1. Find the count of each character

public class CountChars

{

public static void main(String[] args)

{

String s = "tonu in CAB janu";

char ch = 'z';

countChars(s,ch);

}

static void countChars(String s,char c)

{

char[] ch = s.toCharArray();

int[] a = new int[123];

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

a[ch[i]]++;

for(int i='A';i<='Z';i++)

{

if(a[i]!=0)

System.out.println((char)i+"->"+a[i]);

int lc = i+32;

if(a[lc]!=0)

System.out.println((char)lc+"->"+a[lc]);

}

System.out.println("-------------");

System.out.println(c+"->"+a[c]);

}

}

1. Count of different types of characters in the sentence

public class CountProgram

{

public static void main(String[] args)

{

int u\_count=0,l\_count=0,n\_count=0;

String s = "Tonu Monu went to SCHOOL on 28-9-2018";

char[] ch = s.toCharArray();

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

{

if(ch[i]>='A'&&ch[i]<='Z') u\_count++;

if(ch[i]>=97&&ch[i]<=122) l\_count++;

if(ch[i]>=48&&ch[i]<='9') n\_count++;

}

System.out.println("Uppercase:"+u\_count);

System.out.println("Lowercase:"+l\_count);

System.out.println("Numbers:"+n\_count);

// System.out.println('A'+0);

// System.out.println('Z'+0);

// System.out.println('a'+0);

// System.out.println('z'+0);

// System.out.println((int)'0');

// System.out.println('9'+0);

}

}

1. Count words

public class CountWords

{

public static void main(String[] args)

{

String s = " we are coding ";

char[] ch = s.toCharArray();

int count =0;

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

{

if((i==0&&ch[i]!=' ')||(ch[i]!=' '&&ch[i-1]==' '))

count++;

}

System.out.println("Word count = "+count);

}

}

1. Display each word in the new line

public class DisplayWord

{

public static void main(String[] args)

{

String s = " we are coding";

char[] ch = s.toCharArray();

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

{

if((i==0&&ch[i]!=' ')||(ch[i]!=' '&&ch[i-1]==' '))

{

displayWord(i,ch);

}

}

}

static void displayWord(int start,char[] ch)

{

for(;start<ch.length && ch[start]!=' ';start++)

System.out.print(ch[start]);

System.out.println();

}

}

1. Find the character

public class FindChar

{

public static void main(String[] args)

{

String s = "programming";

char key = 'm';

int index = -1;

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

{

if(s.charAt(i)==key)

{

index = i;

//break; //for first occurrence

}

}

System.out.println(index);

}

}

1. Finding substring

public class FindSubString

{

public static void main(String[] args)

{

String s = "aacaa aacaa";

String s1 = "a";

int count =0;

char[] ch = s.toCharArray();

char[] ch1 = s1.toCharArray();

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

{

if(ch[i]==ch1[0])

{

int j=0;

while(j<ch1.length&& (i+j)<ch.length)

{

if(ch[i+j]!=ch1[j])

break;

j++;

}

if(j==ch1.length)

count++;

}

}

System.out.println(count);

}

}

1. Finding the word

public class FindWord

{

public static void main(String[] args)

{

String s = "Chitradurga is also known as durga";

String s1 = "Chitradurga";

int count =0;

char[] ch = s.toCharArray();

char[] ch1 = s1.toCharArray();

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

{

if((i==0&&ch[i]!=' ')||(ch[i]!=' '&&ch[i-1]==' '))

{

count++;

if(ch[i]==ch1[0])

{

int j=0;

while(j<ch1.length&& (i+j)<ch.length)

{

if(ch[i+j]!=ch1[j])

break;

j++;

}

if(j==ch1.length)

System.out.println("Word found at "+count);

else

System.out.println("Word Not found..");

// System.out.println(j);

}

}

} } }

1. Find the sentence is Panagram

public class Panagram

{

public static void main(String[] args)

{

String s = "Te quick brown fox jumps over la dog";

if(validatePanagram(s))

System.out.println("Panagram");

else

System.out.println("Not Panagram");

}

static boolean validatePanagram(String s)

{

char[] ch = s.toCharArray();

int[] a = new int[123];

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

a[ch[i]]++;

for(int i='A';i<='Z';i++)

{

int lc = i+32;

if(a[i]+a[lc]==0)

{

System.out.println((char)i+" is not present");

return false;

}

}

return true;//if all chars available

}

}

1. Do reversing of each word

public class ReverseEachWord

{

public static void main(String[] args)

{

String s = "tonu monu sonu janu";

char[] ch = s.toCharArray();

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

{

//start index of each word

if((i==0&&ch[i]!=' ')||(ch[i]!=' '&&ch[i-1]==' '))

{

findEndIndex(i,ch);

}

}

System.out.println(new String(ch));

}

static void findEndIndex(int start,char[] ch)

{

int end=start;

//ending index of each word

for(;end<ch.length && ch[end]!=' ';end++);

doReverse(start,end-1,ch);

}

static void doReverse(int start, int end, char[] ch)

{

while(start<ch.length && start<end)

{

char t = ch[start];

ch[start] = ch[end];

ch[end] = t;

start++;

end--;

}

}

}

1. Do String reverse

public class StringReverse

{

public static void main(String[] args) {

String s = "Tonu monu Sonu";

s = reverseString(s);

System.out.println(s);

}

static String reverseString(String s)

{

char[] ch = s.toCharArray();

for(int i=0;i<ch.length/2;i++)

{

char t = ch[i];

ch[i] = ch[ch.length-1-i];

ch[ch.length-1-i] = t;

}

return new String(ch);

}

}

1. Convert the given sentence to uppercase

public class UpperCase

{

public static void main(String[] args) {

String s = "convert ME to uppercase";

s = toUpperCase(s);

System.out.println(s);

}

static String toUpperCase(String s)

{

char[] ch = s.toCharArray();

//logic of conversion

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

{

if(ch[i]>=97 && ch[i]<=122)

ch[i] = (char)(ch[i]-32);

}

return new String(ch);

}

}

1. Count of character in each word

public class WordCharCount

{

public static void main(String[] args)

{

String s = "I Love My India";

String res="";

char[] ch = s.toCharArray();

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

{

if((i==0&&ch[i]!=' ')||(ch[i]!=' '&&ch[i-1]==' '))

{

int j=i;

while(i<ch.length&&ch[i]!=' ')

res = res+ch[i++];

res=res+(i-j);

// --i;

//i=j;

// countWord(ch,i);

}

}

System.out.println(res);

}

}

1. Display the palindrome words

public class WordPalindrome

{

public static void main(String[] args)

{

String s = "madam";

int count=0;

char[] ch = s.toCharArray();

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

{

//start index of each word

if((i==0&&ch[i]!=' ')||(ch[i]!=' '&&ch[i-1]==' '))

{

if(count==0)

{

findEndIndex(i,ch);

}

count++;

}

}

//System.out.println(new String(ch));

}

static void findEndIndex(int start,char[] ch)

{

int end=start;

//ending index of each word

for(;end<ch.length && ch[end]!=' ';end++);

if(checkPalindrome(start,end-1,ch))

System.out.println("Palindrome");

else

System.out.println("Not Palindrome");

}

static boolean checkPalindrome(int start, int end, char[] ch)

{

while(start<ch.length && start<end)

{

if(ch[start]!=ch[end])

return false;

start++;

end--;

}

return true;

}

}