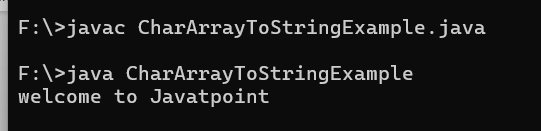
STEPS TO RUN A JAVA FILE USING NOTEPAD

**Save the file with the name**

**CharArrayToStringExample.java**

1. **public** **class** CharArrayToStringExample
2. {
3. **public** **static** **void** main(String args[])
4. {
5. //character array
6. **char**[] ch = {'w', 'e', 'l', 'c', 'o', 'm', 'e', ' ' , 't', 'o', ' ', 'J', 'a', 'v', 'a', 't', 'p', 'o', 'i', 'n', 't'};
7. //constructor of the String class that parses char array as a parameter
8. String string = **new** String(ch);
9. //prints the string
10. System.out.println(string);
11. }
12. }



Actions class

**driver.get() :** It's used to go to the particular website , But it doesn't maintain the browser History and cookies so , we can't use forward and backward button , if we click on that , page will not get schedule

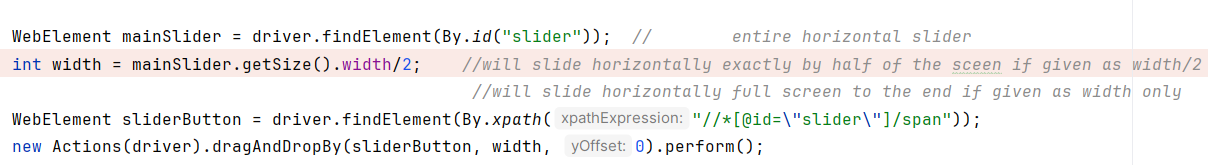
**driver.navigate() :** it's used to go to the particular website , but it maintains the browser history and cookies, so we can use forward and backward button to navigate between the pages during the coding of Testcase

In **Selenium WebDriver**for the closing browser session, there are two WebDriver commands driver.quit() and driver.close(). And both have different objective **close()** command closes the browser window which is currently active and **quit()** command closes all the browser windows and terminate a WebDriver session.

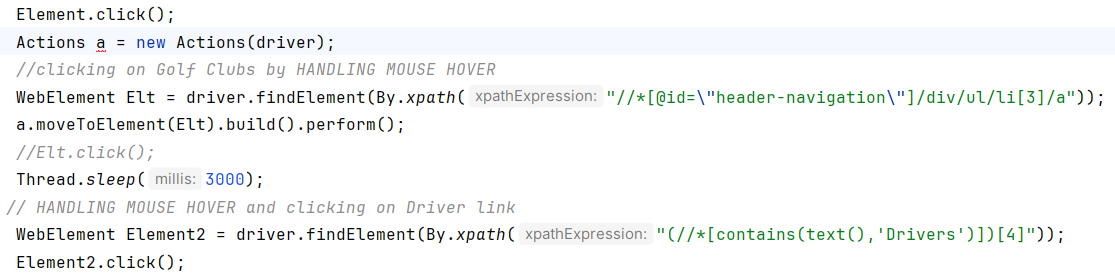
========================================================================

Horizontal Slider

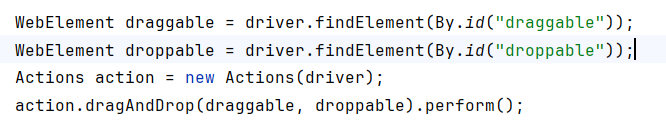
URL = driver.get("https://jqueryui.com/resources/demos/slider/default.html");



MOUSE HOVER



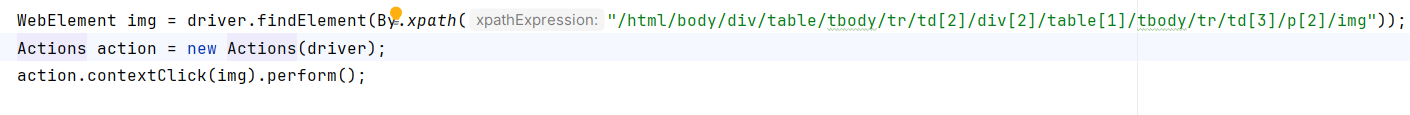
DRAG AND DROP

url = [https://jqueryui.com/resources/demos/droppable/default.html](https://jqueryui.com/resources/demos/droppable/default.html)

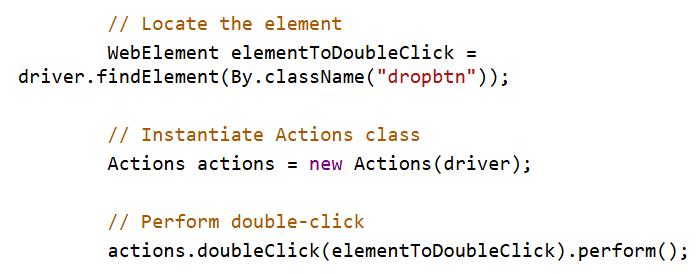
RESIZE



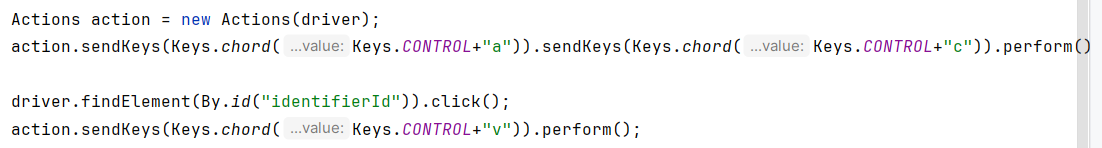
RIGHT CLICK



DOUBLE CLICK

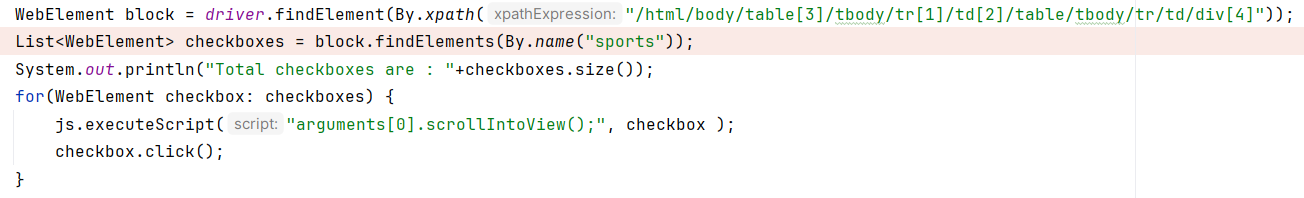


SELECT, COPY, and PASTE

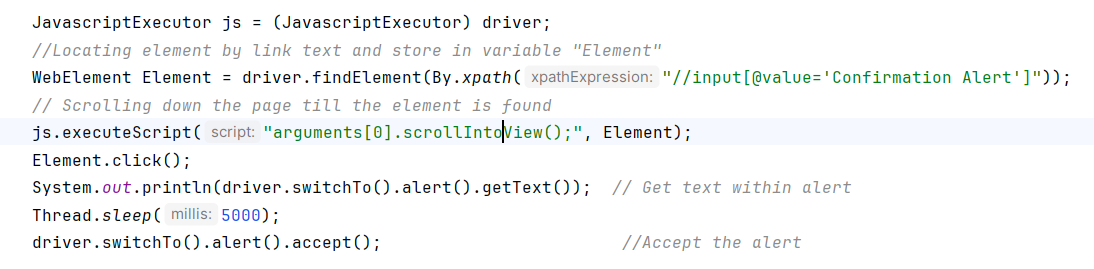


NOT INTERACTABLE CHECKBOXES, need to use javascriptexecutor

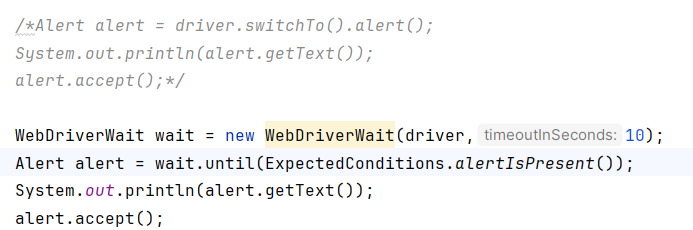
*driver*.get("http://www.tizag.com/htmlT/htmlcheckboxes.php");



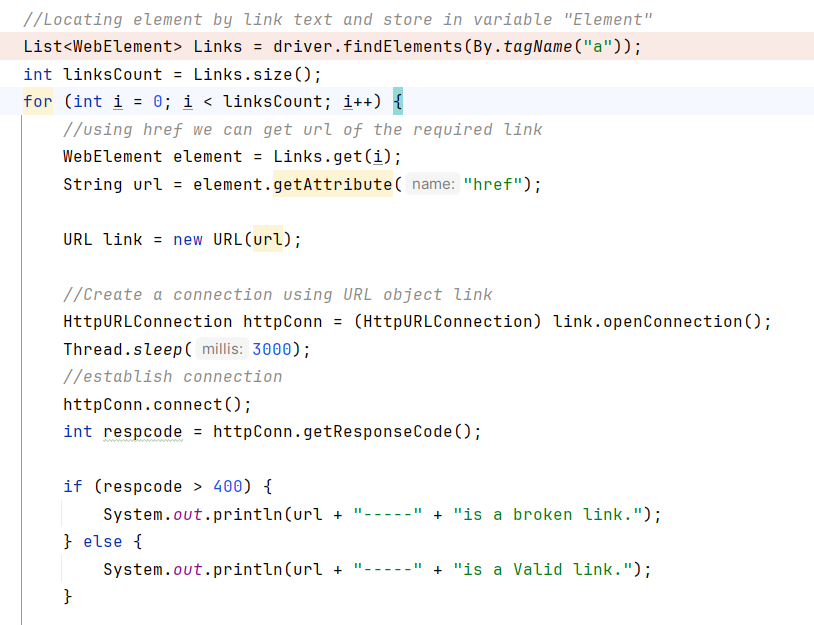
ScrollIntoView



ALERTS

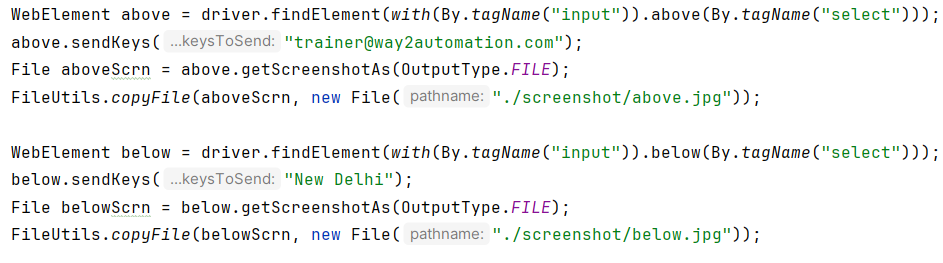


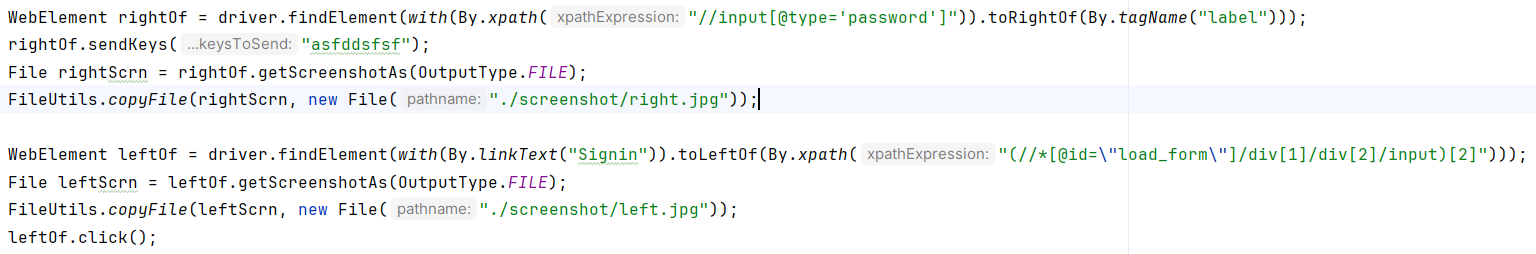
BROKEN LINKS



RELATIVE LOCATORS :

driver.get("https://www.way2automation.com/way2auto\_jquery/index.php");





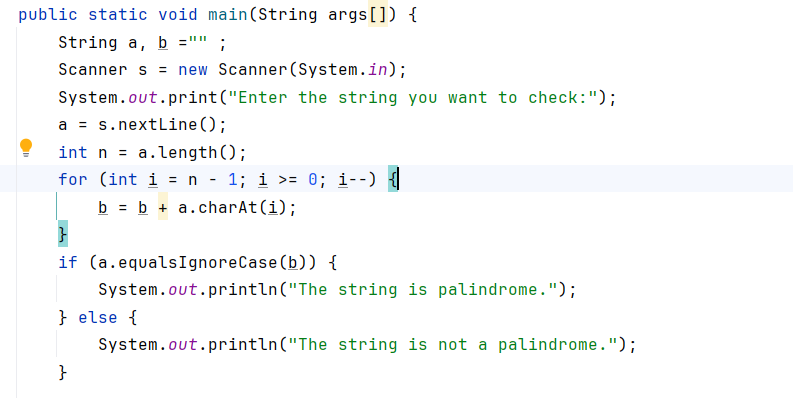


WEBTABLE

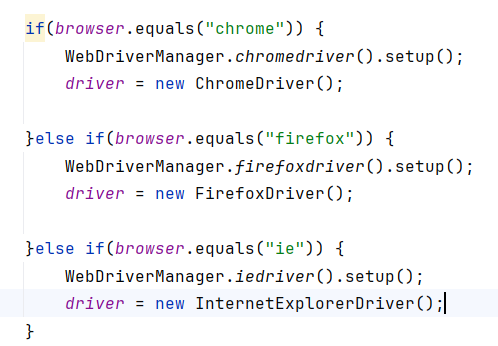
driver.get("https://money.rediff.com/gainers/bsc/daily/groupa");



PALLINDROME



WEBDRIVER MANAGER

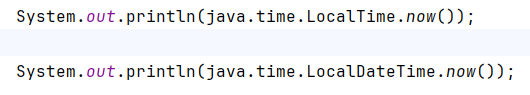


SELENIUM MANAGER:

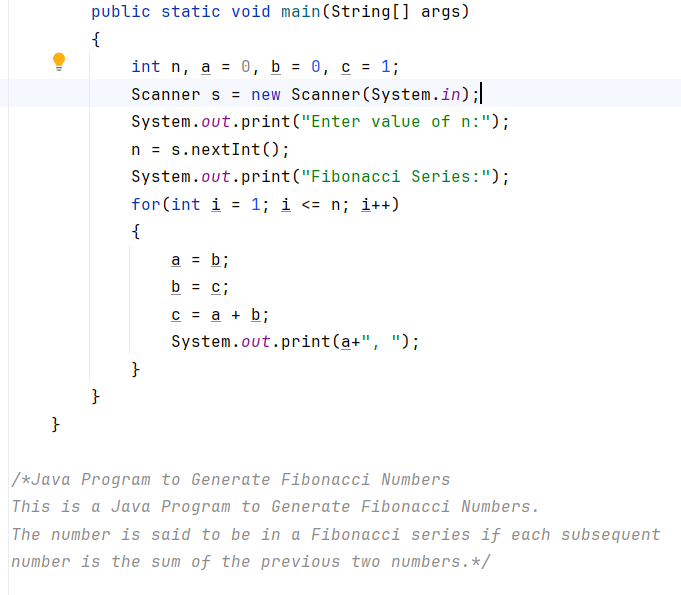
Now need to mention the path of the drivers (chrome, firefox, etc) just need to import the below class

Import org.openqa.selenium.manager.SeleniumManager;

DATE AND TIME



FIBONACCI SERIES

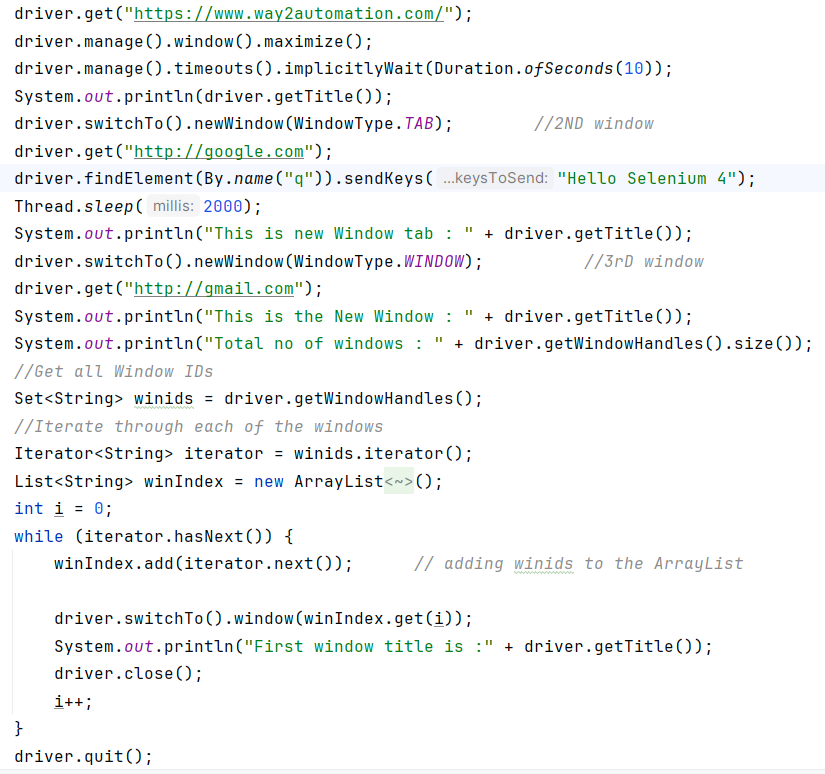


SCREEN COORDINATES

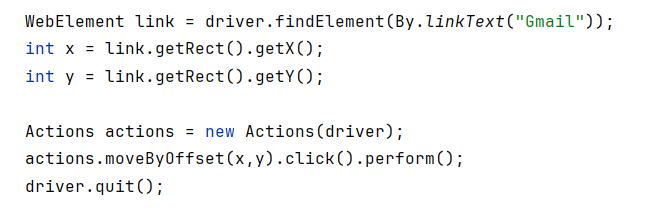


SWITCH TO WINDOWS

driver.get("https://www.way2automation.com/");



CLICKING ON AN OBJECT USING X, Y COORDINATES



JDBC

Selenium Webdriver is limited to[Testing](https://www.guru99.com/software-testing.html)your applications using Browser. To use Selenium Webdriver for Database Verification you need to use the JDBC ("Java Database Connectivity").

JDBC (Java Database Connectivity) is a[SQL](https://www.guru99.com/sql.html)level API that allows you to execute SQL statements. It is responsible for the connectivity between the[Java](https://www.guru99.com/java-tutorial.html)Programming language and a wide range of databases. The JDBC API provides the following classes and interfaces

* Driver Manager
* Driver
* Connection
* Statement
* ResultSet
* SQLException
* Package htmldriver;
* import java.sql.Connection;
* import java.sql.Statement;
* import java.sql.ResultSet;
* import java.sql.DriverManager;
* import java.sql.SQLException;
* public class SQLConnector {
* public static void main(String[] args) throws ClassNotFoundException, SQLException {
* //Connection URL Syntax: "jdbc:mysql://ipaddress:portnumber/db\_name"
* String dbUrl = "jdbc:mysql://localhost:3036/emp";
* //Database Username
* String username = "root";
* //Database Password
* String password = "guru99";
* //Query to Execute
* String query = "select \* from employee;";
* //Load mysql jdbc driver
* Class.forName("com.mysql.jdbc.Driver");
* //Create Connection to DB
* Connection con = DriverManager.getConnection(dbUrl,username,password);
* //Create Statement Object
* Statement stmt = con.createStatement();
* // Execute the SQL Query. Store results in ResultSet
* ResultSet rs= stmt.executeQuery(query);
* // While Loop to iterate through all data and print results
* while (rs.next()){
* String myName = rs.getString(1);
* String myAge = rs.getString(2);
* System. out.println(myName+" "+myAge);
* }
* // closing DB Connection
* con.close();
* }
* }

DATAPROVIDER

public class getExcelData {  
  
 @DataProvider(name= "getXLData")  
 public Object[][] getXLData() throws IOException{  
   
 String filepath= System.*getProperty*("user.dir")+"\\src\\test\\java\\com\\Pat\\TestData\\LoginData.xlsx";  
   
 *//FileInputStream fis= new FileInputStream("D:\\MyWorkspace\\pat\_Auto\\src\\test\\java\\com\\Pat\\TestData\\LoginData.xlsx");* FileInputStream fis= new FileInputStream(filepath);  
   
 XSSFWorkbook wb= new XSSFWorkbook(fis);  
 XSSFSheet ws=wb.getSheet("Sheet1");  
 int totalRows= ws.getLastRowNum();  
 int totalColumns= ws.getRow(0).getPhysicalNumberOfCells();  
   
 Object obj[][]= new Object[totalRows][totalColumns];  
 Hashtable<String,String> table = null;

for(int i=0; i<totalRows; i++) {

table = new Hashtable<String,String>();

for (int j = 0; j < totalColumns; j++) {  
  
 obj[i][j] = ws.getRow(i + 1).getCell(j).toString();  
 *//obj[i][1] = ws.getRow(i + 1).getCell(1).toString();  
 //obj[i][2] = ws.getRow(i + 1).getCell(2).toString();  
 //obj[i][j] = ws.getRow(i + 1).getCell(j).*

*//* System.*out*.println(obj[i][j]);

*//table.put(excel.getCellData(sheetName, colNum, 1), excel.getCellData(sheetName, //colNum, rowNum));*

*table.put( ws.getRow(1).getCell(j).toString, ws.getRow(i + 1).getCell(j).toString();*

*data[*totalRows *- 2][0] = table;*  
 }  
 }  
 return obj;  
 }  
  
}

@DataProvider(name="dp")  
public Object[][] getData(Method m) {  
  
 String sheetName = m.getName();  
 int rows = *excel*.getRowCount(sheetName);  
 int cols = *excel*.getColumnCount(sheetName);  
  
 Object[][] data = new Object[rows - 1][1];  
   
 Hashtable<String,String> table = null;  
  
 for (int rowNum = 2; rowNum <= rows; rowNum++) { *// 2* table = new Hashtable<String,String>();  
   
 for (int colNum = 0; colNum < cols; colNum++) {  
  
 *// data[0][0]* table.put(*excel*.getCellData(sheetName, colNum, 1), *excel*.getCellData(sheetName, colNum, rowNum));  
 data[rowNum - 2][0] = table;  
 }  
  
 }  
  
 return data;  
  
}

public class AlertHandlingWithoutSwitchTo {  
  
 public static void main(String[] args) throws InterruptedException {  
 *// WebDriverManager.chromedriver().setup();* WebDriver driver = new ChromeDriver();  
 driver.get("https://testautomationpractice.blogspot.com/");  
 driver.manage().window().maximize();  
 *// driver.findElement(By.id("alertBtn")).click();//opens the alertbox* Thread.*sleep*(3000);  
 */\* //1 SwitchTo() method  
 Alert alert = driver.switchTo().alert(); alert.accept();  
\*/  
  
 /\* // 2 Explicit wait method  
 WebDriverWait wait = new WebDriverWait(driver, Duration.ofSeconds(10));  
 Alert alert = wait.until(ExpectedConditions.alertIsPresent());  
 alert.accept();  
 String alertMsg = alert.getText();\*/  
  
/\*// 3 javascriptexecutor method  
 JavascriptExecutor js = (JavascriptExecutor) driver;  
 try { // to handle the exception  
 js.executeScript("window.alert=function{};");  
 }catch (Exception e)  
 {  
 System.out.println(e.getMessage());  
 }\*/  
  
 //Prompt Alert* driver.findElement(By.*id*("promptBtn")).click();  
 *//WebDriverWait wait = new WebDriverWait(driver, Duration.ofSeconds(10));  
 // Alert alert = wait.until(ExpectedConditions.alertIsPresent());\*/* Alert alert = driver.switchTo().alert();  
 alert.sendKeys("Subash Patil");  
 Thread.*sleep*(8000);  
 alert.dismiss();  
 }  
}

public class ReverseEachWordOfString {  
 public static void main(String[] args) {  
 String str = "I am learning Java";  
 String[] words = str.split(" ");  
  
 String reverseString = "";  
 List<String> strWords = new ArrayList<String>();  
  
 for (int i = words.length-1 ; i>=0; i--) {  
 strWords.add(words[i]);  
 }  
 System.*out*.println(strWords);  
 }  
}

public class AllWaitsApplied {  
  
 public static void main(String[] args) throws InterruptedException {  
 */\* WebDriverManager.firefoxdriver().setup();  
 WebDriver driver = new FirefoxDriver();\*/  
 //WebDriverManager.chromedriver().setup();* WebDriver driver= new ChromeDriver();  
 driver.get("http://gmail.com");  
 driver.manage().window().maximize();  
 driver.manage().timeouts().implicitlyWait(Duration.*ofSeconds*(30));  
 *//WebDriverWait wait = new WebDriverWait(driver, Duration.ofSeconds(30));* Wait<WebDriver> wait = new FluentWait<WebDriver>(driver)  
 .withTimeout(Duration.*ofSeconds*(30))  
 .pollingEvery(Duration.*ofSeconds*(5))  
 .withMessage("Time out as the condition is not met")  
 .ignoring(NoSuchElementException.class);  
 *//driver.manage().window().minimize();* WebElement username = driver.findElement(By.*id*("identifierId"));  
 username.sendKeys("trainer@way2automation.com");  
 *//driver.manage().window().fullscreen();* driver.findElement(By.*xpath*("//\*[@id=\"identifierNext\"]/div/button/span")).click();  
 *//Thread.sleep(10000);  
 //driver.findElement(By.xpath("//\*[@id=\"password\"]/div[1]/div/div[1]/input")).sendKeys("asdfsf");* wait.until(ExpectedConditions.*elementToBeClickable*(By.*xpath*("//\*[@id=\"password\"]/div[1]/div/div[1]/input"))).sendKeys("sadfdf");  
 driver.findElement(By.*xpath*("//\*[@id=\"passwordNext\"]/div/button")).click();  
  
 System.*out*.println(driver.findElement(By.*xpath*("//\*[@id=\"view\_container\"]/div/div/div[2]/div/div[1]/div/form/span/section/div/div/div[1]/div[2]/div[2]/span")).getText());  
 }  
 }

public class DateTime {  
 public static void main(String[] args) {  
 System.*out*.println(java.time.LocalDate.*now*());  
 System.*out*.println(java.time.LocalTime.*now*());  
 System.*out*.println(java.time.LocalDateTime.*now*());  
 }  
}

=========================================================================

*/\*Each number (every third number) is equal to the sum of the preceding two numbers.  
For example, 0, 1, 1, 2, 3, 5, 8, 13, 21\*/*public class Fibonacci {  
  
 public static void main(String[] args)  
 {  
 int n, a = 0, b = 0, c = 1;  
 Scanner s = new Scanner(System.*in*);  
 System.*out*.print("Enter value of n:");  
 n = s.nextInt();  
 System.*out*.print("Fibonacci Series:");  
 for(int i = 1; i <= n; i++)  
 {  
 a = b;  
 b = c;  
 c = a + b;  
 System.*out*.print(a+", ");  
 }  
 }  
 }  
*/\*The number is said to be in a Fibonacci series if each subsequent  
number is the sum of the previous two numbers.\*/*

While When the number of iterations are unknown  
For Number of iterations are known.  
Do while Will execute atleast once irrespetive of the condition.

*/\*double d = Math.random()\*100; math.random will return a number between 1 and 100  
 \*/*public class GenerateRandomNumbersFunction {  
  
 public static void main(String[] args) {  
 *GenerateRandomNumbers*(35);  
 }  
 public static void GenerateRandomNumbers(int n){  
 double d = Math.*random*()\*n;  
 int num = (int)d; *//this foribly converts d which is double to integer* System.*out*.println(d);  
 System.*out*.println(num);  
 }  
}

public class HandlingFrames {  
 public static void main(String[] args) throws InterruptedException {  
WebDriver driver = new ChromeDriver();  
  
 driver.get("https://www.paytm.com");  
 driver.manage().window().maximize();  
 Actions a = new Actions(driver);  
  
 driver.manage().timeouts().implicitlyWait(Duration.*ofSeconds*(5));  
  
 *//Locating element by link text and store in variable "Element"* WebElement Element = driver.findElement(By.*xpath*("//span[contains(text(),'Sign In')]"));  
 Element.click();  
Thread.*sleep*(2000);  
 JavascriptExecutor js = (JavascriptExecutor) driver;  
 *//use executeScript() method and pass the arguments  
 //Here i pass values based on css style. Yellow background color with solid red color border.  
  
 // WebElement iframe = driver.findElement(By.xpath("//\*[@id=\"app\"]/div[1]/div"));  
 // js.executeScript("arguments[0].setAttribute('style', 'background: yellow; border: 2px solid red;');", iframe);*List<WebElement> iframes =driver.findElements(By.*tagName*("iframe"));  
 System.*out*.println(iframes.size());  
 *//Switch to the frame* driver.switchTo().frame(iframes.get(0));  
 driver.switchTo().frame(0);  
  
 driver.findElement(By.*xpath*("//a[@class='\_1mvLksqBDbXvIRjc9YWvF x-link'][contains(text(),'privacy policy')]")).click();  
 Thread.*sleep*(2000);  
  
 *//2nd page*

WebElement Element2 = driver.findElement(By.*xpath*("//span[contains(text(),'Sign In')]"));  
 Element.click();  
 Thread.*sleep*(2000);  
 driver.switchTo().frame(iframes.get(1));  
  
 driver.findElement(By.*xpath*("//a[@class='\_1mvLksqBDbXvIRjc9YWvF x-link'][contains(text(),'privacy policy')]")).click();  
 Thread.*sleep*(2000);  
 }  
}

public class HandlingWindowFrames {  
 public static void main(String[] args) throws InterruptedException {  
 *//System.setProperty("webdriver.chrome.driver","./src/resources/chromedriver");* WebDriver driver = new ChromeDriver();  
 driver.manage().window().maximize();  
 driver.get("https://demoqa.com/browser-windows");  
  
 *// Open new window by clicking the button "New Window"* driver.findElement(By.*id*("windowButton")).click();  
  
 *// Click on the new window element and read the text displayed on the window  
 // WebElement text = driver.findElement(By.id("sampleHeading"));*Thread.*sleep*(3000);  
  
  
 String parentWindow = driver.getWindowHandle();  
 System.*out*.println(parentWindow);  
 Set<String> handles = driver.getWindowHandles();  
 for(String windowHandle : handles)  
 {  
 if(!windowHandle.equals(parentWindow))  
 {  
 driver.switchTo().window(windowHandle);  
  
 WebElement text2 = driver.findElement(By.*xpath*(" //h1[@id='sampleHeading']"));  
  
 *// Fetching the text using method and printing it* System.*out*.println("Element found using text: " + text2.getText());  
  
 driver.close(); *//closing child window* driver.switchTo().window(parentWindow); *//cntrl to parent window* }  
 }  
 driver.quit();  
 }  
 }

public class OnlyMainFunctionExecutes {  
  
 public static void main(String[] args) {  
 System.*out*.println("Inside Main");  
 }  
  
 public static void testMe(String[] args) {  
 System.*out*.println("Inside testMe");  
 }  
}

public class Pallindrome {  
 public static void main(String args[]) {  
 String a, b ="" ;  
 Scanner s = new Scanner(System.*in*);  
 System.*out*.print("Enter the string you want to check:");  
 a = s.nextLine();  
 int n = a.length();  
 for (int i = n - 1; i >= 0; i--) {  
 b = b + a.charAt(i);  
 }  
 if (a.equalsIgnoreCase(b)) {  
 System.*out*.println("The string is palindrome.");  
 } else {  
 System.*out*.println("The string is not a palindrome.");  
 }  
 }  
}

public class RectangleTest {  
  
 public static void main(String[] args) {  
 WebDriverManager.*chromedriver*().setup();  
 WebDriver driver = new ChromeDriver();  
 driver.get("https://www.google.com");  
 driver.manage().window().maximize();  
 driver.manage().timeouts().implicitlyWait(Duration.*ofSeconds*(10));  
  
 JavascriptExecutor js = (JavascriptExecutor) driver;  
  
 *//WebElement img = driver.findElement(By.xpath("//\*[@id=\"logo\"]"));* WebElement img = driver.findElement(By.*xpath*("//\*[@alt=\"Google\"]"));  
 *// Scrolling down the page till the element is found* js.executeScript("arguments[0].scrollIntoView();", img);  
  
 Rectangle rect = img.getRect();  
  
 System.*out*.println("Height : "+rect.getHeight());  
 System.*out*.println("Width : "+rect.getWidth());  
 System.*out*.println("X Coord : "+rect.getX());  
 System.*out*.println("Y Coord : "+rect.getY());  
 }  
 }

public class ReverseANumber {  
  
 public static void main(String[] args) {  
  
 *//METHOD 1::* Scanner s = new Scanner(System.*in*);  
 System.*out*.println("Enter the number to be reversed");  
 int num = s.nextInt();  
 int rev = 0;  
  
 while (num != 0) {  
 rev = rev \* 10 + num % 10;  
 num = num / 10;  
  
 }  
 System.*out*.println("The reversed number by 1st method is " + rev);  
  
 *// METHOD 2:* s = new Scanner(System.*in*);  
 System.*out*.println("Enter the number to be reversed");  
 int num2 = s.nextInt();  
  
 *// Using StringBuffer class* StringBuffer sb = new StringBuffer(String.*valueOf*(num2));  
 StringBuffer rev2 = sb.reverse();  
 System.*out*.println("The reversed number by String Buffer is " + rev2);  
  
 *//METHOD 3 :* s = new Scanner(System.*in*);  
 System.*out*.println("Enter the number to be reversed");  
 int num3 = s.nextInt();  
  
 StringBuilder sbl= new StringBuilder(String.*valueOf*(num3));  
 *// sbl.append(num3);* StringBuilder rev3= sbl.reverse();  
  
 System.*out*.println("The reversed number by String Builder is " + rev3);  
 }  
 }

public class ReverseEachCharOfString {  
 public static void main(String[] args) {  
 String str = "I am learning Java";  
*//// METHOD 1  
 /\* String reverseString = "";  
  
 for (int i= str.length()-1 ; i>=0; i--) {  
 reverseString= reverseString+str.charAt(i);  
  
 }  
 System.out.println(reverseString);  
 }\*/  
 //// METHOD 2* StringBuffer sb = new StringBuffer(str);  
 System.*out*.println(sb.reverse());  
  
 }  
}

public class ReverseEachWordOfString {  
 public static void main(String[] args) {  
 String str = "I am learning Java";  
 String[] words = str.split(" ");  
  
 String reverseString = "";  
 List<String> strWords = new ArrayList<String>();  
  
 for (int i = words.length-1 ; i>=0; i--) {  
 strWords.add(words[i]);  
 }  
 System.*out*.println(strWords);  
 }  
}

public class ReverseWordsInString {  
  
 public static void printWords(String str)  
 {  
 *// word variable to store word* String word;  
 *// making a string stream* StringTokenizer iss = new StringTokenizer(str);  
 *// Read and print each word.* while (iss.hasMoreTokens()) {  
 word = iss.nextToken();  
 System.*out*.print(  
 new StringBuilder(word).reverse().toString()  
 + " ");  
 }  
 }  
  
 public static void main(String[] args)  
 throws java.lang.Exception  
 {  
 String s = "GeeksforGeeks is good to learn";  
 *printWords*(s);  
 }  
}

public class StaleElementReferenceExceptionError {  
  
 public static void main(String [] args) throws InterruptedException  
 {  
 WebDriver driver;  
  
 driver = new ChromeDriver();  
 driver.manage().window().maximize();  
 driver.get("https://www.pavantestingtools.com/");  
  
 Thread.*sleep*(3000);  
 WebElement login=driver.findElement(By.*xpath*("//span[contains(text(),'Log In')]"));  
 login.click();  
  
 Thread.*sleep*(2000);  
 WebElement userName=driver.findElement(By.*xpath*("//input[@placeholder='E-mail address']"));  
 userName.sendKeys("subashtpatil@gmail.com");  
  
 WebElement pwd= driver.findElement(By.*xpath*("//input[@placeholder='Password']"));  
 pwd.sendKeys("test@123");  
 driver.navigate().refresh();  
 Thread.*sleep*(2000);  
 try {  
 userName.sendKeys("subashtpatil@gmail.com");  
 pwd.sendKeys("test@123");  
 throw new StaleElementReferenceException(null, null);  
 }  
 catch(StaleElementReferenceException e){  
 */\*userName=driver.findElement(By.xpath("//input[@placeholder='E-mail address']"));  
 userName.sendKeys("subashtpatil@gmail.com");  
 pwd= driver.findElement(By.xpath("//input[@placeholder='Password']"));  
 pwd.sendKeys("test@123");  
 driver.findElement(By.xpath("//div[@class='ui fluid large blue submit button']")).click();\*/* e.printStackTrace();  
 }  
 }  
 }

1). Static method cannot call a non-static method directly, we have to create an object  
 of the class to call non-static methods.  
2). U can call a non-static method from another non-static method directly without creating objects.  
3). From static main method u can directly call another static method.  
4). Non-static method can directly call a static method  
  
Static --------------- Static allowed  
Non-static ---------◊ Nonstatic, static (allowed)  
Static ----------------◊ from static methods non-static methods can be called only by creating an object of the class  
Non-static method can directly call a static method

public class SwitchToParentFrame {  
 public static void main(String[] args) {  
 WebDriverManager.*chromedriver*().setup();  
 WebDriver driver = new ChromeDriver();  
 driver.get("https://www.w3schools.com/jsref/tryit.asp?filename=tryjsref\_win\_frames2");  
 driver.manage().window().maximize();  
 driver.manage().timeouts().implicitlyWait(Duration.*ofSeconds*(10));  
   
 System.*out*.println(driver.findElements(By.*tagName*("iframe")).size());  
  
 driver.switchTo().frame("iframeResult");  
 *//driver.findElement(By.xpath("/html/body/button")).click();* System.*out*.println(driver.findElements(By.*tagName*("iframe")).size());  
  
 driver.switchTo().frame(1);  
 driver.findElement(By.*xpath*("//\*[@id=\"nav\_translate\_btn\"]/i")).click();  
  
 *//driver.switchTo().defaultContent();  
 //driver.switchTo().frame("iframeResult");* driver.switchTo().parentFrame();  
 driver.findElement(By.*xpath*("/html/body/button")).click();  
 System.*out*.println(driver.findElements(By.*tagName*("iframe")).size());  
 }  
 }

public class TestAlert {  
  
 public static void main(String[] args) {  
 *//WebDriverManager.chromedriver().setup();* WebDriver driver = new ChromeDriver();  
 driver.get("https://mail.rediff.com/cgi-bin/login.cgi");  
 driver.manage().window().maximize();  
 driver.manage().timeouts().implicitlyWait(10, TimeUnit.*SECONDS*);  
 driver.findElement(By.*name*("proceed")).click();  
   
 */\* //1st method using SwitchTo()  
 Alert alert = driver.switchTo().alert();  
 System.out.println(alert.getText());  
 alert.accept();\*/  
   
 /\* //2nd method using Explicit wait  
 WebDriverWait wait = new WebDriverWait(driver, Duration.ofSeconds(10));  
 Alert alert = wait.until(ExpectedConditions.alertIsPresent());  
 System.out.println(alert.getText());  
 alert.accept();\*/  
  
 // 3rd method using Javascriptexecutor* JavascriptExecutor js = (JavascriptExecutor)driver;  
 try {  
 js.executeScript("window.alert=function{};");  
 } catch (Exception e) {  
 System.*out*.println(e.getMessage());} }  
}

public class TestAlerts\_Actions\_MoveToElement {  
  
 public static void main(String[] args) throws InterruptedException {  
 WebDriverManager.*chromedriver*().setup();  
 WebDriver driver = new ChromeDriver();  
 driver.get("http://www.tizag.com/javascriptT/javascriptalert.php");  
 driver.manage().window().maximize();  
  
 JavascriptExecutor js = (JavascriptExecutor) driver;  
 *//Locating element by link text and store in variable "Element"* WebElement Element = driver.findElement(By.*xpath*("//input[@value='Confirmation Alert']"));  
 *// Scrolling down the page till the element is found* js.executeScript("arguments[0].scrollIntoView();", Element);  
 Element.click();  
 System.*out*.println(driver.switchTo().alert().getText()); *// Get text within alert* Thread.*sleep*(5000);  
 driver.switchTo().alert().accept(); *//Accept the alert  
  
//HANDLING A normal alert which is as below  
  
 /\* WebDriverManager.chromedriver().setup();  
 WebDriver driver = new ChromeDriver();  
  
 driver.get("https://americangolf.co.uk/");  
 driver.manage().window().maximize();  
 Actions a = new Actions(driver);  
  
 driver.manage().timeouts().implicitlyWait(Duration.ofSeconds(5));  
 //Locating element by link text and store in variable "Element"  
 WebElement Element = driver.findElement(By.xpath("//button[contains(text(),'Accept')]"));  
 Element.click();  
 //clicking on Golf Clubs by HANDLING MOUSE HOVER  
 WebElement Elt = driver.findElement(By.xpath("//\*[@id=\"header-navigation\"]/div/ul/li[3]/a"));  
 a.moveToElement(Elt).build().perform();  
 //Elt.click();  
 Thread.sleep(3000);  
 // HANDLING MOUSE HOVER and clicking on Driver link  
 WebElement Element2 = driver.findElement(By.xpath("(//\*[contains(text(),'Drivers')])[4]"));  
 Element2.click();\*/* }  
  
 public class TestBrowsers {  
   
 public static String *browser* = "chrome"; *//excel sheet* public static WebDriver *driver*;  
 public static void main(String[] args) {   
 if(*browser*.equals("chrome")) {  
 WebDriverManager.*chromedriver*().setup();  
 *driver* = new ChromeDriver();  
   
 }else if(*browser*.equals("firefox")) {  
 WebDriverManager.*firefoxdriver*().setup();  
 *driver* = new FirefoxDriver();  
   
 }else if(*browser*.equals("ie")) {  
 WebDriverManager.*iedriver*().setup();  
 *driver* = new InternetExplorerDriver();  
 }  
   
 *driver*.get("http://way2automation.com");  
   
 String title = *driver*.getTitle();  
 System.*out*.println(title.length());  
 System.*out*.println(title);  
   
 *//driver.close(); //current browser window  
 driver*.quit(); *//current browser window + all related browser in the same session* }  
}

public class TestCheckboxes {   
public static WebDriver *driver*;   
 public static boolean isElementPresent(By by) {  
 */\*try {  
 driver.findElement(By.xpath(locator));  
 return true;  
 }catch(Throwable t) {  
   
 return false;  
 }\*/* int size = *driver*.findElements(by).size();  
 return size != 0;  
 }  
  
 public static void main(String[] args) {  
 WebDriverManager.*chromedriver*().setup();  
 *driver* = new ChromeDriver();  
 *driver*.get("http://www.tizag.com/htmlT/htmlcheckboxes.php");  
 *driver*.manage().window().maximize();  
 *driver*.manage().timeouts().implicitlyWait(10, TimeUnit.*SECONDS*);  
 JavascriptExecutor js = (JavascriptExecutor) *driver*;  
  
 */\* driver.findElement(By.xpath("//div[4]/input[1]")).click();  
   
 driver.findElement(By.xpath("//div[4]/input[2]")).click();  
   
 driver.findElement(By.xpath("//div[4]/input[3]")).click();  
   
 driver.findElement(By.xpath("//div[4]/input[4]")).click();  
\*/   
 /\*  
 for(int i=1; i<=4; i++) {  
   
 driver.findElement(By.xpath("//div[4]/input["+i+"]")).click();   
 }\*/   
 /\*int i=1;  
 int count=0;   
 while(isElementPresent(By.xpath("//div[4]/input["+i+"]"))) {  
   
 driver.findElement(By.xpath("//div[4]/input["+i+"]")).click();  
 i++;  
 count++;  
 }  
 System.out.println("Total checkboxes are : "+count);\*/* WebElement block = *driver*.findElement(By.*xpath*("/html/body/table[3]/tbody/tr[1]/td[2]/table/tbody/tr/td/div[4]"));  
 List<WebElement> checkboxes = block.findElements(By.*name*("sports"));  
 System.*out*.println("Total checkboxes are : "+checkboxes.size());  
 for(WebElement checkbox: checkboxes) {  
 js.executeScript("arguments[0].scrollIntoView();", checkbox );  
 checkbox.click();  
 }  
 }  
}

public class TestDragandDrop {  
  
 public static void main(String[] args) {  
 WebDriverManager.*firefoxdriver*().setup();  
 WebDriver driver = new FirefoxDriver();  
 driver.get("https://jqueryui.com/resources/demos/droppable/default.html");  
 driver.manage().window().maximize();  
 driver.manage().timeouts().implicitlyWait(10, TimeUnit.*SECONDS*);  
   
 WebElement draggable = driver.findElement(By.*id*("draggable"));  
 WebElement droppable = driver.findElement(By.*id*("droppable"));  
 Actions action = new Actions(driver);  
 action.dragAndDrop(draggable, droppable).perform();  
 }  
}

public class TestDropdowns {  
  
 public static void main(String[] args) {  
 WebDriverManager.*chromedriver*().setup();  
 WebDriver driver = new ChromeDriver();  
 driver.get("https://www.wikipedia.org/");  
 driver.manage().window().maximize();  
 driver.manage().timeouts().implicitlyWait(10, TimeUnit.*SECONDS*);  
 *//driver.findElement(By.id("searchLanguage")).sendKeys("Eesti");* WebElement dropdown = driver.findElement(By.*id*("searchLanguage"));  
 Select select = new Select(dropdown);  
   
 *//select.selectByVisibleText("Eesti");* select.selectByVisibleText("English");  
   
 List<WebElement> values = dropdown.findElements(By.*tagName*("option"));  
 System.*out*.println("Total values are : "+values.size());  
   
 System.*out*.println(values.get(7).getText());   
   
 for(int i=0; i<values.size(); i++) {  
   
 System.*out*.println(values.get(i).getAttribute("lang"));  
   
 }  
 WebElement block = driver.findElement(By.*xpath*("//\*[@id='www-wikipedia-org']/div[11]/div[3]"));  
   
 List<WebElement> links = block.findElements(By.*tagName*("a"));  
   
 System.*out*.println("----Printing links----");  
   
 System.*out*.println("Total links are: "+links.size());  
   
   
 for(WebElement link: links) {  
   
 System.*out*.println(link.getText()+" -- URL IS ---"+link.getAttribute("href"));  
 }  
 */\*  
 WebElement secondBlock = driver.findElement(By.id("blockid"));  
   
 secondBlock.findElements(By.id("abc")).get(2).click();\*/* }  
}

public class TestElementScreenshot {  
 public static WebDriver *driver*;  
  
 public static void captureScreenshot() throws IOException {  
  
 Date d = new Date();  
 String fileName = d.toString().replace(":", "\_").replace(" ", "\_") + ".jpg";  
  
 File screenshot = ((TakesScreenshot) *driver*).getScreenshotAs(OutputType.*FILE*);  
 FileUtils.*copyFile*(screenshot, new File(".//screenshot//" + fileName));  
 }  
 public static void captureEleScreenshot(WebElement ele) throws IOException {  
  
 Date d = new Date();  
 String fileName = d.toString().replace(":", "\_").replace(" ", "\_") + ".jpg";   
 File screenshot = ((TakesScreenshot) *driver*).getScreenshotAs(OutputType.*FILE*);  
 BufferedImage fullImg = ImageIO.*read*(screenshot);  
  
 Point point = ele.getLocation();  
 int eleWidth = ele.getSize().getWidth();  
 int eleHeight = ele.getSize().getHeight();  
  
 BufferedImage eleScreenshot = fullImg.getSubimage(point.getX(), point.getY(), eleWidth, eleHeight);  
 ImageIO.*write*(eleScreenshot, "jpg", screenshot);  
  
 File screenshotLocation = new File(".\\screenshot\\"+fileName);  
 FileUtils.*copyFile*(screenshot, screenshotLocation);  
 }  
  
 public static void main(String[] args) throws IOException {  
 WebDriverManager.*firefoxdriver*().setup();  
 *driver* = new FirefoxDriver();  
 *driver*.get("http://www.way2automation.com/");  
 *driver*.manage().window().maximize();  
 *driver*.manage().timeouts().implicitlyWait(10, TimeUnit.*SECONDS*);  
  
 WebElement ele = *driver*.findElement(By.*xpath*("/html/body/div[4]/div/header/div[2]/div/div[1]"));  
  
 *captureScreenshot*();  
 }  
}======

public class TestFindingElements {  
  
 public static void main(String[] args) {  
 WebDriverManager.*firefoxdriver*().setup();  
 WebDriver driver = new FirefoxDriver();  
 driver.get("http://gmail.com");  
 driver.manage().window().maximize();  
 *// driver.manage().timeouts().implicitlyWait(10, TimeUnit.SECONDS);  
 // WebDriverWait wait = new WebDriverWait(driver, 30);* Wait<WebDriver> wait = new FluentWait<WebDriver> (driver)  
 .withTimeout(Duration.*ofSeconds*(10))  
 .pollingEvery(Duration.*ofSeconds*(2))  
 .withMessage("User defined Timed out after 30 seconds")  
 .ignoring(NoSuchElementException.class);  
  
 */\* String title = driver.getTitle(); \*/* System.*out*.println(driver.getTitle().length());  
  
 */\*  
 \* WebElement username =  
 \* driver.findElement(By.xpath("//\*[@id='identifierId']"));  
 \* username.sendKeys("trainer@way2automation.com");  
 \*   
 \* WebElement nextBtn =  
 \* driver.findElement(By.xpath("//\*[@id=\"identifierNext\"]/content"));  
 \* nextBtn.click();  
 \*/  
  
 // driver.findElement(By.xpath("//\*[@id='identifierId']")).sendKeys("trainer@way2automation.com");* wait.until(ExpectedConditions.*presenceOfElementLocated*(By.*xpath*("//\*[@id='identifierId']")))  
 .sendKeys("trainer@way2automation.com");  
  
 driver.findElement(By.*xpath*("//\*[@id=\"identifierNext\"]/content")).click();  
 */\*  
 \* try { Thread.sleep(1000); } catch (InterruptedException e) { // TODO  
 \* Auto-generated catch block e.printStackTrace(); }  
 \*/  
 // Implicit Wait and ExplicitWait, FluentWait  
 // driver.findElement(By.name("password")).sendKeys("sdfsdfsdf");* wait.until(ExpectedConditions.*visibilityOfElementLocated*(By.*name*("password"))).sendKeys("asdfsdfds");  
  
 driver.findElement(By.*xpath*("//\*[@id=\"passwordNext\"]/content/span")).click();  
  
 System.*out*.println(wait  
 .until(ExpectedConditions.*presenceOfElementLocated*(By.*xpath*("//\*[@id=\"password\"]/div[2]/div[2]/div")))  
 .getText());  
 }  
}

public class TestIFrames {  
   
 public static WebDriver *driver*;   
 public static void captureScreenshot() throws IOException {   
 Date d = new Date();  
 String fileName = d.toString().replace(":", "\_").replace(" ", "\_")+".jpg";  
  
 File screenshot = ((TakesScreenshot) *driver*).getScreenshotAs(OutputType.*FILE*);  
 FileUtils.*copyFile*(screenshot, new File(".//screenshot//"+fileName));   
 }  
 public static void main(String[] args) throws IOException {  
 WebDriverManager.*chromedriver*().setup();  
 *driver* = new ChromeDriver();  
 *driver*.get("https://www.w3schools.com/jsref/tryit.asp?filename=tryjsref\_submit\_get");  
 *driver*.manage().window().maximize();  
 *driver*.manage().timeouts().implicitlyWait(10, TimeUnit.*SECONDS*);  
  
 */\*Set<String> winids = driver.getWindowHandles();  
 Iterator<String> iterate = winids.iterator();  
 String first\_window = iterate.next(); \*/  
   
 driver*.switchTo().frame("iframeResult");  
 *//driver.findElement(By.xpath("/html/body/button")).click();   
   
 //((JavascriptExecutor) driver).executeScript("myFunction()");* JavascriptExecutor js = (JavascriptExecutor) *driver*;  
 js.executeScript("myFunction()");  
   
 js.executeScript("arguments[0].style.border='3px solid red'", *driver*.findElement(By.*id*("mySubmit")));  
   
 *//driver.switchTo().window(first\_window);  
 driver*.switchTo().defaultContent();   
 List<WebElement> frames = *driver*.findElements(By.*tagName*("iframe"));  
 System.*out*.println(frames.size());  
 for (WebElement frame : frames) {  
 System.*out*.println(frame.getAttribute("id"));  
 }   
   
 *captureScreenshot*();   
 }  
}

public class TestKeyboardEvents\_CopyPaste {  
  
 public static void main(String[] args) {  
 *//WebDriverManager.chromedriver().setup();* WebDriver driver = new ChromeDriver();  
 driver.get("http://gmail.co.in");  
 driver.manage().window().maximize();  
 driver.manage().timeouts().implicitlyWait(10, TimeUnit.*SECONDS*);  
  
 driver.findElement(By.*id*("identifierId")).sendKeys("trainer@way2automation.com");  
 driver.findElement(By.*xpath*("/html/body/div[1]/div[1]/div[2]")).click();  
 *//driver.findElement(By.id("identifierId")).sendKeys(Keys.ENTER);* Actions action = new Actions(driver);  
 action.sendKeys(Keys.*chord*(Keys.*CONTROL*+"a")).sendKeys(Keys.*chord*(Keys.*CONTROL*+"c")).perform();   
 driver.findElement(By.*id*("identifierId")).click();  
 action.sendKeys(Keys.*chord*(Keys.*CONTROL*+"v")).perform();   
 }  
}

public class TestMouseOver {  
  
 public static void main(String[] args) {  
 *//WebDriverManager.firefoxdriver().setup();* WebDriver driver = new FirefoxDriver();  
 driver.get("http://google.com");  
 driver.manage().window().maximize();  
 driver.manage().timeouts().implicitlyWait(10, TimeUnit.*SECONDS*);  
 driver.findElement(By.*name*("q")).sendKeys("way2automation");   
 WebDriverWait wait = new WebDriverWait(driver, Duration.*ofSeconds*(10));  
 wait.until(ExpectedConditions.*visibilityOfElementLocated*(By.*xpath*("/html/body/div/div[3]/form/div[2]/div/div[2]/div[2]/div/center/input[1]"))).click();  
   
 driver.findElement(By.*xpath*("/html/body/div[6]/div[3]/div[10]/div[1]/div[2]/div/div[2]/div[2]/div/div/div/div[1]/div/div/div/div/div[1]/a/h3")).click();  
   
 WebElement menu = driver.findElement(By.*xpath*("/html/body/div[4]/div/header/div[3]/div/nav/div[2]/ul/li[8]/a"));  
   
 Actions action = new Actions(driver);  
 action.moveToElement(menu).perform();  
   
 driver.findElement(By.*linkText*("Practice site 1")).click();   
 }  
}

public class TestMultipleElementsWithRelativeLocators {  
  
 public static void main(String[] args) {  
 WebDriverManager.*chromedriver*().setup();  
 WebDriver driver = new ChromeDriver();  
 driver.get("https://www.way2automation.com/way2auto\_jquery/index.php");  
 driver.manage().window().maximize();  
 driver.manage().timeouts().implicitlyWait(Duration.*ofSeconds*(5));  
  
 List<WebElement> labels = driver.findElements(*with*(By.*tagName*("label")).below(By.*xpath*("//\*[@id=\"load\_form\"]/h3")));  
 for(WebElement label: labels) {  
 System.*out*.println(label.getText());  
 }  
 }  
 }

public class TestNewTabsAndWindows {  
 public static void main(String[] args) throws InterruptedException {  
 WebDriver driver = new ChromeDriver();  
 driver.get("https://www.way2automation.com/");  
 driver.manage().window().maximize();  
 driver.manage().timeouts().implicitlyWait(Duration.*ofSeconds*(10));  
 System.*out*.println(driver.getTitle());  
 driver.switchTo().newWindow(WindowType.*TAB*); *//2ND window* driver.get("http://google.com");  
 driver.findElement(By.*name*("q")).sendKeys("Hello Selenium 4");  
 Thread.*sleep*(2000);  
 System.*out*.println("This is new Window tab : " + driver.getTitle());  
 driver.switchTo().newWindow(WindowType.*WINDOW*); *//3rD window* driver.get("http://gmail.com");  
 System.*out*.println("This is the New Window : " + driver.getTitle());  
 System.*out*.println("Total no of windows : " + driver.getWindowHandles().size());  
 *//Get all Window IDs* Set<String> winids = driver.getWindowHandles();  
 *//Iterate through each of the windows* Iterator<String> iterator = winids.iterator();  
 List<String> winIndex = new ArrayList<String>();  
 int i = 0;  
 while (iterator.hasNext()) {  
 winIndex.add(iterator.next()); *// adding winids to the ArrayList* driver.switchTo().window(winIndex.get(i));  
 System.*out*.println("First window title is :" + driver.getTitle());  
 driver.close();  
 i++;  
 }  
 driver.quit();  
  
 */\* driver.switchTo().window(winIndex.get(0));  
 System.out.println("First window title is :" + driver.getTitle());  
 driver.close();  
  
 driver.switchTo().window(winIndex.get(1));  
 System.out.println("second window title is :" + driver.getTitle());  
 driver.close();  
 driver.switchTo().window(winIndex.get(2));  
 System.out.println("third window title is :" + driver.getTitle());\*/* driver.quit();  
 }  
}

public class TestResizable {  
 public static void main(String[] args) {  
 WebDriver driver = new ChromeDriver();  
 driver.get("https://jqueryui.com/resources/demos/resizable/default.html");  
 driver.manage().window().maximize();  
 driver.manage().timeouts().implicitlyWait(10, TimeUnit.*SECONDS*);  
   
 *// element to be resized* WebElement resizable = driver.findElement(By.*xpath*("//\*[@id=\"resizable\"]/div[3]"));  
 Actions action = new Actions(driver);  
 action.dragAndDropBy(resizable, 800, 400).perform();}}

public class TestRightClick {  
 public static void main(String[] args) {  
 *//WebDriverManager.firefoxdriver().setup();* WebDriver driver = new FirefoxDriver();  
 driver.get("http://deluxe-menu.com/popup-mode-sample.html");  
 driver.manage().window().maximize();  
 driver.manage().timeouts().implicitlyWait(10, TimeUnit.*SECONDS*);  
  
 WebElement img = driver.findElement(By.*xpath*("/html/body/div/table/tbody/tr/td[2]/div[2]/table[1]/tbody/tr/td[3]/p[2]/img"));  
 Actions action = new Actions(driver);  
 action.contextClick(img).perform();  
 }  
}

public class TestSliders {  
  
 public static void main(String[] args) {  
 WebDriver driver = new ChromeDriver();  
 driver.get("https://jqueryui.com/resources/demos/slider/default.html");  
 driver.manage().window().maximize();  
 driver.manage().timeouts().implicitlyWait(10, TimeUnit.*SECONDS*);  
  
 WebElement mainSlider = driver.findElement(By.*id*("slider")); *// entire horizontal slider* int width = mainSlider.getSize().width/2; *//will slide horizontally exactly by half of the sceen if given as width/2  
 //will slide horizontally full screen to the end if given as width only* WebElement sliderButton = driver.findElement(By.*xpath*("//\*[@id=\"slider\"]/span"));  
 new Actions(driver).dragAndDropBy(sliderButton, width, 0).perform();  
 }  
}

public class TestTabsandPopups {  
  
 public static void main(String[] args) {  
 WebDriver driver = new ChromeDriver();  
 driver.get("http://hdfcbank.com");  
 driver.manage().window().maximize();  
 driver.manage().timeouts().implicitlyWait(10, TimeUnit.*SECONDS*);  
  
 System.*out*.println("----Generating window ids from first window----");  
 Set<String> winids = driver.getWindowHandles();  
 Iterator<String> iterate = winids.iterator();  
  
 String first\_window = iterate.next();  
 System.*out*.println(first\_window);  
  
 driver.findElement(By.*id*("loginsubmit")).click();  
  
 *// 2nd window* System.*out*.println("----Generating window ids from Second window----");  
  
 winids = driver.getWindowHandles();  
 iterate = winids.iterator();  
  
 System.*out*.println(iterate.next()); *// first window* String second\_window = iterate.next(); *// second window* System.*out*.println(second\_window);  
  
 driver.switchTo().window(second\_window);  
 driver.findElement(By.*xpath*("/html/body/div[4]/div[3]/div/div/div[1]/div/a")).click();  
 driver.findElement(By.*xpath*("/html/body/div[1]/div[2]/div[2]/p[3]/a")).click();  
  
 *// 3rd window* System.*out*.println("----Generating window ids from third window----");  
  
 winids = driver.getWindowHandles();  
 iterate = winids.iterator();  
  
 System.*out*.println(iterate.next()); *// first window* System.*out*.println(iterate.next()); *// second window* String third\_window = iterate.next(); *//3rd window  
   
 /\*while(iterate.hasNext()) {  
   
 iterate.next();  
 }\*/* System.*out*.println(third\_window);  
 driver.switchTo().window(third\_window);  
   
 System.*out*.println(driver.getTitle().contains("JetPrivilege HDFC Bank Signature"));  
  
 */\* driver.close(); //3rd window  
 driver.switchTo().window(second\_window);  
 driver.close();\*/* driver.quit();  
 }  
}

public class TestTimeStamp {  
  
 public static void main(String[] args) {  
 Date d = new Date();  
 System.*out*.println(d.toString().replace(":", "\_").replace(" ", "\_")+".jpg");  
 }  
}

public class TestWebTable {  
  
 public static void main(String[] args) {  
  
 *//WebDriverManager.chromedriver().setup();* WebDriver driver = new ChromeDriver();  
 driver.get("https://money.rediff.com/gainers/bsc/daily/groupa");  
 driver.manage().window().maximize();  
 driver.manage().timeouts().implicitlyWait(10, TimeUnit.*SECONDS*);  
  
 List<WebElement> rowNum = driver.findElements(By.*xpath*("//table[@class='dataTable']/tbody/tr"));  
 System.*out*.println("Total rows are : " + rowNum.size());  
  
 List<WebElement> colNum = driver.findElements(By.*xpath*("//table[@class='dataTable']/tbody/tr[1]/td"));  
 System.*out*.println("Total cols are : " + colNum.size());  
 for(int rows=1; rows<=rowNum.size(); rows++) {  
 for(int cols=1; cols<=colNum.size(); cols++) {  
 System.*out*.print(driver.findElement(By.*xpath*("//table[@class='dataTable']/tbody/tr["+rows+"]/td["+cols+"]")).getText()+" ");  
 }  
 System.*out*.println();  
 } }  
}

public class UsingXY\_Coordinates {  
  
 public static void main(String[] args) {  
 *// WebDriverManager.chromedriver().setup();* WebDriver driver = new ChromeDriver();  
 driver.get("http://google.com");  
 driver.manage().window().maximize();  
 driver.manage().timeouts().implicitlyWait(10, TimeUnit.*SECONDS*);  
  
 WebElement link = driver.findElement(By.*linkText*("Gmail"));  
 int x = link.getRect().getX();  
 int y = link.getRect().getY();  
  
 Actions actions = new Actions(driver);  
 actions.moveByOffset(x,y).click().perform();  
 driver.quit();  
 }}

public class WindowHandleTest {  
  
 @Test  
 public void windowsExampleCode() {  
  
 WebDriver driver = new ChromeDriver();  
 driver.manage().timeouts().implicitlyWait(Duration.*ofMillis*(500));  
 *// Navigate to Url* driver.get("https://www.selenium.dev/selenium/web/window\_switching\_tests/page\_with\_frame.html");  
 *//fetch handle of this* String currHandle = driver.getWindowHandle();  
 System.*out*.println(currHandle);  
  
 *//click on link to open a new window* driver.findElement(By.*linkText*("Open new window")).click();  
 *//fetch handles of all windows, there will be two, [0]- default, [1] - new window* Object[] windowHandles = driver.getWindowHandles().toArray();  
 driver.switchTo().window((String) windowHandles[1]);  
 *//assert on title of new window* String title = driver.getTitle();  
 Assert.*assertEquals*("Simple Page", title);  
  
 *//closing current window* driver.close();  
 *//Switch back to the old tab or window* driver.switchTo().window((String) windowHandles[0]);  
  
 *//Opens a new tab and switches to new tab* driver.switchTo().newWindow(WindowType.*TAB*);  
 Assert.*assertEquals*("", driver.getTitle());  
  
 *//Opens a new window and switches to new window* driver.switchTo().newWindow(WindowType.*WINDOW*);  
 Assert.*assertEquals*("", driver.getTitle());  
  
 *//quitting driver* driver.quit(); *//close all windows  
 // }* }  
}

public class WindowHandling {  
 public static void main(String[] args) throws InterruptedException {  
  
 *// WebDriverManager.chromedriver().setup();* WebDriver driver = new ChromeDriver();  
 driver.get("https://www.way2automation.com/");  
 driver.manage().window().maximize();  
 driver.manage().timeouts().implicitlyWait(Duration.*ofSeconds*(10));  
 System.*out*.println(driver.getTitle());  
  
 driver.switchTo().newWindow(WindowType.*TAB*);  
 driver.get("https://www.google.com");  
 Thread.*sleep*(2000);  
 System.*out*.println(driver.getTitle());  
  
 driver.switchTo().newWindow(WindowType.*WINDOW*);  
 driver.get("https://www.gmail.com");  
 Thread.*sleep*(2000);  
 System.*out*.println(driver.getTitle());  
  
 System.*out*.println("Total num of windows : "+ driver.getWindowHandles().size());  
*//adding all winids to a set, as a set does not hold duplicates* Set<String> winIds= driver.getWindowHandles();  
 *// iterate through each winIds* Iterator<String> iterator = winIds.iterator();  
  
 List<String> al = new ArrayList<String>();  
 while (iterator.hasNext()){  
 al.add(iterator.next());  
 *//System.out.println(iterator);* }  
  
 driver.switchTo().window(al.get(0));  
 System.*out*.println(driver.getTitle());  
 driver.close();  
  
 driver.switchTo().window(al.get(1));  
 System.*out*.println(driver.getTitle());  
 driver.close();  
  
 driver.switchTo().window(al.get(2));  
 System.*out*.println(driver.getTitle());  
 driver.close();  
 }  
}

public class ScreenshotUsingAshot {  
 public static void main(String[] args) throws IOException {  
 *// Replace with your WebDriver instance* WebDriver driver = new ChromeDriver(); *// Example using ChromeDriver* driver.get("https://www.example.com"); *// Replace with the URL  
  
 // Take the screenshot* Screenshot screenshot = new AShot()  
 .shootingStrategy(ShootingStrategies.*viewportPasting*(1000)) *// Scrolls the page to capture the full content* .takeScreenshot(driver);  
  
 *// Save the screenshot* File file = new File("./screenshot/full\_page\_screenshot.png");  
 ImageIO.*write*(screenshot.getImage(), "PNG", file);  
  
 System.*out*.println("Screenshot saved to: " + file.getAbsolutePath());  
  
 driver.quit();  
 }  
}

public class TestRelativeLocators {  
  
 public static void main(String[] args) throws IOException {  
 WebDriver driver = new ChromeDriver();  
 driver.get("https://www.way2automation.com/way2auto\_jquery/index.php");  
 driver.manage().window().maximize();  
 driver.manage().timeouts().implicitlyWait(Duration.*ofSeconds*(5));  
  
 WebElement above = driver.findElement(*with*(By.*tagName*("input")).above(By.*tagName*("select"))); *// Email field* above.sendKeys("trainer@way2automation.com"); *//entering data in email field* File aboveScrn = above.getScreenshotAs(OutputType.*FILE*);  
 FileUtils.*copyFile*(aboveScrn, new File("./screenshot/above.jpg"));  
  
 WebElement below = driver.findElement(*with*(By.*tagName*("input")).below(By.*tagName*("select")));  
 below.sendKeys("New Delhi"); *//CITY field* File belowScrn = below.getScreenshotAs(OutputType.*FILE*);  
 FileUtils.*copyFile*(belowScrn, new File("./screenshot/below.jpg"));  
 *//WebElement near = driver.findElement(RelativeLocator.with(By.partialLinkText("THE")).near(By.linkText("Signin")));  
 //near.click();* WebElement rightOf = driver.findElement(*with*(By.*xpath*("//input[@type='password']")).toRightOf(By.*tagName*("label")));  
 rightOf.sendKeys("asfddsfsf"); *//Password* File rightScrn = rightOf.getScreenshotAs(OutputType.*FILE*);  
 FileUtils.*copyFile*(rightScrn, new File("./screenshot/right.jpg"));  
  
 WebElement leftOf = driver.findElement(*with*(By.*linkText*("Signin")).toLeftOf(By.*xpath*("(//\*[@id=\"load\_form\"]/div[1]/div[2]/input)[2]"))); *//SUBMIT* File leftScrn = leftOf.getScreenshotAs(OutputType.*FILE*);  
 FileUtils.*copyFile*(leftScrn, new File("./screenshot/left.jpg"));  
 leftOf.click();  
  
 *//CHAINING RELATIVE LOCATORS* driver.findElement(*with*(By.*tagName*("input"))  
 .above(By.*linkText*("Signin"))  
 .below(By.*xpath*("//\*[@id=\"load\_form\"]/fieldset[6]/input")))  
 .sendKeys("ksdfksfdsdf");  
 }  
}

public class StringToIntConversion {  
  
public static void main(String[] args)  
{  
 String s = "12345";  
 *//String s = "subhas";  
  
 // Convert the string to an integer using Integer.parseInt()* int n = Integer.*parseInt*(s);  
  
 System.*out*.println("Converted String to int : " + n);  
 *//2ND METHOD  
 // Convert String to Integer using valueOf()* String t = "217";  
  
 *// Convert the string to an Integer object  
 // using Integer.valueOf()* int p = Integer.*valueOf*(t);  
  
 System.*out*.println("Using 2nd method, Converted String to int" + p);  
  
 *// CONVERT INTEGER TO STRING* int number= 11;  
 String str =String.*valueOf*(number);  
 System.*out*.println("CONVERT INTEGER TO STRING : " + str);  
  
 *// CONVERT INTEGER TO STRING 2nd method* int number2= 1111;  
 String str2 =Integer.*toString*(number2);  
 System.*out*.println("CONVERT INTEGER TO STRING 2nd method : " + str2);  
}  
}

*/\*Since all arrays hold only respective data types, we need to create an array  
which can hold all types of data in a single array.\*/*public class ObjectArray {  
  
 public static void main(String[] args) {  
 Object ar[]= new Object[5];  
 ar[0]="Hello";  
 ar[1]= 10293;  
 ar[2]= "333";  
 ar[3]= true;  
 ar[4]= 33.333;  
  
 System.*out*.println(ar.toString());  
 System.*out*.println(String.*valueOf*(ar));  
 System.*out*.println(ar[0]);  
 }  
}

public class MaxAndMinElementsInArray {  
 public static void main(String[] args) {  
 int a[] = {50, 30, 40, 20, 60};  
 int max = a[0];  
 for (int i = 0; i < a.length; i++) {  
 if (a[i] > max) {  
 max = a[i];  
 }  
 }  
 System.*out*.println("Maximum value of element in an array = "+max);  
 */// Minimum element in an array* int arr[] = {50, 30, 40, 20, 60};  
 int min = arr[0];  
  
 for (int i = 0; i < a.length; i++) {  
 *//if (a[i] > max) {* if (arr[i] < min) {  
 min = arr[i];  
 }  
 }  
 System.*out*.println("Minimum value of element in an array = "+min);  
 }  
}

public class IteratingArraylist {  
 public static void main(String args[]) {  
 ArrayList<String> names = new ArrayList<String>();  
 names.add("Subash");  
 names.add("Amrita");  
 names.add("Varun");  
 names.add("Shreyas");  
 names.add("Patil");  
  
 *//iterating ArrayList* for (String name : names) {  
 System.*out*.println(name);  
 }  
 *// Sorting an ArrayList using Collections.sort() method* Collections.*sort*(names);  
  
 System.*out*.println("After Sort : " + names);  
  
 Collections.*sort*(names, Collections.*reverseOrder*());  
  
 System.*out*.println("Sorting in reverse order : " + names);  
  
 names.sort(Comparator.*naturalOrder*());  
 System.*out*.println("Comparator Sorted Names naturalOrder : " + names);  
  
 names.sort(Comparator.*reverseOrder*());  
 System.*out*.println("Comparator Sorted Names Reverse order : " + names);  
 }  
}

public class IsEmptyExample {  
 public static void main(String args[]) {  
 *//ArrayList of Integer Type* ArrayList<Integer> al = new ArrayList<Integer>();  
 *//Checking whether the list is empty* System.*out*.println("Is ArrayList Empty: "+al.isEmpty());  
  
 *//Adding Integer elements* al.add(1);  
 al.add(88);  
 al.add(9);  
 al.add(17);  
  
 *//Again checking for isEmpty* System.*out*.println("Is ArrayList Empty: "+al.isEmpty());  
 *//Displaying elements of the list* for (Integer num: al) {  
 System.*out*.println(num);  
 }  
 }  
}============

public class AccessElementsFromArrayListExample {  
 public static void main(String[] args) {  
 List<String> topCompanies = new ArrayList<>();  
  
 *// Check if an ArrayList is empty* System.*out*.println("Is the topCompanies list empty? : " + topCompanies.isEmpty());  
  
 topCompanies.add("Google");  
 topCompanies.add("Apple");  
 topCompanies.add("Microsoft");  
 topCompanies.add("Amazon");  
 topCompanies.add("Facebook");  
  
 *// Find the size of an ArrayList* System.*out*.println("Here are the top " + topCompanies.size() + " companies in the world");  
 System.*out*.println(topCompanies);  
  
 *// Retrieve the element at a given index* String bestCompany = topCompanies.get(0);  
 String secondBestCompany = topCompanies.get(1);  
 String lastCompany = topCompanies.get(topCompanies.size() - 1);  
  
 System.*out*.println("Best Company: " + bestCompany);  
 System.*out*.println("Second Best Company: " + secondBestCompany);  
 System.*out*.println("Last Company in the list: " + lastCompany);  
  
 *// Modify the element at a given index* topCompanies.set(4, "Walmart");  
 System.*out*.println("Modified top companies list: " + topCompanies);  
 }  
}

public class TwoDArray {  
  
 public static void main(String[] args) {  
 int[][] myArray=new int[2][3];   
   
 myArray[0][0]=1;  
 myArray[0][1]=2;  
 myArray[0][2]=3;  
   
 myArray[1][0]=4;  
 myArray[1][1]=5;  
 myArray[1][2]=6;  
   
 */\* System.out.println(myArray[1][1]);  
   
 System.out.println(myArray.length);  
   
 System.out.println(myArray[0].length);\*/* for(int i=0; i<myArray.length; i++) {  
   
 for(int j=0;j<myArray[i].length; j++) {  
   
 System.*out*.print(myArray[i][j]+" ");  
   
 }  
   
 System.*out*.println();  
 }  
 }  
}

public class ThreeDArray {  
  
 public static void main(String[] args) {  
 int[][][] array = new int[2][3][4];  
   
   
 for(int i=0; i<array.length; i++) {  
   
 for(int j=0; j<array[i].length; j++) {  
   
   
 for(int k=0; k<array[i][j].length; k++) {  
   
 array[i][j][k] = (int)(Math.*random*()\*1000);  
 System.*out*.print(array[i][j][k]+ " ");  
 }  
 System.*out*.println();  
 }  
 System.*out*.println();   
 }   
 }  
}

STRINGS

public class CharAt {  
 public static void main(String args[]) {  
 String str = "BeginnersBook";  
 String str2;  
 for(int i=0; i<=str.length()-1; i++) {  
 System.*out*.println(str.charAt(i));  
 */\* str2 += str2 + str.charAt(i);  
 System.out.println(str2);\*/* }  
 }  
}

public class CharAtExample {  
 public static void main(String args[]) {  
 String str = "Welcome to string handling tutorial";  
 *//This will return the first char of the string* char ch1 = str.charAt(0);  
  
 *//This will return the 6th char of the string* char ch2 = str.charAt(5);  
  
 *//This will return the 12th char of the string* char ch3 = str.charAt(11);  
  
 *//This will return the 21st char of the string* char ch4 = str.charAt(20);  
  
 *// IndexOutOfBoundsExceptionExample  
 //char ch5 = str.charAt(50);* System.*out*.println("Character at 0 index is: "+ch1);  
 System.*out*.println("Character at 5th index is: "+ch2);  
 System.*out*.println("Character at 11th index is: "+ch3);  
 System.*out*.println("Character at 20th index is: "+ch4);  
 }  
 }

*/\*Java String charAt() example to count the occurrence of a character  
In this example, we will use the charAt() method to count the occurrence of a particular  
character in the given string. Here we have a string and we are counting the occurrence  
of character ‘B’ in the string.\*/*public class CountOccurenceOfCharacter {  
 public static void main(String[] args) {  
 String str = "BeginnersBook B";  
  
 *//initialized the counter to 0* int counter = 0;  
  
 for (int i=0; i<=str.length()-1; i++) {  
 if(str.charAt(i) == 'B') {  
 *//increasing the counter value at each occurrence of 'B'* counter++;  
 */\*int cnt =1;  
 cnt++;\*/* System.*out*.println(" B occured at index : " +i);  
  
 }  
 }  
 System.*out*.println("Char 'B' occurred "+counter+" times in the string");  
 }  
}

public class CountWordsInString {  
 public static void main(String[] args) {  
  
 Scanner s = new Scanner(System.*in*);  
 System.*out*.println("Enter the string");  
 String str = s.nextLine();  
 int cnt = 0;  
 for (int i = 0; i < str.length(); i++) {  
 if ((str.charAt(i) == ' ') && (str.charAt(i + 1) != ' '))  
 {  
 cnt = cnt + 1;  
 }  
 }  
 System.*out*.println("Number of words in String ==" + cnt);  
 }  
}

public class CountWordsInString2 {  
  
 public static void main(String[] args) {  
 Scanner s = new Scanner(System.*in*);  
 System.*out*.println("Enter the string");  
 String str = s.nextLine();  
 *// calling the method to count the number of words in a string* int cnt = *countWordsUsingSplit*(str);  
 System.*out*.println("Number of words in string are - " + cnt);  
 }  
  
 public static int countWordsUsingSplit(String input) {  
 if (input == null || input.isEmpty()) {  
 return 0;  
 }  
 String[] words = input.split("\\s+");  
  
 return words.length;  
 }  
}

public class DuplicateCharacters {  
 public static void main(String argu[]) {  
 String str = "beautiful sea uu";  
 char[] carray = str.toCharArray();  
 System.*out*.println("The string is: " + str);  
 System.*out*.print("Duplicate Characters in above string are: ");  
 *// nested for loop to print duplicate characters* for (int i = 0; i < str.length(); i++) {  
 for (int j = i + 1; j < str.length(); j++) {  
 if (carray[i] == carray[j]) {  
 System.*out*.print(carray[j] + " ");  
 break;  
 }  
 }  
 }  
 }  
}

public class DuplicateCharactersInString {

public static void main(String[] args) {

String str = "how to doin java com vv";

char[] chars = str.toCharArray();

Map<Character, Integer> hm = new HashMap<Character, Integer>();

for (char c : chars) {

if (hm.containsKey(c)) {

hm.put(c, hm.get(c) + 1);

} else {

hm.put(c, 1);

}

}

System.out.println(hm);

for (Character x : hm.keySet()) {

if (hm.get(x) > 1 && (!Character.isWhitespace(x))) {

// if (hm.get(x) > 1 ) {

System.out.println("Duplicate element is = " + x);

}

/\* hm.forEach((key, value) -> {

if(hm.get(key)>1) {

System.out.println(" Duplicate element is Alphabet = " + key + " which occurs = " + value+" times");

}

});\*/

}

}}

public class DuplicateWordsInString {  
  
 public static void main(String[] args) {  
 String value = "This is testing Program , a , testing Program";  
 String words[]=value.split("\\s");  
  
 HashMap<String, Integer> hm = new HashMap<String, Integer>();  
  
 for (String word: words) {  
  
  
 if (hm.containsKey(word)) {  
 hm.put(word, hm.get(word) + 1);  
  
 } else {  
 hm.put(word, 1);  
 }  
 }  
  
 System.*out*.println(hm);  
 for (String word:hm.keySet())  
 {  
 if(hm.get(word)> 1)  
 {  
 System.*out*.println("Duplicate word is : "+word);  
 }  
  
 }  
 }  
}

*/\*As you can observe in the output that when we compared the String str1 (value “Hello”)  
with the string “hello”, the equals() method returned false because this method case sensitive  
and considers the case while comparing strings. On the other hand the equalsIgnoreCase() method  
compares strings while ignoring their cases, which we will see in the next section.\*/*public class EqualsMethod{  
 public static void main(String args[]){  
 String str1= new String("Hello");  
 String str2= new String("Hi");  
 String str3= new String("Hello");  
 System.*out*.println("str1 equals to str2:"+str1.equals(str2));  
 System.*out*.println("str1 equals to str3:"+str1.equals(str3));  
 System.*out*.println("str1 equals to Welcome:"+str1.equals("Welcome"));  
 System.*out*.println("str1 equals to Hello:"+str1.equals("Hello"));  
 System.*out*.println("str1 equals to hello:"+str1.equals("hello"));  
 }  
}  
  
*/\*str1 equals to str2:false  
str1 equals to str3:true  
str1 equals to Welcome:false  
str1 equals to Hello:true  
str1 equals to hello:false\*/*

public class HighlighterClass {  
  
 @Test  
 public void highlighterElement() {  
  
 *//System.setProperty("webdriver.gecko.driver", "D:\\Selenium Environment\\Drivers\\geckodriver.exe");* System.*setProperty*("webdriver.chrome.driver",  
 "E:\\MyWorkspace\_Completed\\FreeCRMTest\\Software\\chromedriver.exe");  
 *//WebDriver driver = new FirefoxDriver();* WebDriver driver = new ChromeDriver();  
 driver.manage().window().maximize();  
 driver.get("https://www.gmail.com");  
 WebElement ele = driver.findElement(By.*xpath*("//\*[@id='identifierId']"));  
 *//Call the highlighterMethod and pass webdriver and WebElement which you want to highlight as arguments.* highlightObject(driver,ele);  
 *//ele.sendKeys("SoftwareTestingMaterial.com");  
  
 //ALTERNATIVE METHOD TO ENTER TEXT OTHER THAN SENDKEYS* JavascriptExecutor js = (JavascriptExecutor)driver;  
 js.executeScript("arguments[0].value='SoftwareTestingMaterial.com';", ele);  
 }  
 *//Creating a custom function* public void highlightObject(WebDriver driver, WebElement element){  
 JavascriptExecutor js = (JavascriptExecutor) driver;  
 js.executeScript("arguments[0].setAttribute('style', 'background: yellow; border: 2px solid red;');", element);  
 }  
}

public class HighlighterClass {  
  
 @Test  
 public void highlighterElement() {  
 WebDriver driver = new ChromeDriver();  
 driver.manage().window().maximize();  
 driver.get("https://www.gmail.com");  
 WebElement ele = driver.findElement(By.*xpath*("//\*[@id='identifierId']"));  
 *//Call the highlighterMethod and pass webdriver and WebElement which you want to highlight as arguments.* highlightObject(driver,ele);  
 *//ele.sendKeys("SoftwareTestingMaterial.com");  
  
 //ALTERNATIVE METHOD TO ENTER TEXT OTHER THAN SENDKEYS* JavascriptExecutor js = (JavascriptExecutor)driver;  
 js.executeScript("arguments[0].value='SoftwareTestingMaterial.com';", ele);  
 }  
 *//Creating a custom function* public void highlightObject(WebDriver driver, WebElement element){  
 JavascriptExecutor js = (JavascriptExecutor) driver;  
 js.executeScript("arguments[0].setAttribute('style', 'background: yellow; border: 2px solid red;');", element);  
 }  
}

public class IndexOfExample{  
 public static void main(String args[]) {  
 String str1 = new String("This is a BeginnersBook tutorial");  
 String str2 = new String("Beginners");  
 String str3 = new String("Book");  
 String str4 = new String("Books");  
 System.*out*.println("Index of B in str1: "+str1.indexOf('B'));  
 System.*out*.println("Index of B in str1 after 15th char:"+str1.indexOf('B', 15));  
 System.*out*.println("Index of B in str1 after 30th char:"+str1.indexOf('B', 30));  
 System.*out*.println("Index of string str2 in str1:"+str1.indexOf(str2));  
 System.*out*.println("Index of str2 after 15th char"+str1.indexOf(str2, 15));  
 System.*out*.println("Index of string str3:"+str1.indexOf(str3));  
 System.*out*.println("Index of string str4"+str1.indexOf(str4));  
 System.*out*.println("Index of hardcoded string:"+str1.indexOf("is"));  
 System.*out*.println("Index of hardcoded string after 4th char:"+str1.indexOf("is", 4));  
  
 String str = "Java String";  
 char ch = 'J';  
 char ch2 = 'S';  
 String subStr = "tri";  
 int posOfJ = str.indexOf(ch);  
 int posOfS = str.indexOf(ch2);  
 int posOfSubstr = str.indexOf(subStr);  
 System.*out*.println(posOfJ);  
 System.*out*.println(posOfS);  
 System.*out*.println(posOfSubstr);  
  
  
 }  
}  
  
*/\*  
Index of B in str1: 10  
Index of B in str1 after 15th char:19  
Index of B in str1 after 30th char:-1  
Index of string str2 in str1:10  
Index of str2 after 15th char-1  
Index of string str3:19  
Index of string str4-1  
Index of hardcoded string:2  
Index of hardcoded string after 4th char:5\*/*

public class ParseInt{  
  
 public static void main(String args[]){  
 *//Integer a = new Integer("456");  
 // Casting not possible  
 // int a = (int)"456";  
 // Casting not possible  
 // int c="456";  
 // Casting possible using methods  
 // from Integer Wrapper class* int b = Integer.*parseInt*("456");  
 System.*out*.print(b);  
 }  
}

public class RegExCaseInSensitive {  
 public static void main(String[] args)  
 {  
 *//public int cnt;  
 // the string we want to search in* String str = "happYyy Y Y"; *// happy new year";  
  
 // search for this simple pattern* String sub\_str ="Y"; *// " ye.\* ";  
  
 // set everything up  
 //Pattern p = Pattern.compile(sub\_str); //Case sensitive* Pattern p = Pattern.*compile*(sub\_str, Pattern.*CASE\_INSENSITIVE*);  
 Matcher m = p.matcher(str);  
 int res=0;  
  
 *// now see if we find a match* while (m.find()) {  
 res++;  
 }  
 System.*out*.println("Found a match, "+ res+" times ");  
 }  
}

public class RegExChInString {  
  
 *// Method that returns the count of the given  
 // character in the string  
 //public static int RegExcount(String s, String ch) {* public static int RegExcount(String s, char ch) {  
 *// Use Matcher class of java.util.regex  
 // to match the character* Matcher matcher= Pattern.*compile*(String.*valueOf*(ch)).matcher(s);  
 *//System.out.println((String.valueOf(ch)));* int res=0;  
 while(matcher.find()) {  
 res++;  
 }  
 return res;  
 }  
  
 public static void main(String[] args) {  
 */\*String str = "geeksforgeeks";  
 char c = 'e';\*/* String str = "subash is a bad bad bad boy";  
 char c = 'b';  
 *//String c = "bad";* System.*out*.println(*RegExcount*(str, c));  
 }  
}

public class RegExWordInString {  
  
 *// Method that returns the count of the given  
 // character in the string  
 //public static int RegExcount(String s, String ch) {* public static int RegExcount(String s, String word) {  
 *// Use Matcher class of java.util.regex  
 // to match the character* Matcher matcher= Pattern.*compile*(String.*valueOf*(word)).matcher(s);  
 *//System.out.println((String.valueOf(ch)));* int res=0;  
 while(matcher.find()) {  
 res++;  
 }  
 return res;  
 }  
 public static void main(String[] args) {  
  
 String str = "subash is a bad bad bad boy";  
 String word = "bad";  
  
 System.*out*.println(*RegExcount*(str, word));  
 }  
}

public class ReverseANumber {  
  
 public static void main(String[] args) {  
 @SuppressWarnings("resource")  
 Scanner s = new Scanner(System.*in*);  
 System.*out*.println("Enter the number to be reversed");  
 int num = s.nextInt();  
 *// Using StringBuffer class* StringBuffer sb = new StringBuffer(String.*valueOf*(num));  
 StringBuffer rev = sb.reverse();  
 System.*out*.println("The reversed number is " + rev);  
  
 */// /========================================================  
 /\* METHOD TWO\*/* Scanner sC = new Scanner(System.*in*);  
 System.*out*.println("Enter the number to be reversed");  
 int num2 = sC.nextInt();  
 int rev2 = 0;  
  
 while (num2 != 0) {  
 rev2 = rev2 \* 10 + num % 10;  
 num = num / 10;  
 }  
 System.*out*.println("The reversed number is " + rev2);  
 }  
}

public class StringConcatExample {  
  
*/\*Syntax: String concate(String str);  
In order to concatenate multiple strings, we use concat() method in Java.  
We can also do String concatenation using + Operator, using StringBuilder and StringBuffer class to join Strings in Java.  
Example:  
 \*/* public static void main(String args[]) {  
  
 *//using string.concat* String s1="Hello";  
 String s2="all";  
 System.*out*.println(s1.concat(s2)); *// Helloall --  
 //adds second object's string to first object's end of string without any spaces.  
  
 //using + operator to concatenate String* String first = "Raj";  
 String last = "Chris";  
  
 String name = first + " " + last;  
 System.*out*.println(name);  
  
 *//using StringBuilder* StringBuilder strBuilder = new StringBuilder(14);  
 strBuilder.append(first).append(" ").append(last);  
 System.*out*.println(strBuilder.toString());  
  
 *//using StringBuffer* StringBuffer strBuffer = new StringBuffer(15);  
 strBuffer.append(first).append(" ").append(last);  
 System.*out*.println(strBuffer.toString());  
 }  
}

public class StringContainsExample {  
  
 public static void main(String args[]) {  
  
 String s = "Hello Java";  
 System.*out*.println(s.contains("Hello Java"));*// returns true* System.*out*.println(s.contains("Java"));*//true as Java is present in the given String* System.*out*.println(s.contains("hello Java"));*//false - as contains() method is case sensitive.* System.*out*.println(s.contains("o"));*//true* System.*out*.println(s.contains("aa"));*//false as 'aa' is not present,  
  
 //if else example* String str = "Java String example";  
 if (str.contains("example")) {  
 System.*out*.println("Success - String found");  
 } else {  
 System.*out*.println("Failed - String not found");  
 }  
 }  
}

*/\*Strings are immutable :-  
Immutable means changes made to existing object will not be affected to the object.  
If we want to make any changes to the existing object, we need to create a new object,  
so that the changes are affected to the new object.  
 Example:\*/*import org.apache.xmlbeans.impl.xb.xsdschema.Public;  
  
public class StringImmutable {  
 public static void main(String[] args) {  
 String s= new String("Hello");  
 s.concat("Java");  
 System.*out*.println(s);*// prints Hello  
  
 /\*  
In the above program we are concatenating the string object but concatenation  
is not done to the object 's' and prints just 'Hello' but not 'Hello Java'.  
\*/ //The same can be achieved by using the below :* String S= new String("Hello");  
 String s1;  
 s1=S.concat("Java");  
 System.*out*.println(S);*// prints Hello* System.*out*.println(s1);*// prints Hello Java  
 /\* Difference between String literal and creating String object with new operator:  
 String s="Hello"; Here compiler creates a single String object in String Constant Pool with the value "Hello"\*/* }  
}

public class StringReplace {  
 public static void main(String[] args) {  
 String s = "Hello Java";  
  
 System.*out*.println(s.replace(" ", "").length());  
 System.*out*.println(s.replace(" ", ""));  
 *// returns integer number as 9* }  
}

public class StringSplitExample {  
  
 public static void main(String[] args) {  
 String s = "Java String Split Example";  
 String[] str1 = s.split(" ");  
 for (int i = 0; i < str1.length; i++) {  
 System.*out*.println(str1[i]); *// output  
 //Java  
 //String  
 //Split  
 //Example* }  
  
 System.*out*.println("Split with limit , will split the string based on the limit we specify");  
 *//Split with limit -  
 //it will split the string based on the limit we specify* String[] str2 = s.split(" ", 3);  
 for (int i = 0; i < str2.length; i++) {  
 System.*out*.println(str2[i]); *// output  
 // Java  
 //String Split Example* }  
 }  
 }

public class TestAppend {  
 public static void main(String args[]) {  
 StringBuffer sBuffer = new StringBuffer("test");  
 sBuffer.append(" String Buffer");  
 System.*out*.println(sBuffer);  
 }  
 }

Super keyword is used to call the parent class constructor with matching arguments.  
 Super keyword should always be written as the first statement in the child class constructor.  
 Since it is always the first statement, there can be no 2nd Super() keyword in the child class.  
  
 The super keyword refers to superclass (parent) objects. It is used to call superclass methods,  
 and to access the superclass constructor. The most common use of the super keyword  
 is to eliminate the confusion between superclasses and subclasses that have methods  
 with the same name.  
   
 Now to understand the concept of Super() keyword.   
 The Super() keyword is used to call parent class constructor with matching arguments.

public class A {  
  
 public A(){  
 System.*out*.println("Parent class constructor");  
 }  
  
 public A(int i){  
 System.*out*.println("Parent class constructor with value of variable i = "+i);  
  
 }  
}

public class B extends A{ *//B is child of A and also inherits it  
 //Now if I want to call the parent class constructor A(int i)  
 // from B Class, we use the Super keyword* public B(){  
 super(101); *//Calling the parent class constructor with matching arguments.* System.*out*.println("Child class constructor");  
 }  
  
 public B(int i){  
 System.*out*.println("Child class constructor with value of variable i = ");  
 }  
  
 public static void main(String[] args) {  
 B obj = new B();  
 }  
}

BASIC GIT COMMANDS

