**DAY 3**

**1.isEven?**

return input1%2==0?2:1;

**2.isOdd?**

return input1%2!=0?2:1;

**3. Return last digit of the number**

return Math.abs(input1%10);

**4. Return second last digit of the number**

if (input1<0)

         input1=input1\*(-1);

         String s=Integer.toString(input1);

         if(s.length()==1)

         return -1;

         String str=s.substring(s.length()-2,s.length()-1);

         int n=Integer.parseInt(str);

         return n;

**5. Return third last digit of the number**

input1=Math.abs(input1);

        input2=Math.abs(input2);

        return input1%10+input2%10;

**DAY 4**

**1.Is N an exact multiple of M?**

input1=Math.abs(input1)

if(input1==0 || input2==0)

      return 3;

 return input1%input2==0?2:1;

**2.Of given 5,how many are even?**

int count=0;

        if(input1%2==0)

        count++;

        if(input2%2==0)

        count++;

        if(input3%2==0)

        count++;

        if(input4%2==0)

        count++;

         if(input5%2==0)

        count++;

        return count;

**4. Of given 5,how many are odd?**

int count=0;

if(input1%2!=0)

count++;

if(input2%2!=0)

count++;

if(input3%2!=0)

count++;

if(input4%2!=0)

count++;

if(input5%2!=0)

count++;

return count;

**5.of given 5,how many odd and even?**

        if(input6.equals("even")){

        int even=0;

if(input1%2==0)

even++;

if(input2%2==0)

even++;

if(input3%2==0)

even++;

if(input4%2==0)

even++;

if(input5%2==0)

even++;

        return even;

}

else{

int odd=0;

if(input1%2!=0)

odd++;

if(input2%2!=0)

odd++;

if(input3%2!=0)

odd++;

if(input4%2!=0)

odd++;

if(input5%2!=0)

odd++;

return

odd;

    }

**DAY 6**

**1.IsPrime?**

int count=0;

        for(int i=2;i<=Math.sqrt(input1);i++){

            if(input1%i==0){

                    count=1;

                    break;

                }

        }

        if(count==0)

        return 2;

        else

        return 1;

**2.Factorial of a number**

int fact=1;

        for(int i=1;i<=input1;i++){

            fact=fact\*i;

        }

        return fact;

**3.Nth Fibonnoci**

int a=0;

int b=1;

int c=0;

int i=3;

while(i<=input1){

c=a+b;

a=b;

b=c;

i++;

}

return c;

**4.Nth Prime**

int num=1, count=0, i;

        while (count < input1){

  num=num+1;

  for (i = 2; i <= num; i++){

    if (num % i == 0) {

      break;

    }

  }

if ( i == num){

    count = count+1;

}

}

return num;

**DAY 7**

**1.Number of prime numbers with in a specified range.**

int count=0,flag=0;

while(input1<=input2)

{

count=0;

for(int i=2;i<input1;i++){

if(input1%i==0){

count=1;

}

}

if(count==0)

flag++;

input1++;

}

return flag;

**2.All Digits Count**

String str=Integer.toString(input1);

return str.length();

**3.Unique Digits count**

String s=Integer.toString(input1);

        String str = "";

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

if (str.indexOf(s.charAt(i)) == - 1){

str= str + s.charAt(i);

}

}

    return str.length();

**4.Non repeating Digits count**

String str=Integer.toString(input1);

        int n=str.length();

        int count=0;

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

int j;

for (j = 0; j < n; j++)

if (i != j && str.charAt(i) == str.charAt(j) )

break;

if (j == n)

count++;

         }

        return count;

**DAY 8**

**1.DigitSum**

   if(input1==0)

        return 0;

        if(input1>0){

         if(input1%9==0)

            return 9;

         else

            return input1%9;

        }

        else{

         if(input1%9==0)

            return -9;

     else

            return input1%9;

        }

**2.Even digit sum**

int even=0,r=0;

        while(input1!=0){

            r=input1%10;

            if(r%2==0)

            even=even+r;

            input1=input1/10;

        }

        return even;

**3.Odd digit sum**

int r=0,odd=0;

        while(input1!=0){

            r=input1%10;

            if(r%2!=0)

            odd+=r;

            input1=input1/10;

        }

        return odd;

**4.Even or Odd Digit count**

int r=0,odd=0,even=0;

while(input1!=0){

r=input1%10;

if(r%2!=0)

odd+=r;

            else

            even+=r;

input1=input1/10;

}

        if(input2.equals("odd"))

return odd;

        else

        return even;

**Day 9**

**1.Is Palindrome number?**

int rev=0,r=0;

        int temp=input1;

        while(input1!=0){

            r=input1%10;

            rev=rev\*10+r;

            input1/=10;

        }

        if(rev==temp)

        return 2;

        else

        return 1;

**2.Is palindrome possible?**

String str=Integer.toString(input1);

        int arr[]=new int[256];

String s=Integer.toString(input1);

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

arr[s.charAt(i)]++;

}

int odd=0;

for(int i=0;i<256;i++){

if((arr[i] & 1)==1)

odd++;

}

if(odd>1)

return 1;

else

return 2;

**3.Create PIN using alpha,beta,gamma**

int o1=input1%10,o2=input2%10,o3=input3%10;

        int t1=(input1/10)%10,t2=(input2/10)%10,t3=(input3/10)%10;

        int h1=(input1/100)%10,h2=(input2/100)%10,h3=(input3/100)%10;

        int o\_min=Math.min(o1,Math.min(o2,o3));

        int t\_min=Math.min(t1,Math.min(t2,t3));

        int h\_min=Math.min(h1,Math.min(h2,h3));

        int th1=Math.max(o1,Math.max(o2,o3));

        int th2=Math.max(t1,Math.max(t2,t3));

        int th3=Math.max(h1,Math.max(h2,h3));

        int t\_max=Math.max(th1,Math.max(th2, th3));

        return t\_max\*1000+h\_min\*100+t\_min\*10+o\_min;

**4.Weight of a hill pattern**

   int result=input2,r=0;

        for(int i=2;i<=input1;i++){

            r+=input2+input3;

            result+=r\*i;

            System.out.println(result);

         r=r-input2;

            }

        return result;

**DAY 10**

**1.Return second word in uppercase**

int count=1;

        for(int i=0;i<input1.length()-1;i++){

            if((input1.charAt(i)==' ') && (input1.charAt(i+1)!=' '))

            count++;

        }

     if(count==1)

return "LESS";

     String str=input1.split(" ")[1];

     return str.toUpperCase();

**2.Is palindrome possible(String)?**

   input1=input1.toLowerCase();

StringBuilder s=new StringBuilder(input1);

        s.reverse();

        String str= s.toString();

        if(input1.equals(str)){

                return 2;

        }

     return 1;

**3.Weight of the string**

int sum=0;

        input1=input1.toLowerCase();

            System.out.println(input1);

        if(input2==0){

            String s=input1.replaceAll("[aeiouEIOU]","");

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

                if(Character.isAlphabetic(s.charAt(i))){

                    sum=sum+(int)(s.charAt(i))-(int)'a'+1;

                    }

            }

            return sum;

        }

        else{

            for(int i=0;i<input1.length();i++){

                if(Character.isAlphabetic(input1.charAt(i))){

                    sum=sum+(int)(input1.charAt(i))-(int)'a'+1;

                    }

            }

            return sum;

        }

**4.Most frequent Digit  String**

String s = String.valueOf(input1) + String.valueOf(input2) + String.valueOf(input3) + String.valueOf(input4);

        int[] arr = new int[10];

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

            arr[Integer.parseInt(String.valueOf(s.charAt(i)))]++;

        }

        int max = 0;

        for (int i = 0; i <= 9; i++) {

            //System.out.println(arr[max]);

            max = arr[i] >= arr[max] ? i : max;

        }

        return max;