❯ JUnit

  ❯ TestNG

  ❯ Cucumber

  ❯ ....

SELENIUM 4 NEW FEATURES

WORLD WIDE WEB CONSORTIUM W3C

> **World Wide Web Consortium W3C is created by Tim Berners-Lee from MIT and CERN**

> **No need for the API codes to resolve, hence the direct communications happens**

SCREENSHOT

> **With selenium 3, taking the entire screen show was possible**

> **With selenium 4, we can even take the screenshots of specific elements**

```
WebElement logo = driver.findElement (By.xpath("//div[@id='divLogo']//img"));
File file = logo.getScreenshotAs(OutputType.FILE);
File destFile =new File("logo.png");
FileUtils.copyFile(file,destfile);
```

NEW TABS AUTOMATION

> **With selenium 4, we can open new URLs in multiple tabs**

> **For example, below code will open 2 different URL in 2 different tabs**

```
driver.get(https://www.google.com/);

newTab = driver.switchTo().newWindow(WindowType.TAB);

newTab.get("https://www.techproeducation.com/");
```

NEW WINDOW AUTOMATION

> **With selenium 4, we can open new URLs in multiple windows**

> **For example, below code will open 2 different URL in 2 different window**

```
driver.get("https://www.google.com/");

newWindow = driver.switchTo().newWindow(WindowType.WINDOW);

newWindow.get("https://www.techproeducation.com/");

System.out.println(driver.getTitle());
```

POSITIONS OF THE ELEMENTS

> **With Selenium 4, we can get the coordinates, height, width, etc. of specific elements**

> **For example, below code will return the dimensions of the logo**

```
<picture>
  <img class="lnXdpd" alt="Seasonal Holidays 2021"
  height="200" src="/logos/doodles/2021/seasonal-ho
  lidays-2021-6753651837109324-6752733080595603-cs
  t.gif" title="Seasonal Holidays 2021" width="500"
  id="hplogo" data-atf="1" data-frt="0"> == $0
</picture>
```

```
WebElement logo1=driver.findElement(By.xpath("//div[@id='divLogo']//img"));
System.out.println("Height:" +logo.getRect().getDimension().getHeight());
System.out.println("Height:" +logo.getRect().getDimension().getWidth());
System.out.println("X Location: " +logo.getRect().getX());
System.out.println("Y Location: " +logo.getRect().getY());
```

RELATIVE LOCATORS

❯ **With Selenium 4, we can find elements relative to other elements**

❯ **This locator is also known as Friendly Locators**

❯ **5 new locators are :**

below( ): finds the elements below the selected element

toLeftOf( ): finds the elements on the left of the selected element

toRightOf( ): finds the elements on the right of the selected element

above( ): finds the elements above the selected element

near( ): finds the elements near to the selected element-
approximately 50 pixel

Example: toLeftOf() and below() locators

WebElement book =
driver.findElement(RelativeLocators.withTagName("li").toLeftOf(By.id("pid1")) .below(By.id("pid2")));
String id1=book.getAttribute ("id1");
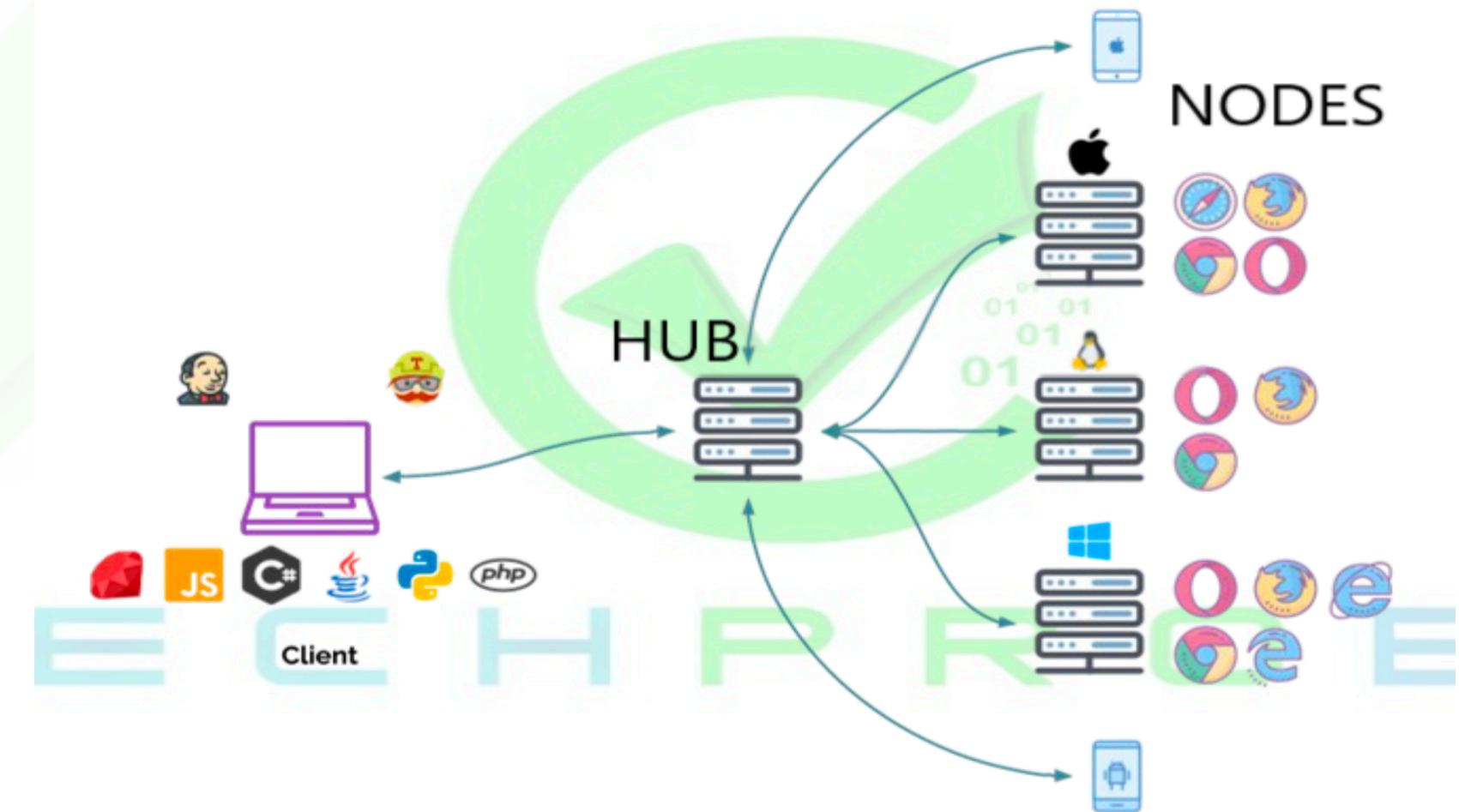
Example: toRightOf() and above() locators

WebElement book1 =
driver.findElement(RelativeLocators.withTagName("li").toRightOf(By.id("pid1")) .above(By.id("pid2")));
String id2=book1.getAttribute ("id2");

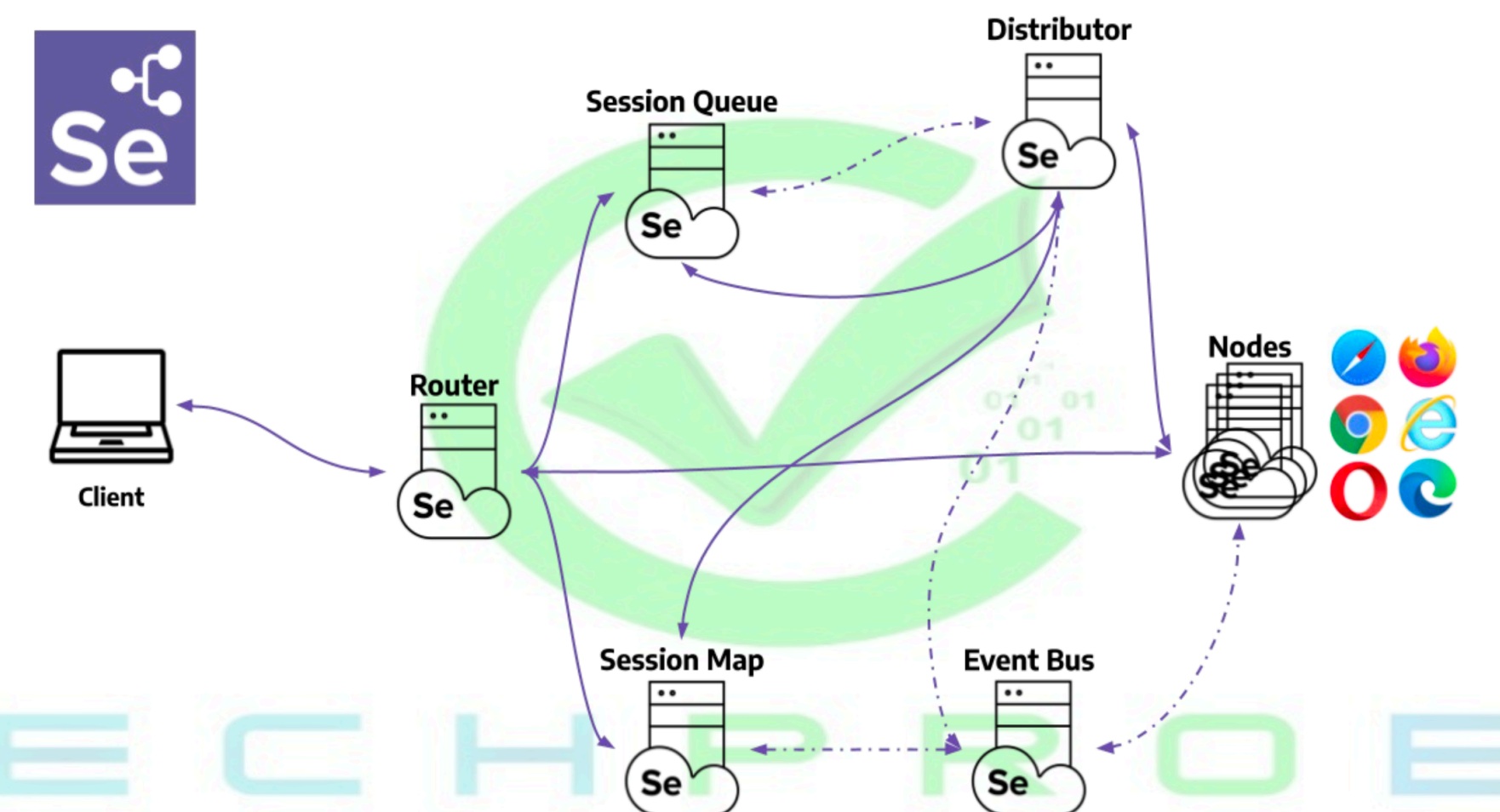# SELENIUM GRID AND DESIRED CAPABILITIES

Selenium Grid 3

> Selenium 3 uses desired capabilities for defining drivers and use selenium grid

> Selenium 4 now uses Options object that makes coding shorter.

> Selenium grid architecture is also now different

Firefox – FirefoxOptions
Chrome – ChromeOptions
Internet Explorer (IE) – InternetExplorerOptions
Microsoft Edge – EdgeOptions
Safari – SafariOptions
ChromeOptions options = new ChromeOptions();
options.setAcceptInsecureCerts(true);
driver.get("https://www.techproeducation.com");

Selenium Grid 4

ACTIONS CLASS CHANGES

> **Actions class is used for mouse and keyboard actions, such as right click, double click, drag and drop, etc.**

> **With selenium 4, the changes are below:**

click(WebElement) instead of **moveToElement(onElement).click()**

clickAndHold(WebElement) instead of **moveToElement(onElement).clickAndHold()**

contextClick(WebElement) instead of **moveToElement(onElement).contextClick()**

doubleClick(WebElement) instead of **moveToElement(element).doubleClick()**

release() is now in Actions class

FLUENTWAIT

> **With selenium 4, pollingEvery and withTomeout now accepts only one parameter**

```
// selenium 3
FluentWait wait = new FluentWait(driver)
.pollingEvery(20, TimeUnit.MILLISECONDS)
.withTimeout(20, TimeUnit.SECONDS)
.ignoring(NoSuchElementException.class);
```

```
//selenium 4
FluentWait wait = new FluentWait(driver)
.pollingEvery(Duration.ofMillis(500))
.withTimeout(Duration.ofSeconds(60))
.ignoring(NoSuchElementException.class);
```