**Vuram Grooming:**

1. Find the reminder without using % operator?

**class** DivWithOutDiv {

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

**int** a=3,b=2,r,q;

q=a/b;

r=a-(q\*b);

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

}

}

1. Find out the sum without using + operator?

**class** AddTowWithoutPlusOp {

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

**int** a = 3, b = 2, c;

**while** (b != 0) {

c = a & b;

a = a ^ b;

b = c << 1;

}

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

}

}

1. Find out sub without –

**class** SubTowWithoutPlusOp {

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

**int** a = 3, b = 2, c;

**while** (b != 0) {

c = (~a) & b;

a = a ^ b;

b = c << 1;

}

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

}

}

1. Leap Year

**class** LeapYear {

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

**for** (**int** i = 100; i <=500; i++) {

**if** ((i%4==0&&i%100!=0)||i%400==0) {

System.***out***.println(i+" is a leap Year");

}

}

}

}

1. Multiply without using \*

**class** MultiplyWithout {

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

**int** a = 2, b = 3;

**int** m = 0;

**for** (**int** i = 0; i < 3; i++) {

m = m + a;

}

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

}

}

1. Duplicate Character find out?

**class** CharFreq {

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

String str = "allahah";

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

**for** (**int** i = 0; i < ch.length - 1; i++) {

**int** count = 1;

**for** (**int** j = i + 1; j < ch.length; j++) {

**if** (ch[j] != '\*') {

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

count++;

ch[j] = '\*';

}

}

}

**if** (ch[i] != '\*') {

System.***out***.println(ch[i] + "-->" + count);

}

}

}

}

OR

**class** CharcterFrequency {

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

String str = "allahaaal";

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

**int** n = ch.length;

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

**int** count = 1;

**for** (**int** j = i + 1; j < n; j++) {

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

count++;

**int** k = j;

**while** (k < n - 1) {

ch[k] = ch[k + 1];

k++;

}

j--;

n--;

}

}

System.***out***.println(ch[i] + "-->" + count);

}

}

}

1. Factorial?

**class** Factorialfor {

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

**int** n = 5;

**int** s = 1;

**for** (**int** i = 5; i > 0; i--) {

s = s \* i;

}

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

}

}

OR Using Recursive

**class** Factorial {

**static** **int** rec(**int** n) {

**int** r = 1;

**if** (n == 1 || n == 0) {

**return** 1;

}

**if** (n > 1) {

r = n \* *rec*(n - 1);

}

**return** r;

}

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

System.***out***.println(*rec*(5));

}

}

1. Armstrong?

**class** ArmStrong {

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

**int** n = 154;

**int** t = n;

**int** r, sum = 0;

**while** (t > 0) {

r = t % 10;

sum = sum + r \* r \* r;

t = t / 10;

}

System.***out***.println(sum == n);

}

}

**class** Palindrom {

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

**int** n = 121;

**int** t = n;

**int** sum = 0;

**while** (t > 0) {

**int** r = t % 10;

sum = 10 \* sum + r;

t = t / 10;

}

**if** (sum == n) {

System.***out***.println("palindrom");

} **else** {

System.***out***.println("not palindrom");

}

}

}

1. Revers String Recursiv

**class** RevString {

**static** String rec(String str) {

**if** (str.length() == 1)

**return** str;

**return** (str.charAt(str.length() - 1)) + *rec*(str.substring(0, str.length() - 1));

}

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

System.***out***.println(*rec*("azam"));

}

}

1. Febonaci

**class** Febonaci {

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

**int** f=0,s=1,r;

System.***out***.print(f+" "+s+" ");

**for** (**int** j = 1; j <=10; j++) {

r=f+s;f=s;s=r;

System.***out***.print(r+" ");

}

}

}

OR through recursive:

**class** FebRec1 {

**static** **int** rec(**int** n) {

**if** (n == 0 || n == 1) {

**return** n;

} **else** {

**return** *rec*(n - 1) + *rec*(n - 2);

}

}

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

**for** (**int** i = 0; i <= 10; i++) {

System.***out***.print(*rec*(i) + " ");

}

}

}

OR

**class** FebRec {

**static** **void** rec(**int** f, **int** s, **int** c) {

**if** (c > 0) {

**int** r = f + s;

System.***out***.print(r + " ");

*rec*(f = s, s = r, c - 1);

} **else** {

**return**;

}

}

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

**int** f = 0, s = 1, c = 10;

System.***out***.print(f + " " + s + " ");

*rec*(f, s, c);

}

}

1. Odd even without using Mode

OddEvenWithOutMode {

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

**int** n=2;

**if** ((n&1)==0) {

System.***out***.println("even");

}**else**{

System.***out***.println("odd");

}

}

}

1. Bubble Sort

**class** BubbleSort {

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

**int**[] arr={1,4,3,6,8,2};

**for** (**int** i = 1; i < arr.length; i++) {

**for** (**int** j = 0; j < arr.length-1; j++) {

**if** (arr[j]>arr[j+1]) {

**int** t;

t=arr[j];

arr[j]=arr[j+1];

arr[j+1]=t;

}

}

}

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

System.***out***.print(arr[i]+" ");

}

}

}

1. Largest no

**class** BubbleSort {

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

**int**[] arr={1,4,3,6,8,2};

**for** (**int** i = 1; i < arr.length; i++) {

**for** (**int** j = 0; j < arr.length-1; j++) {

**if** (arr[j]>arr[j+1]) {

**int** t;

t=arr[j];

arr[j]=arr[j+1];

arr[j+1]=t;

}

}

}

System.***out***.print(arr[arr.length-1]+" largest no ");

}

}

1. Reverse the no, prime no, sort the no

**class** NuberArray {

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

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

System.***out***.println("Enter your no: ");

String input = sc.nextLine();

**long** num = Long.*parseLong*(input);

**int**[] arr = **new** **int**[input.length()];

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

arr[i] = (**int**) (num % 10);

num = num / 10;

}

System.***out***.println("Putting the no in an array: ");

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

System.***out***.println(arr[i] + " ");

}

System.***out***.println("Now Sorting the array ");

**for** (**int** i = 1; i < arr.length; i++) {

**for** (**int** j = 0; j < arr.length - 1; j++) {

**if** (arr[j] < arr[j + 1]) {

**int** temp;

temp = arr[j];

arr[j] = arr[j + 1];

arr[j + 1] = temp;

}

}

}

**long** k = 0;

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

k = k \* 10 + arr[i];

}

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

// add the array no

**int** add = 0;

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

add = add + arr[i];

}

System.***out***.println("added array is: " + add);

System.***out***.println("now chek whether no is prime or not");

**int** count = 0;

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

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

count++;

}

}

**if** (count > 0) {

System.***out***.println(add + " is not a prime no");

} **else** {

System.***out***.println(add + " is a prime no");

}

sc.close();

}

}

1. Strong no

**class** StrongNO {

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

**int** n=145;

**int** t=n;

**int** sum=0,fact,r;

**while** (n>0) {

r=n%10;

fact=1;

**while** (r>0) {

fact=fact\*r;

r--;

}

sum=sum+fact;

n=n/10;

}

**if** (sum==t) {

System.***out***.println("s");

}**else** {

System.***out***.println("ns");

}

}

}

1. Perfect no

**class** PerfectNo {

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

**int** n = 28;

**int** r = 0;

**for** (**int** i = 1; i <= n / 2; i++) {

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

r = r + i;

}

}

**if** (r == n) {

System.***out***.println("per");

} **else** {

System.***out***.println("no");

}

}

}

1. Swap tow no without using temp

**class** SwapWithoutTemp {

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

**int** a=2,b=3;

a=a+b;

b=a-b;

a=a-b;

System.***out***.println(a+" "+b);

}

}

1. Replace character with “\*\*”

ReplaceCharacter {

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

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

System.***out***.print("Enter the String :");

String str=sc.next();

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

System.***out***.println("Enter the character to replace");

String s=sc.next();

**char** c[]=s.toCharArray();

str="";

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

**if** (ch[i]==c[0]) {

str=str+"\*\*";

}**else** {

str=str+ch[i];

}

}

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

}

1. **Count Vowels,Characters,constants,SpeciallCharacters,WordCount**

**class** CountVowels {

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

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

System.***err***.println("Enter your String");

String st=sc.nextLine();

**char** ch[]=st.toCharArray();

**int** vc=0,cc=0,dc=0,spc=0,wc=1;

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

**if**((ch[i]>=65&&ch[i]<=122)||(ch[i]<=65&&ch[i]>=90))

{

**if**(ch[i]=='a'|| ch[i]=='b' ||ch[i]=='c'||ch[i]=='d'||ch[i]=='e'||

ch[i]=='A'||ch[i]=='B'||ch[i]=='C'||ch[i]=='D'||ch[i]=='E')

vc++;

**else**

cc++;

}

**else** **if**(ch[i]>=48&&ch[i]<=57)

dc++;

**else** **if**(ch[i]==32&&ch[i+1]!=32)

wc++;

**else**

spc++;

}

System.***out***.println("vowels="+vc);

System.***out***.println("consonents="+cc);

System.***out***.println("digits="+dc);

System.***out***.println("special characters="+spc);

System.***out***.println("word count="+wc);

}

}

1. **UpperCase or LowerCase The Characters**

**class** UpperCaseToLowerCase {

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

String str="Azam and Tousif is good in Technical";

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

**for** (**int** i = 0; i < ch.length-1; i++) {

**if** (ch[i]>=65&&ch[i]<=90) {

ch[i]=(**char**) (ch[i]+32);

}**else** **if** (ch[i]>=97&&ch[i]<=122){

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

}

}

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

System.***out***.print(ch[i]);

}

}

}

1. **CamelCase**

**class** InitCap {

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

String str = "azam and tousif is good in android also";

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

str="";

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

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

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

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

}

**else** **if** (ch[i]>=65&&ch[i]<=90)

ch[i]=(**char**) (ch[i]+32);

str=str+ch[i];

}

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

}

}

23.**Reverse Words**

**class** ReverseWords {

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

String str="Azam and Tousif is Doing Program";

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

str="";

String temp="";

**int** count=0;

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

**if** (ch[i]!=' ') {

temp=temp+ch[i];

}**else** **if** (ch[i-1]!=' ') {

str=temp+" "+str;

temp="";

}**if** (i==ch.length-1) {

str=temp+" "+str;

}

}

System.***out***.print(str);

}

}

24.**WordCount**

**class** WordCount9 {

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

String str= "Jspider and Qspider";

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

**int** count=0;

str="";

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

**if** (ch[i]!=' ') {

count++;//7

str=str+ch[i];//jspider

}**else** **if** (ch[i-1]!=' ') {

str=str+count+' ';//jspider7

count=0;

}

**if** (i==ch.length-1) {

str=str+count;

}

}

System.***out***.print(str);

}

}