**public** **class** StringReverse {

**public** **void** stringReverse() {

String name="Sun is fun";

String t="";

**for** (**int** i=name.length()-1;i>=0;i--) {

t=t+name.charAt(i);

}

System.***out***.println(name);

System.***out***.println(t);

}

}

**public** **class** ReverseString {

**public** **static** **void** main(String[] args) {

String str = "aabbccddaabbccddssnniizz";

**char**[] strArr = str.toCharArray();

**int** strLen = strArr.length;

**char**[] revArr = **new** **char**[strLen];

**for**(**int** i = strLen; i>0; i--) {

revArr[strLen-i] = strArr[i-1];

}

System.***out***.println(revArr);

}

}

**public** **class** FirstNonRepeatedCharacter {

**public** **void** firstNonRepeatedChar() {

String str = "jvbsdchEShfsivncsfhhcsdjkch";

Hashtable<Character, Integer> ht = **new** Hashtable<Character, Integer> ();

Character c;

**for** (**int** i=0;i<str.length();i++) {

c=str.charAt(i);

**if**(ht.containsKey(c)) {

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

}

**else** {

ht.put(c, 1);

}

}

**for** (**int** i=0;i<str.length();i++) {

c=str.charAt(i);

**if** (ht.get(c)==1) {

System.***out***.println("First non repeated character is "+ c);

**break**;

}

}

}

}

**public** **class** StringHasUniqueCharacters {

**public** **void** stringIsUnique() {

String input = "Find me";

**boolean** flag=**true**;

**for**(**int** i=0; i < input.length();i++)

{

**char** charcterofinputstring=input.charAt(i);

**int** count=0;

**for**(**int** j=i; j < input.length();j++)

{

**if** (charcterofinputstring==input.charAt(j))

count++;

}

**if**(count > 1)

flag = **false**;

}

**if**(flag)

System.***out***.println("String has all unique characters");

**else**

System.***out***.println("String does not have all unique characters");

}

}

**public** **class** RemoveDuplicatesChar {

**public** **static** **void** main(String[] args)

{

String str = "aabbccddaabbccddssnniizz";

**char** strArr[] = str.toCharArray();

**int** n = strArr.length;

**char**[] strArrNew = **new** **char**[n];

**int** k=0;

**for** (**int** i=0;i<n;i++) {

**int** j;

**for** (j=0;j<i;j++) {

**if**(strArr[i]==strArr[j]) {

**break**;

}

}

**if** (i==j) {

strArrNew[k++] = strArr[i];

}

}

System.***out***.println(strArrNew);

}

}

**public** **class** Palindrome {

**public** **void** palindrome() {

String str="Civic";

String revstr="";

**for**(**int** i=str.length()-1;i>=0;i--) {

revstr = revstr+str.charAt(i);

}

**if** (str.equalsIgnoreCase(revstr)) {

System.***out***.println("String is a palindrome");

}

**else** {

System.***out***.println("String is not a palindrome");

}

}

}

**public** **class** IntersectionOfArrays {

**public** **void** intersection() {

**int**[] arr1= {1,4,7,9,2,3,8};

**int**[] arr2= {1,7,3,4,8,5};

ArrayList<Integer> intersect= **new** ArrayList<Integer>();

**for**(**int** i = 0; i<arr1.length;i++) {

**for**(**int** j=0; j<arr2.length;j++) {

**if**(arr1[i]==arr2[j]) {

intersect.add(arr1[i]);

}

}

}

System.***out***.println(intersect);

}

}

**public** **class** NumberPrime {

**public** **static** **void** main(String[] args) {

Scanner sc = **new** Scanner(System.***in***);

System.***out***.print("Enter any number: ");

**int** number = sc.nextInt();

**boolean** flag = **true**;

**if** (number == 0 || number == 1)

System.***out***.println(number + " is not prime");

**int** m = number / 2;

**for** (**int** i = 2; i <= m; i++) {

**if** (number % i == 0) {

flag = **false**;

**break**;

}

}

**if** (flag)

System.***out***.println(number + " is prime");

**else**

System.***out***.println(number + " is not prime");

}

}

**public** **class** NumberInArray {

**public** **int** findPosition(**int** num, **int**[] nos) {

**int** position =0;

**for**(**int** i=0; i< nos.length; i++) {

**if** (nos[i]==num) {

position = i+1;

**break**;

}

**else**

position = -1;

}

**return** position;

}

**public** **static** **void** main(String[] args) {

NumberInArray obj = **new** NumberInArray();

**int** numArray[] = {3,6,8,1,5,9,2,4,0,7};

**int** presentAt = 0;

**int** numToCheck = 4;

presentAt = obj.findPosition(numToCheck,numArray);

**if** (presentAt == -1)

System.***out***.println(numToCheck + " is not present");

**else**

System.***out***.println(numToCheck + " is present at " + presentAt + " position");

}

}

**public** **void** takeScreenshotSavetoFolder() {

File errorScreenshot = ((TakesScreenshot) driver).getScreenshotAs(OutputType.***FILE***);

**try** {

FileUtils.*copyFile*(errorScreenshot, **new** File("/Users/nimishajain/Desktop/errorScr.png"));

} **catch** (IOException e) {

System.***out***.println(e.getMessage());

}

}

**public** **void** select() {

WebElement day\_dropdown = driver.findElement(By.*xpath*("//select[@id='day']"));

Select daySelect = **new** Select(day\_dropdown);

daySelect.selectByValue("12");

WebElement month\_dropdown = driver.findElement(By.*xpath*("//select[@id='month']"));

Select monthSelect = **new** Select(month\_dropdown);

monthSelect.selectByVisibleText("Aug");

WebElement year\_dropdown = driver.findElement(By.*xpath*("//select[@id='year']"));

Select yearSelect = **new** Select(year\_dropdown);

yearSelect.selectByVisibleText("1980");

}