Driver Initilization	
WebDriver driver = new ChromeDriver();	
WebDriver driver = new FirefoxDriver();	
WebDriver driver = new InternetExplorerDriver();	
WebDriver driver = new HtmlUnitDriver();	
Element Locators	
driver.findElement(By.id("Id Value"));	
driver.findElement(By.name("Name Value"));	
driver.findElement(By.className("Class Name Value"));	
driver.findElement(By.linkText("Link text Value"));	
driver.findElement(By.partialLinkText("Partial Text Constant	
Value"));	
driver.findElement(By.tagName("Tag Name Value"));	
driver.findElement(By.cssSelector("CSS Value"));	
driver.findElement(By.xpath("Xpath Value"));	
driver.findElement(new ByAll(By.className("ElementClass	
Name"), By.id("Element Id"), By.name("Element Name")))	
Elements Operations	
WebElement element =	
driver.FindElement(By.ElementLocator("Value of Element	
Locator"));	
element.click();	
element.sendKeys("Input Text");	
element.clear();	
element.submit();	
element.getAttribute("type");	
String innerText = element.getText();	
boolean enabledstatus = element.isEnabled();	
boolean displayedstatus = element.isDisplayed();	
boolean selectedstatus = element.isSelected();	
//Operation on drop down	
Select select = new Select(element);	
select.selectByIndex(Integer Index);	
select.selectByVisibleText("Text");	
select.SelectByValue("Value");	
select.deselectAll();	
select.deselectByIndex(Integer Index);	
select.deselectByVisibleText("Text");	
select.deselectByValue("Value");	
WebElement selectedOptions = select.getOptions();	
Browser Operations	
String pageTitle = driver.getTitle();	
String currentURL = getCurrentUrl();	
String currentPageSource = driver.getPageSource();	
// Navigation history	
driver.get("https://www.facebook.com/");	
driver.manage().window().maximize();	
driver.navigate().to("https://www.google.com/");	
driver.navigate().back();	
driver.navigate().forward();	
driver.navigate().refresh();	

driver_cautt(); // Handle Alert Alert alert = driver.switchTo().alert(); alert.accept(); ale	driver.close();		
Manual Calert = driver.switchTo().alert();			
Alert alert = driver.switchTo().alert(); alert.accept(); alert.accept(); alert.getText(); driver.manage().getCookie(); alert.getText(); alert.getText(); betText(); alert.getText();			
alert.accept(): alert.dsmiss(): alert.getText(): alert.sendKeys("input Data"); //Handle Cookies Cookie cookie = new Cookie("cookieName", "cookieValue"); driver.manage().addCookie(cookie); driver.manage().getCookieNi; driver.manage().getCookieNi; driver.manage().getCookieNi; driver.manage().deltexalCookie(s(); driver.manage().deltexalCookieNi; driver.manage().deltexalCookieNi; driver.manage().deltexalCookieNi; driver.manage().deltexalCookieNi; driver.switchTo().frame(int Frame Index); driver.switchTo().frame(int Frame Index); driver.switchTo().frame("frameName"); WebElement element = driver.FindElement(By.ElementLocator("Value of Element Locator")); driver.switchTo().frame(element); driver.switchTo().frame(element); driver.switchTo().defaultContent(); Screenshots Capture TakesScreenshot screenshot = ((TakesScreenshot)driver); File srcFile= screenshot.getScreenshotAs(OutputType.FiLE); File+andler.copy(srcFile, destFile); Manage Timeouts driver.manage().timeouts().implicitlyWait(10, TimeUnit.SECONDS); welement = wait.until(syntax: WebDriverWait wait = new WebDriverWait(driver, timeout); ExpectedConditions.elementToBeClickable(locator)); welement.click(); Thread.sleep(long milli-seconds) driver.manage().timeouts().pageLoadTimeout(30, TimeUnit.SECONDS); Scroll Down or Up Web Page JavascriptExecutor j = (JavascriptExecutor)driver; js.executeScript("window.scrollTo(0, document.body.scrollHeight"); WebElement element = driver.findElement(By.ElementLocator("Value of Element Locator")); js.executeScript("window.scrollTo(0, document.body.scrollHeight"); WebElement element = driver.findElement(By.ElementLocator("Value of Element); Different approach to create XPath and CSS Selector			
alert.getText(); alert.getText(); alert.getText(); alert.getText(); alert.getCookies Cookie cookie = new Cookie("cookieName", "cookieValue"); driver.manage().addCookie(cookie); driver.manage().getCookieNg); driver.manage().getCookieNg); driver.manage().getCookieNg); driver.manage().deleteAllCookieS(); driver.manage().deleteAllCookieS(); driver.manage().deleteCookieNamed(arg0); // Handle frames driver.minto().frame(int Frame Index); driver.switchTo().frame(int Frame Index); driver.switchTo().frame(int Frame Index); driver.switchTo().frame(int Frame Index); driver.switchTo().frame(element); driver.switchTo().frame(element); driver.switchTo().frame(element); driver.switchTo().defaultContent(); Screenshots Capture TakesScreenshot screenshot =((TakesScreenshot)driver); File srcFiles screenshot.getScreenshotAs(OutputType.File.); FileHandler.copy(srcFile, destFile); Manage Timeouts driver.manage().timeouts().implicitlyWait(10, TimeUnit.ScCONDS); welement = wait.until(Syntax: WebDriverWait wait = new WebDriverWait(driver, timeouts); ExpectedConditions.elementToBeClickable(locator)); welement.click(); Thread.sleep(long milli-seconds) driver.manage().timeouts().pageLoadTimeout(30, TimeUnit.ScCONDS); Scroil Down or Up Web Page JavascriptExecutor js = JlavascriptExecutor)driver; js.executeScript("window.scroilTo(0, document.body.scroilHeight("); WebEllement element = driver.FindElement(By.ElementLocator("Value of Element Locator")); js.executeScript("window.scroilTo(0, document.body.scroilHeight("); WebEllement element = driver.FindElement(By.ElementLocator("Value of Element); Different approach to create XPath and CSS Selector			
alert.getText(); alert.sendKeys("Input Data"); //Handle Cookies Cookie cookie = new Cookie("cookieName", "cookieValue"); driver.manage(),addCookie(cookie); driver.manage(),getCookieName(arg0); driver.manage(),getCookieName(arg0); driver.manage(),deleteAllCookies(); driver.manage(),deleteCookieName(arg0); // Handle frames driver.switchTo(),frame(int Frame Index); driver.switchTo(),frame(element); driver.switchTo(),deaultContent(); Screenshot Screenshot ([TakesScreenshot)driver); File srcFile= screenshot,getScreenshotAs(OutputType,File); FilesrcFile= screenshot,getScreenshot,getScreenshot,getScreenshot,getScreenshot,getScreenshot,getScreenshot,getScreenshot,getScreenshot,getScreenshot,getScreenshot,getScreenshot,getScreenshot,getScreenshot,getScreenshot,getScreenshot,getScreenshot,getScreenshot,getScreenshot,getScreenshot,getScreenshot,getScreenshot,getScreenshot,getScreenshot,getScreenshot,getScreenshot,getScree			
alert.sendKeys("Input Data"); //Handle Cookies Cookie cookie = new Cookie("cookieName", "cookieValue"); driver.manage(),addCookie(cookie); driver.manage(),getCookieNg); driver.manage(),getCookieNg); driver.manage(),getCookieNg); driver.manage(),deleteAllCookieS(); driver.manage(),deleteAllCookieS(); driver.manage(),deleteAllCookieNg); // Handle frames driver.manage(),fetCookieNamed(arg0); // Handle frames driver.manage(),frame(int Frame Index); driver.switchTo(),frame(int Frame Index); driver.switchTo(),frame(int Frame Index); driver.switchTo(),frame(int Frame Index); driver.switchTo(),frame(element); driver.switchTo(),frame(element); driver.switchTo(),frame(element); driver.switchTo(),frame(element); driver.switchTo(),frame(element); friver.switchTo(),frame(element); driver.switchTo(),frame(element); friver.switchTo(),frame(element); driver.switchTo(),frame(element); friver.switchTo(),frame(element); friver.switchTo(),frame(element); friver.switchTo(),frame(element); friver.switchTo(),frame(element); friver.switchTo(),frame(element); friver.switchTo(),frame(element); friver.switchTo(),frame(element); fries srcPile = screenshot screenshot = ((TakesScreenshot)driver); File srcPile = screenshot screenshot = ((TakesScreenshot)driver); File srcPile = screenshot screenshot = ((TakesScreenshot)driver); File srcPile = screenshot = ((TakesScreenshot)driver); File			
//Handle Cookies = new Cookie("cookieName", "cookieValue"); driver.manage().addCookie(); driver.manage().getCookies(); driver.manage().getCookies(); driver.manage().getCookies(); driver.manage().deleteAllCookies(); driver.manage().deleteCookieNamed(arg0); // Handle frames driver.switchTo().frame(int Frame Index); driver.manage().timeouts().timplicitlyWait(10, TimeUnit.SECONDS); welement = wait.until(Syntax: WebDriverWait wait = new WebDriverWait(driver, timeout); ExpectedConditions.elementToBcClickable(locator)); welement = wait.until(Syntax: WebDriverWait wait = new WebDriverWait(driver, timeout); ExpectedConditions.elementToBcClickable(locator)); welement.click(); Thread.sleep(Long milli-seconds) driver.manage().timeouts().pageLoadTimeout(30, TimeUnit.SECONDS); Scroll Down or Up Web Page JavascriptExecutor js = (JavascriptExecutor)driver; js.executeScript("windox.scrollTo(0, document.body.scrollHeight)"); WebElement element = driver.findElement(By.ElementLocator("Value of Element Locator")); js.executeScript("windox.scrollTo(0, document.body.scrollHeight)"); WebElement element = driver.findElement(By.ElementLocator("Value of Element Locator")); js.executeScript("windox.scrollTo(0, document.body.scrollHeight)"); WebElement element = driver.findElement(By.ElementLocator("Value of Element);			
Cookie cookie = new Cookie("cookieName", "cookieValue"); driver.manage().addCookie(cookie); driver.manage().getCookieS(); driver.manage().getCookieNamed(arg0); driver.manage().deleteAlCookieS(); driver.switchTo().frame("frameName"); WebElement element = driver.fiver.fiver.manage().deleteAlCookieS(); driver.switchTo().frame(element); driver.switchTo().defaultContent(); Screenshot Screenshot screenshot =((TakesScreenshot)driver); File srcFile= screenshot.getScreenshotAs(OutputType.FiLE); FileHandler.copy(srcFile, destFile); Manage Timeouts driver.manage().timeouts().implicitlyWait(10, TimeUnit.SECONDS); welement = wait.until(Syntax: WebDriverWait wait = new WebDriverWait(driver, timeout); ExpectedConditions.elementToBeClickable(locator)); welement = wait.until(Syntax: WebDriverWait wait = new WebDriverWait(driver, timeout); ExpectedConditions.elementToBeClickable(locator)); welement.click(); Thread.sleep(Long milli-seconds) driver.manage().timeouts().pageLoadTimeout(30, TimeUnit.SECONDS); Scroll Down or Up Web Page JavascriptExecutor js = (JavascriptExecutor)driver; js.executeScript("window.scrollBy(0,100)"); js.executeScript("window.scrollBy			
driver.manage().addCookie(cookie); driver.manage().getCookies(); driver.manage().getCookieNamed(arg0); driver.manage().deleteAllCookies(); driver.manage().deleteAllCookies(); driver.manage().deleteCookieNamed(arg0); // Handle frames driver.switchTo().frame(int Frame Index); driver.switchTo().drame(int Frame Index); driver.switchTo().drame(element); driver.switchTo().drame(element); driver.switchTo().draultContent(); Screenshots Capture TakesScreenshot screenshot =((TakesScreenshot)driver); File srcFile= screenshot getScreenshotAs(OutputType.FILE); FileHandler.copy(srcFile, destFile); Manage Timeouts driver.manage().timeouts().implicitlyWait(10, TimeUnit.SECONDS); welement = wait.until(Syntax: WebDriverWait wait = new WebDriverWait(driver, timeout); ExpectedConditions.elementToBeClickable(locator)); welement.click(); Thread.sleep(Long milli-seconds) driver.manage().timeouts().pageLoadTimeout(30, TimeUnit.SECONDS); Scroil Down or Up Web Page JavascriptExecutor js = (JavascriptExecutor)driver; js.executeScript("window.scroilBy(0,100)"); js.executeScript("window.scroilBy(0,100)"); js.executeScript("window.scroilBy(0,100)"); js.executeScript("window.scroilFo(0, document.body.scroilHeight"); WebElement element = driver.FindElement(By.ElementLocator("Value of Element Locator")); js.executeScript("window.scroilFo(0, document.body.scroilHeight"); WebElement element = driver.FindElement(By.ElementLocator("Value of Element); Different approach to create XPath and CSS Selector			
driver.manage().getCookies(); driver.manage().getCookieNamed(arg0); driver.manage().detetAliCookies(); driver.manage().detetAliCookies(); driver.manage().detetAliCookies(); driver.manage().detetAliCookies(); driver.manage().detetAliCookies(); driver.manage().detetAliCookies(); driver.manage().detetAliCookies(); driver.switchTo().frame(int Frame Index); driver.switchTo().frame(int Frame Index); driver.switchTo().frame(int Frame Index); driver.switchTo().defaultContent(); Greenshot Scpture TakesScreenshot screenshot =((TakesScreenshot)driver); File srcFile= screenshot.getScreenshotAs(OutputType.FILE); FileHandler.copy(srcFile, destFile); Manage Timeouts driver.manage().timeouts().implicitlyWait(10, TimeUnit.SECONDS); welement = wait.until(Syntax: WebDriverWait wait = new WebDriverWait(driver, timeout); ExpectedConditions.elementToBeClickable(locator)); welement.click(); Tihread.sleep(Long milli-seconds) driver.manage().timeouts().pageLoadTimeout(30, TimeUnit.SECONDS); Scroll Down or Up Web Page JavascriptExecutor js = (JavascriptExecutor)driver; js.executeScript("window.scrollBy(0,100)");	cookie cookie - new cookie cookievanie, cookievalue j,		
driver.manage().deleteAllCookies(); driver.manage().deleteAllCookies(); driver.manage().deleteAllCookies(); driver.manage().deleteCookieNamed(arg0); // Handle frames driver.switchTo().frame(int Frame Index); driver.switchTo().frame(int Frame Index); driver.switchTo().frame(int Frame Name"); WebElement element = driver.FindElement(By.ElementLocator("Value of Element Locator")); driver.switchTo().defaultContent(); Screenshots Capture TakesScreenshot screenshot =((TakesScreenshot)driver); File srcFile = screenshot.getScreenshotAs(OutputType.FILE); FileHandler.copy(srcFile, destFile); Manage Timeouts driver.manage().timeouts().implicitlyWait(10, TimeUnit.SECONDS); welement = wait.until(Syntax: WebDriverWait wait = new WebDriverWait(driver, timeout); ExpectedConditions.elementToBeClickable(locator)); welement = wait.until(Syntax: WebDriverWait wait = new WebDriverWait(driver, timeout)) Timed.sleep(Long milli-seconds) driver.manage().timeouts().pageLoadTimeout(30, TimeUnit.SECONDS); Scroll Down or Up Web Page JavascriptExecutor js = (JavascriptExecutor)driver; js.executeScript("window.scrollBy(0,100)"); js.exec	driver.manage().addCookie(cookie);		
driver.manage().deleteAllCookies(); driver.manage().deleteCookieNamed(arg0); // Handle frames driver.switchTo().frame(int Frame Index); driver.switchTo().frame(iframeName"); WebElement element = driver.FindElement(By.ElementLocator("Value of Element Locator")); driver.switchTo().frame(element); driver.switchTo().frame(element); driver.switchTo().defaultContent(); Screenshots Capture TakesScreenshot screenshot =((TakesScreenshot)driver); File srcFile= screenshot.getScreenshotAs(OutputType.FiLE); FileHandler.copy(srcFile, destFile); Manage Timeouts driver.manage().timeouts().implicitlyWait(10, TimeUnit.SECONDS); welement = wait.until(Syntax: WebDriverWait wait = new WebDriverWait(driver, timeout); ExpectedConditions.elementToBeClickable(locator)); welement.click(); Thread.sleep(Long milli-seconds) driver.manage().timeouts().pageLoadTimeout(30, TimeUnit.SECONDS); Scroll Down or Up Web Page JavascriptExecutor js = (JavascriptExecutor)driver; js.executeScript("window.scrollBy(0,100)"); js.executeScript("window.scrollBy(0,1	driver.manage().getCookies();		
driver.manage().deleteAllCookies(); driver.manage().deleteCookieNamed(arg0); // Handle frames driver.switchTo().frame(int Frame Index); driver.switchTo().frame(iframeName"); WebElement element = driver.FindElement(By.ElementLocator("Value of Element Locator")); driver.switchTo().frame(element); driver.switchTo().frame(element); driver.switchTo().defaultContent(); Screenshots Capture TakesScreenshot screenshot =((TakesScreenshot)driver); File srcFile= screenshot.getScreenshotAs(OutputType.FiLE); FileHandler.copy(srcFile, destFile); Manage Timeouts driver.manage().timeouts().implicitlyWait(10, TimeUnit.SECONDS); welement = wait.until(Syntax: WebDriverWait wait = new WebDriverWait(driver, timeout); ExpectedConditions.elementToBeClickable(locator)); welement.click(); Thread.sleep(Long milli-seconds) driver.manage().timeouts().pageLoadTimeout(30, TimeUnit.SECONDS); Scroll Down or Up Web Page JavascriptExecutor js = (JavascriptExecutor)driver; js.executeScript("window.scrollBy(0,100)"); js.executeScript("window.scrollBy(0,1	driver.manage().getCookieNamed(arg0);		
driver.manage().deleteCookieNamed(arg0); // Handle frames driver.switchTo().frame(int Frame Index); driver.switchTo().frame(int Frame Index); driver.switchTo().frame(int Frame Index); driver.switchTo().frame(int Frame Index); driver.findElement(By.ElementLocator("Value of Element Locator")); driver.switchTo().defaultContent(); Screenshot Screenshot = ((TakesScreenshot)driver); File srcFile= screenshot screenshot = ((TakesScreenshot)driver); File srcFile= screenshot.getScreenshotAs(OutputType.FILE); FileHandler.copy(srcFile, destFile); Manage Timeouts driver.manage().timeouts().implicitlyWait(10, TimeUnit.SECONDS); welement = wait.until(Syntax: WebDriverWait wait = new WebDriverWait(driver, timeout); ExpectedConditions.elementToBeClickable(locator)); welement.click(); Thread.sleep(Long milli-seconds) driver.manage().timeouts().pageLoadTimeout(30, TimeUnit.SECONDS); Scroll Down or Up Web Page JavascriptExecutor is = (JavascriptExecutor)driver; is.executeScript("window.scrollBy(0,100)"); is.executeScript("window.scrollTo(0, document.body.scrollHeight)"); WebElement element = driver.FindElement(By.ElementLocator("Value of Element Locator")); is. executeScript("arguments[0].scrollIntoView()", element); Different approach to create XPath and CSS Selector			
// Handle frames driver.switchTo().frame(int Frame Index); driver.switchTo().frame("frameName"); WebElement element = driver.FindElement(By.ElementLocator("Value of Element Locator")); driver.switchTo().frame(element); driver.switchTo().defaultContent(); Screenshots Capture TakesScreenshot screenshot =((TakesScreenshot)driver); File srcFile= screenshot.getScreenshotAs(OutputType.FiLE); FileHandler.copy(srcFile, destFile); Manage Timeouts driver.manage().timeouts().implicitlyWait(10, TimeUnit.SECONDS); welement = wait.until(Syntax: WebDriverWait wait = new WebDriverWait(driver, timeout); ExpectedConditions.elementToBeclickable(locator)); welement.click(); Thread.sleep(Long milli-seconds) driver.manage().timeouts().pageLoadTimeout(30, TimeUnit.SECONDS); Scroll Down or Up Web Page JavascriptExecutor js = (JavascriptExecutor)driver; js.executeScript("window.scrollBv(0,100)"); js.executeScript("window.scrollBv(0,100)"); js.executeScript("window.scrollTo(0, document.body.scrollHeight)"); WebElement element = driver.FindElement(By.ElementLocator("Value of Element Locator")); js. executeScript("arguments[0].scrollIntoView()", element); Different approach to create XPath and CSS Selector Description XPath			
driver.switchTo().frame("frameName"); WebElement element = driver.FindElement(By.ElementLocator("Value of Element Locator")); driver.switchTo().frame(element); driver.switchTo().defaultContent(); Screenshots Capture TakesScreenshot screenshot =((TakesScreenshot)driver); File srcFile= screenshot.getScreenshotAs(OutputType.FiLE); FileHandler.copy(srcFile, destFile); Manage Timeouts driver.manage().timeouts().implicitlyWait(10, TimeUnit.SECONDS); welement = wait.until(Syntax: WebDriverWait wait = new WebDriverWait(driver, timeout); ExpectedConditions.elementToBeClickable(locator)); welement.click(); Thread.sleep(Long milli-seconds) driver.manage().timeouts().pageLoadTimeout(30, TimeUnit.SECONDS); Scroll Down or Up Web Page JavascriptExecutor js = (JavascriptExecutor)driver; js.executeScript("window.scrollBy(0.100)"); js.executeScript("window.scrollBy(0.100)"); js.executeScript("window.scrollTo(0, document.body.scrollHeight)"); WebElement element = driver.FindElement(By.ElementLocator("Value of Element Locator")); js. executeScript("arguments[0].scrollIntoView()", element); Different approach to create XPath and CSS Selector			
driver.switchTo().frame("frameName"); WebElement element = driver.FindElement(By.ElementLocator("Value of Element Locator")); driver.switchTo().frame(element); driver.switchTo().defaultContent(); Screenshots Capture TakesScreenshot screenshot =((TakesScreenshot)driver); File srcFile= screenshot.getScreenshotAs(OutputType.FiLE); FileHandler.copy(srcFile, destFile); Manage Timeouts driver.manage().timeouts().implicitlyWait(10, TimeUnit.SECONDS); welement = wait.until(Syntax: WebDriverWait wait = new WebDriverWait(driver, timeout); ExpectedConditions.elementToBeClickable(locator)); welement.click(); Thread.sleep(Long milli-seconds) driver.manage().timeouts().pageLoadTimeout(30, TimeUnit.SECONDS); Scroll Down or Up Web Page JavascriptExecutor js = (JavascriptExecutor)driver; js.executeScript("window.scrollBy(0.100)"); js.executeScript("window.scrollBy(0.100)"); js.executeScript("window.scrollTo(0, document.body.scrollHeight)"); WebElement element = driver.FindElement(By.ElementLocator("Value of Element Locator")); js. executeScript("arguments[0].scrollIntoView()", element); Different approach to create XPath and CSS Selector	driver.switchTo().frame(int Frame Index):		
WebElement element = driver.FindElement(By.ElementLocator("Value of Element Locator")); driver.switchTo().frame(element); driver.switchTo().defaultContent(); Screenshots Capture TakesScreenshot screenshot =((TakesScreenshot)driver); File srcFile= screenshot.getScreenshotAs(OutputType.FiLE); FileHandler.copy(srcFile, destFile); Manage Timeouts driver.manage().timeouts().implicitlyWait(10, TimeUnit.SECONDS); welement = wait.until(Syntax: WebDriverWait wait = new WebDriverWait(driver, timeout); ExpectedConditions.elementToBeClickable(locator)); welement.click(); Thread.sleep(Long milli-seconds) driver.manage().timeouts().pageLoadTimeout(30, TimeUnit.SECONDS); Scroll Down or Up Web Page JavascriptExecutor js = (JavascriptExecutor)driver; js.executeScript("window.scrollBy(0,100)"); js.executeScript("window.scrollBy(0,100)"); js.executeScript("window.scrollBy(0,100)"); js.executeScript("sindelement(By.ElementLocator("Value of Element driver.FindElement(By.ElementLocator("Value of Element Locator")); js. executeScript("arguments[0].scrollIntoView()", element); Different approach to create XPath and CSS Selector			
driver.FindElement(By.ElementLocator("Value of Element Locator")); driver.switchTo().frame(element); driver.switchTo().defaultContent(); Screenshots Capture TakesScreenshot screenshot =((TakesScreenshot)driver); File srcFile= screenshot.getScreenshotAs(OutputType.FiLE); FileHandler.copy(srcFile, destFile); Manage Timeouts driver.manage().timeouts().implicitlyWait(10, TimeUnit.SECONDS); welement = wait.until(Syntax: WebDriverWait wait = new WebDriverWait(driver, timeout); ExpectedConditions.elementToBeClickable(locator)); welement.click(); Thread.sleep(Long milli-seconds) driver.manage().timeouts().pageLoadTimeout(30, TimeUnit.SECONDS); Scroll Down or Up Web Page JavascriptExecutor js = (JavascriptExecutor)driver; js.executeScript("window.scrollBy(0,100)"); js.executeScript("window.scrollBy(0,100)"); js.executeScript("window.scrollBy(0,100)"); js.executeScript("window.scrollBy(0,100)"); js.executeScript("window.scrollBy(0,100)"); js.executeScript("window.scrollBy(0,100)"); js.executeScript("window.scrollBy(0,100)"); js.executeScript("window.scrollBy(0,100)"); js.executeScript("arguments[0].scrollIntoView()", element); Different approach to create XPath and CSS Selector			
Locator")); driver.switchTo().frame(element); driver.switchTo().defaultContent(); Screenshots Capture TakesScreenshot screenshot =((TakesScreenshot)driver); File srcFile= screenshot.getScreenshotAs(OutputType.FiLE); FileHandler.copy(srcFile, destFile); Manage Timeouts driver.manage().timeouts().implicitlyWait(10, TimeUnit.SECONDS); welement = wait.until(Syntax: WebDriverWait wait = new WebDriverWait(driver, timeout); ExpectedConditions.elementToBeClickable(locator)); welement.click(); Thread.sleep(Long milli-seconds) driver.manage().timeouts().pageLoadTimeout(30, TimeUnit.SECONDS); Scroll Down or Up Web Page JavascriptExecutor js = (JavascriptExecutor)driver; js.executeScript("window.scrollBy(0,100)"); js.executeScript("window.scrollBy(0,100)"); js.executeScript("window.scrollTo(0, document.body.scrolHeight)"); WebElement element = driver.FindElement(By.ElementLocator("Value of Element Locator")); js. executeScript("arguments[0].scrollintoView()", element); Different approach to create XPath and CSS Selector Description XPath CSS Selector			
driver.switchTo().frame(element); driver.SwitchTo().defaultContent(); Screenshots Capture TakesScreenshot screenshot =((TakesScreenshot)driver); File srcFile= screenshot.getScreenshotAs(OutputType.FILE); FileHandler.copy(srcFile, destFile); Manage Timeouts driver.manage().timeouts().implicitlyWait(10, TimeUnit.SECONDS); welement = wait.until(Syntax: WebDriverWait wait = new WebDriverWait(driver, timeout); ExpectedConditions.elementToBeClickable(locator)); welement.click(); Thread.sleep(Long milli-seconds) driver.manage().timeouts().pageLoadTimeout(30, TimeUnit.SECONDS); Scroll Down or Up Web Page JavascriptExecutor js = (JavascriptExecutor)driver; js.executeScript("window.scrollBy(0,100)"); js.executeScript("window.scrollBy(0,100)"); js.executeScript("window.scrollBe(nod, document.body.scrollHeight)"); WebElement element = driver.FindElement(By.ElementLocator("Value of Element Locator")); js. executeScript("arguments[0].scrolllintoView()", element); Different approach to create XPath and CSS Selector Description XPath CSS Selector			
driver.SwitchTo().defaultContent(); Screenshots Capture TakesScreenshot screenshot = ((TakesScreenshot)driver); File srcFile= screenshot.getScreenshotAs(OutputType.FiLE); File srcFile= screenshot.getScreenshotAs(OutputType.FiLE); FileHandler.copy(srcFile, destFile); Manage Timeouts driver.manage().timeouts().implicitlyWait(10,			
Screenshots Capture TakesScreenshot screenshot =((TakesScreenshot)driver); File srcFile= screenshot.getScreenshotAs(OutputType.FILE); FileHandler.copy(srcFile, destFile); Manage Timeouts driver.manage().timeouts().implicitlyWait(10, TimeUnit.SECONDS); welement = wait.until(Syntax: WebDriverWait wait = new WebDriverWait(driver, timeout); ExpectedConditions.elementToBeClickable(locator)); welement.click(); Thread.sleep(Long milli-seconds) driver.manage().timeouts().pageLoadTimeout(30, TimeUnit.SECONDS); Scroll Down or Up Web Page JavascriptExecutor js = (JavascriptExecutor)driver; js.executeScript("window.scrollBy(0,100)"); js.executeScript("window.scrollBy(0,100)"); js.executeScript("window.scrollHeight)"); WebElement element = driver.FindElement(By.ElementLocator("Value of Element Locator")); js. executeScript("arguments[0].scrollIntoView()", element); Different approach to create XPath and CSS Selector			
TakesScreenshot screenshot = ((TakesScreenshot)driver); File screile= screenshot.getScreenshotAs(OutputType.FILE); FileHandler.copy(srcFile, destFile); Manage Timeouts driver.manage().timeouts().implicitlyWait(10, TimeUnit.SECONDS); welement = wait.until(Syntax: WebDriverWait wait = new WebDriverWait(driver, timeout); ExpectedConditions.elementToBeClickable(locator)); welement.click(); Thread.sleep(Long milli-seconds) driver.manage().timeouts().pageLoadTimeout(30, TimeUnit.SECONDS); Scroll Down or Up Web Page JavascriptExecutor js = (JavascriptExecutor)driver; js.executeScript("window.scrollBy(0,100)"); js.executeScript("window.scrollTo(0, document.body.scrollHeight)"); WebElement element = driver.FindElement(By.ElementLocator("Value of Element Locator")); js. executeScript("arguments[0].scrollIntoView()", element); Different approach to create XPath and CSS Selector			
File srcFile= screenshot.getScreenshotAs(OutputType.FILE); FileHandler.copy(srcFile, destFile); Manage Timeouts driver.manage().timeouts().implicitlyWait(10,	·		
Manage Timeouts driver.manage().timeouts().implicitlyWait(10, TimeUnit.SECONDS); welement = wait.until(Syntax: WebDriverWait wait = new WebDriverWait(driver, timeout); ExpectedConditions.elementToBeClickable(locator)); welement.click(); Thread.sleep(Long milli-seconds) driver.manage().timeouts().pageLoadTimeout(30, TimeUnit.SECONDS); Scroll Down or Up Web Page JavascriptExecutor js = (JavascriptExecutor)driver; js.executeScript("window.scrollBy(0,100)"); js.executeScript("window.scrollTo(0, document.body.scrollHeight)"); WebElement element = driver.FindElement(By.ElementLocator("Value of Element Locator")); js. executeScript("arguments[0].scrollIntoView()", element); Different approach to create XPath and CSS Selector Description XPath			
Manage Timeouts driver.manage().timeouts().implicitlyWait(10, TimeUnit.SECONDS); welement = wait.until(Syntax: WebDriverWait wait = new WebDriverWait(driver, timeout); ExpectedConditions.elementToBeClickable(locator)); welement.click(); Thread.sleep(Long milli-seconds) driver.manage().timeouts().pageLoadTimeout(30, TimeUnit.SECONDS); Scroll Down or Up Web Page JavascriptExecutor js = (JavascriptExecutor)driver; js.executeScript("window.scrollBy(0,100)"); js.executeScript("window.scrollTo(0, document.body.scrollHeight)"); WebElement element = driver.FindElement(By.ElementLocator("Value of Element Locator")); js. executeScript("arguments[0].scrollIntoView()", element); Different approach to create XPath and CSS Selector Description XPath	FileHandler conv(srcFile_destFile):		
driver.manage().timeouts().implicitlyWait(10, TimeUnit.SECONDS); welement = wait.until(Syntax: WebDriverWait wait = new WebDriverWait(driver, timeout); ExpectedConditions.elementToBeClickable(locator)); welement.click(); Thread.sleep(Long milli-seconds) driver.manage().timeouts().pageLoadTimeout(30, TimeUnit.SECONDS); Scroll Down or Up Web Page JavascriptExecutor js = (JavascriptExecutor)driver; js.executeScript("window.scrollBy(0,100)"); js.executeScript("window.scrollTo(0, document.body.scrollHeight)"); WebElement element = driver.FindElement(By.ElementLocator("Value of Element Locator")); js. executeScript("arguments[0].scrollIntoView()", element); Different approach to create XPath and CSS Selector Description XPath CSS Selector			
TimeUnit.SECONDS); welement = wait.until(Syntax: WebDriverWait wait = new WebDriverWait(driver, timeout); ExpectedConditions.elementToBeClickable(locator)); welement.click(); Thread.sleep(Long milli-seconds) driver.manage().timeouts().pageLoadTimeout(30, TimeUnit.SECONDS); Scroll Down or Up Web Page JavascriptExecutor js = (JavascriptExecutor)driver; js.executeScript("window.scrollBy(0,100)"); js.executeScript("window.scrollTo(0, document.body.scrollHeight)"); WebElement element = driver.FindElement(By.ElementLocator("Value of Element Locator")); js. executeScript("arguments[0].scrollIntoView()", element); Different approach to create XPath and CSS Selector Description XPath CSS Selector			
welement = wait.until(Syntax: WebDriverWait wait = new WebDriverWait(driver, timeout); ExpectedConditions.elementToBeClickable(locator)); welement.click(); Thread.sleep(Long milli-seconds) driver.manage().timeouts().pageLoadTimeout(30, TimeUnit.SECONDS); Scroll Down or Up Web Page JavascriptExecutor js = (JavascriptExecutor)driver; js.executeScript("window.scrollBy(0,100)"); js.executeScript("window.scrollTo(0, document.body.scrollHeight)"); WebElement element = driver.FindElement(By.ElementLocator("Value of Element Locator")); js. executeScript("arguments[0].scrollIntoView()", element); Different approach to create XPath and CSS Selector Description XPath CSS Selector			
WebDriverWait(driver, timeout); ExpectedConditions.elementToBeClickable(locator)); welement.click(); Thread.sleep(Long milli-seconds) driver.manage().timeouts().pageLoadTimeout(30, TimeUnit.SECONDS); Scroll Down or Up Web Page JavascriptExecutor js = (JavascriptExecutor)driver; js.executeScript("window.scrollBy(0,100)"); js.executeScript("window.scrollTo(0, document.body.scrollHeight)"); WebElement element = driver.FindElement(By.ElementLocator("Value of Element Locator")); js. executeScript("arguments[0].scrollIntoView()", element); Different approach to create XPath and CSS Selector Description XPath CSS Selector	<i>r</i> .		
ExpectedConditions.elementToBeClickable(locator)); welement.click(); Thread.sleep(Long milli-seconds) driver.manage().timeouts().pageLoadTimeout(30, TimeUnit.SECONDS); Scroll Down or Up Web Page JavascriptExecutor js = (JavascriptExecutor)driver; js.executeScript("window.scrollBy(0,100)"); js.executeScript("window.scrollTo(0, document.body.scrollHeight)"); WebElement element = driver.FindElement(By.ElementLocator("Value of Element Locator")); js. executeScript("arguments[0].scrollIntoView()", element); Different approach to create XPath and CSS Selector Description XPath CSS Selector			
welement.click(); Thread.sleep(Long milli-seconds) driver.manage().timeouts().pageLoadTimeout(30, TimeUnit.SECONDS); Scroll Down or Up Web Page JavascriptExecutor js = (JavascriptExecutor)driver; js.executeScript("window.scrollBy(0,100)"); js.executeScript("window.scrollTo(0, document.body.scrollHeight)"); WebElement element = driver.FindElement(By.ElementLocator("Value of Element Locator")); js. executeScript("arguments[0].scrollIntoView()", element); Different approach to create XPath and CSS Selector Description XPath CSS Selector	, , , , , , , , , , , , , , , , , , , ,		
Thread.sleep(Long milli-seconds) driver.manage().timeouts().pageLoadTimeout(30, TimeUnit.SECONDS); Scroll Down or Up Web Page JavascriptExecutor js = (JavascriptExecutor)driver; js.executeScript("window.scrollBy(0,100)"); js.executeScript("window.scrollTo(0, document.body.scrollHeight)"); WebElement element = driver.FindElement(By.ElementLocator("Value of Element Locator")); js. executeScript("arguments[0].scrollIntoView()", element); Different approach to create XPath and CSS Selector Description XPath CSS Selector			
driver.manage().timeouts().pageLoadTimeout(30, TimeUnit.SECONDS); Scroll Down or Up Web Page JavascriptExecutor js = (JavascriptExecutor)driver; js.executeScript("window.scrollBy(0,100)"); js.executeScript("window.scrollTo(0, document.body.scrollHeight)"); WebElement element = driver.FindElement(By.ElementLocator("Value of Element Locator")); js. executeScript("arguments[0].scrollIntoView()", element); Different approach to create XPath and CSS Selector Description XPath CSS Selector	·		
TimeUnit.SECONDS); Scroll Down or Up Web Page JavascriptExecutor js = (JavascriptExecutor)driver; js.executeScript("window.scrollBy(0,100)"); js.executeScript("window.scrollTo(0,			
Scroll Down or Up Web Page JavascriptExecutor js = (JavascriptExecutor)driver; js.executeScript("window.scrollBy(0,100)"); js.executeScript("window.scrollTo(0,			
JavascriptExecutor js = (JavascriptExecutor)driver; js.executeScript("window.scrollBy(0,100)"); js.executeScript("window.scrollTo(0,			
js.executeScript("window.scrollBy(0,100)"); js.executeScript("window.scrollTo(0, document.body.scrollHeight)"); WebElement element = driver.FindElement(By.ElementLocator("Value of Element Locator")); js. executeScript("arguments[0].scrollIntoView()", element); Different approach to create XPath and CSS Selector Description XPath CSS Selector	·		
js.executeScript("window.scrollTo(0, document.body.scrollHeight)"); WebElement element = driver.FindElement(By.ElementLocator("Value of Element Locator")); js. executeScript("arguments[0].scrollIntoView()", element); Different approach to create XPath and CSS Selector Description XPath CSS Selector			
document.body.scrollHeight)"); WebElement element = driver.FindElement(By.ElementLocator("Value of Element Locator")); js. executeScript("arguments[0].scrollIntoView()", element); Different approach to create XPath and CSS Selector Description XPath CSS Selector			
WebElement element = driver.FindElement(By.ElementLocator("Value of Element Locator")); js. executeScript("arguments[0].scrollIntoView()", element); Different approach to create XPath and CSS Selector Description XPath CSS Selector			
driver.FindElement(By.ElementLocator("Value of Element Locator")); js. executeScript("arguments[0].scrollIntoView()", element); Different approach to create XPath and CSS Selector Description XPath CSS Selector			
Locator")); js. executeScript("arguments[0].scrollIntoView()", element); Different approach to create XPath and CSS Selector Description XPath CSS Selector			
js. executeScript("arguments[0].scrollIntoView()", element); Different approach to create XPath and CSS Selector Description XPath CSS Selector			
Different approach to create XPath and CSS Selector Description XPath CSS Selector	777		
Description XPath CSS Selector	js. executescript(arguments[v].scrollintoview()", element);		
·	Different approach to create XPath and CSS Selector		
Whole WebPage /html html	Description	XPath	CSS Selector
	Whole WebPage	/html	html

Whole WebPage body	/html/body	body
image element	//img	img
Link	//a[@href = 'url']	a[href = 'url']
Direct Child	//div/a	div > a
Id	//tagName[@id='idValue']	tagName#idValue
Class	//tagName[@class='classValu	tagName.Value of Class
	e'l	attribute
Attribute	//tagname[@attribute-	tagName[attribute=Valu
	name='value1']	e of attribute]
Multiple Attributes	//input[@type='submit' and	tagname[attribute1='val
·	@name='btnLogin']	ue1'][attribute2='value2'
]
Contains	//*[contains(@type,'sub')]	<html< td=""></html<>
		tag><[attribute*=sub
		String]>
Starts with	//tagname[starts-	<html< td=""></html<>
	with(@attribute, 'Start	tag><[attribute^=prefix
	value')]	of the String]>
Ends with	//tagname[ends-	<html< td=""></html<>
	with(@attribute, 'End value')]	tag><[attribute\$=suffix
	(6111 1111,	of the String]>
Matches	//*[matches(@type,'value')]	N/A
	,, , , , , , , , , , , , , , , , , , , ,	,
First Child	//ul[@id='list']/li[1]	ul#list li:first-child
Last Child	//ul[@id='list']/li[last()]	ul#list li:last-child
nth Child	//ul[@id='list']/li[3]	ul#list li:nth-child(3)
Text Value	//td[text()='textname']	N/A
Element preceding some sibling	//E2/preceding-sibling::E1	N/A
Sibling element immediately preceding	//E/preceding-sibling::*[1]	N/A
User interface element that is disabled	//E[@disabled]	E:disabled
Checkbox (or radio button) that is checked	//*[@checked]	*:checked
Text Value	//td[text()='textname']	N/A
Important Tagname and their Description		
Tag Name	Specification	
	Defines an unordered list of	
	items.	
	Defines a row of cells in a	
	table.	
<title></td><td>Represents title to an HTML</td><td></td></tr><tr><td></td><td>document.</td><td></td></tr><tr><td>></td><td>Used for creates header of a</td><td></td></tr><tr><td></td><td>group of cell in HTML table.</td><td></td></tr><tr><td></td><td></td><td></td></tr><tr><td></td><td>Used to defines a table in an</td><td></td></tr><tr><td></td><td>HTML document.</td><td></td></tr><tr><td></td><td>Used for grouping table rows.</td><td></td></tr><tr><td>></td><td>Used for creates standard</td><td></td></tr><tr><td></td><td>data cell in HTML table.</td><td></td></tr><tr><td rowspan=3></td><td>Used to grouping and</td><td></td></tr><tr><td>applying styles to inline</td><td></td></tr><tr><td>elements.</td><td></td></tr><tr><td></td><td>elements.</td><td></td></tr></tbody></table></title>		

<section></section>	Used to divide a document
	into number of different
	generic section.
<select></select>	Used to create a drop-down
	list.
<menu></menu>	Used to display a unordered
	list of items/menu of
	commands.
	Define a list item either
	ordered list or unordered list.
<iframe></iframe>	Defines a inline frame that
	embedded external content
	into current web document.
	Used to insert image into a
	web document.
<input/>	Define a get information in
	selected input
<div></div>	Define a division part
<body></body>	Defines a main section(body)
	part in HTML document
 	Specific a single line break
<button></button>	Specifies a press/push button
<a>	Specific a anchor (Hyperlink)