NAME : SHASHIDHAR   MOB.NO.:6361662695

1. WAJP TO REVERSE THE GIVRN NUMBER

```
public class Revers {

      public static void main(String[] args) {
            int rev=0;
            int a=123;
            while(a!=0) {
                  int rem=a%10;
                  rev=rev*10+rem;
                  a=a/10;
            }
      System.out.println(rev);
      }
}
```

2. WAJP TO CHECK THE GIVEN NUMBER IS PALINDROME OR NOT

```
public class Palindrome {

      public static void main(String[] args) {
            int rev=0;
            int a=11211;
            int b=a;
            while(a!=0) {
                  int rem=a%10;
                  rev=rev*10+rem;
                  a=a/10;
            }
        if(rev==b) {
            System.out.println("the entered number is PALINDROME");
        }else {
            System.out.println("the entered number is NOT A PALINDROME");
        }
      }
}
```

3. WAJP TO GET COUNT OF DIGITS IN A GIVEN NUMBER

```
public class Count {

      public static void main(String[] args) {
       int a=7806;
       int count=0;
       while(a!=0) {
             count++;
             a=a/10;
       }
     System.out.println("the number of digits is : "+count);
      }
}
```

4. WAJP TO COUNT THE NUMBER OF EVEN AND ODD DIGITS IN A GIVEN NUMBER

```java
public class CuEvOd {

    public static void main(String[] args) {
        int a=125689;
        int evencount=0;
        int oddcount=0;
        while(a!=0) {
        int rem=a%10;
        if(rem%2==0) {
            evencount++;
        }else {
            oddcount++;
        }
        a=a/10;
    }
    System.out.println("EVEN COUNT : "+evencount+" ODD COUNT : "+oddcount);
    }

    }
```

5. WAJP TO GET THE SUM OF DIGITS IN A GINEN NUMBER

```java
public class Sum {

    public static void main(String[] args) {
        int a=123;
        int sum=0;
        while(a!=0) {
            int rem=a%10;
            sum=sum+rem;
            a=a/10;

    }
        System.out.println(sum);
    }
    }
```

6. WAJP TO PRINT PRIME NUMBERS IN THE GIVEN RANGE

```
public class PrimeRange {

      public static void main(String[] args) {
            int n=100;
            for(int a=1; a<=n; a++) {
                  int count=0;
                  for(int i=1; i<=a; i++) {
                        if(a%i==0) {
                              count++;
                        }
                  }
                  if(count==2) {
                        System.out.println(a);
                  }
            }
      }
      }
```

7. WAJP TO PRINT GIVEN NUMBER IS PRIME OR NOT

```
import java.util.Scanner;
public class PrimeYN {

      public static void main(String[] args) {
            Scanner s1=new Scanner(System.in);
            System.out.println("Enter the number");
     int n=s1.nextInt();
     int count=0;
     for(int i=1; i<=n; i++) {
       if(n%i==0) {
             count++;
             s1.close();
       }
     }
     if(count==2) {
       System.out.println("The given number "+n+" is prime number");
     }else {
       System.out.println("The given number "+n+" is not a prime number");
     }
      }

      }
```

8. WAJP TO SWAP TWO NUMBERS USING TEMP VARIABLE AND WITHOUT TEMP VERIABLE

```
public class Swap {

      public static void main(String[] args) {
       int a=5;
       int b=10;
       System.out.println("original a value : "+a);
       System.out.println("original b value : "+b);
       int t=a; //a=a+b;      //b=a+b-(a=b);
       a=b;      //b=a-b;
       b=t;     //a=a-b;
       System.out.println("Swaped a value : "+a);
       System.out.println("Swaped b value : "+b);
      }
      }
```

9. WAJP TO CHECK THE GIVEN NUMBER IS ARMSTRONG NUMBER OR NOT

```
public class Armstrong {

      public static void main(String[] args) {
            int a=153;
            int b=a;
            int total=0;
            int count=0;
            while(b!=0) {
                  count++;
                  b=b/10;
            }
       int c=a;
       while(c!=0) {
         int eachNumberPower=1;
         int rem=c%10;
         for(int i=1; i<=count; i++) {
               eachNumberPower=eachNumberPower*rem;
         }
         c=c/10;
         total=total+eachNumberPower;
       }
       if(a==total) {
         System.out.println("the given number "+a+" is ARMSTRONG.");
       }
      }

      }
```

10.WAJP TO PRINT ARMSTRONG NUMBERS BETWEEN RANGE

```
public class ArmstrongRang {

      public static void main(String[] args) {
            for(int a=1; a<=1000; a++) {
            int b=a;
            int total=0;
            int count=0;
            while(b!=0) {
                  count++;
                  b=b/10;
            }
       int c=a;
       while(c!=0) {
         int eachNumberPower=1;
         int rem=c%10;
         for(int i=1; i<=count; i++) {
               eachNumberPower=eachNumberPower*rem;
         }
         c=c/10;
         total=total+eachNumberPower;
       }
       if(a==total) {
         System.out.println(a);
       }

      }
      }
   }
```

11.WAJP TO GET FACTORIAL OF A GIVEN NUMBER

```
public class Fact {

      public static void main(String[] args) {
            int a=4;
            int fact=1;
            for(int i=1; i<=a; i++) {
                  fact=fact*i;
            }
      System.out.println(a+"! is "+fact);
      }

   }
```

12.WAJP TO GET FACTORIAL OF A GIVEN RANGE OF NUMBERS

```
public class FactRang {

      public static void main(String[] args) {
            for(int a=1; a<=20; a++) {
            int fact=1;
            for(int i=1; i<=a; i++) {
                  fact=fact*i;
            }
      System.out.println(a+" !   is "+fact);
      }
      }
}
```

13.WAJP TO PRINT FIBANOCI SERIES

```
public class Fibanoci {

      public static void main(String[] args) {
            int n1=0;
            int n2=1;
            System.out.print(n1+" "+n2);
            for(int i=1; i<=7; i++) {
       int sum=n1+n2;
       System.out.print(" "+sum);
       n1=n2;
       n2=sum;
      }
      }
   }
```

14.WAJP TO GET LARGEST OF 3 NUMBERS

```
public class Lar03 {

      public static void main(String[] args) {
            int a=9, b=5, c=3;
            if(a>b) {
                  if(a>c) {
                        System.out.println("a is larger");
                  }
                  else {
                        System.out.println("c is larger");
                  }
            }
            else if(b>c) {
                  System.out.println("b is larger");
            }
            else {
                  System.out.println("c is larger");
            }
      }
   }
```

15.WAJP TO GET LARGEST OF 4 NUMBERS

```java
public class Lrgof4 {

    public static void main(String[] args) {
        int a=70, b=60, c=50, d=40;
        if(a>b) {
            if(a>c) {
                if(a>d) {
                    System.out.println("a is largest");
                }
                else {
                    System.out.println("d is largest");
                }
            }
            else if(c>d) {
                System.out.println("c is largest");
            }
            else {
                System.out.println("d is largest");
            }
        }
        else if(b>c) {
            if(b>d) {
                System.out.println("b is largest");
            }
            else {
                System.out.println("d is largest");
            }
        }
        else if(c>d) {
            System.out.println("c is largest");
        }
        else {
            System.out.println("d is largest");
        }
    }

}
```

STAR PATTERNS

```
* * * * *
* * * * *
* * * * *
* * * * *
* * * * *
```

```java
public class Square {

      public static void main(String[] args) {
            for(int r=1; r<=5; r++) {
                  for(int c=1; c<=5; c++) {
                        System.out.print("* ");
                  }
                  System.out.println();
            }
      }
}
```

```
* * * * *
* * * *
* * *
* *
*
```

```java
public class InTri {

      public static void main(String[] args) {

            for (int r = 1; r <= 5; r++) {
                  for (int c = r; c <= 5; c++) {
                        System.out.print("* ");
                  }
                  System.out.println();
            }
      }

}
```

```
1
2 1
3 2 1
4 3 2 1
5 4 3 2 1
```

```
public class DecTri {

      public static void main(String[] args) {


      for(int r=1;r<=5;r++) {
                  int a=r;
                  for(int c=1; c<=r; c++) {
                        System.out.print(a+" ");
                        a--;
                  }
            System.out.println();
      }

      }
}
```

```
//TO PRINT S PATTERN
public class Alpha {

      public static void main(String[] args) {
            for(int r=1;r<=5;r++) {
                  for(int c=1;c<=5;c++) {

      if(r==1&&(c>1&&c<5)||r==3&&(c>1&&c<4)||r==5&&(c<4)||r==2&&c==1||r==4&&c
==4) {
                              System.out.print("* ");
                        }
                        else {
                              System.out.print("  ");
                        }
                  }
                  System.out.println();
            }------------------------------------------------------------------------------------------------------------
------------------------

            //TO PRINT P PATTERN
            for(int r=1;r<=7;r++) {
                  for(int c=1;c<=4;c++) {
                        if(c==1||r==1&&c<4||r==4&&c<4||c==4&&(r>1&&r<4)) {
                              System.out.print("* ");
                        }
                        else {
                              System.out.print("  ");
                        }
                  }
                  System.out.println();
            }
----------------------------------------------------------------------------------------------------------------------------------------

            //TO PRINT R PATTERN
```

```
        for(int r=1;r<=7;r++) {
            for(int c=1;c<=4;c++) {
                if(c==1||r==1&&c<4||r==4&&c<4||c==4&&(r>1&&r<4)||(r-
c==3)) {
                    System.out.print("* ");
                }
                else {
                    System.out.print("  ");
                }
            }
            System.out.println();
        }
```

```
        //TO PRINT A PATTERN
        for(int r=1;r<=6;r++) {
            for(int c=1;c<=5;c++) {

    if(c==1&&r>2||c==5&&r>2||r==4||r==1&&c==3||c==2&&r==2||r==2&&c==4) {
                    System.out.print("* ");
                }
                else {
                    System.out.print("  ");
                }
            }
            System.out.println();
        }
    }


}
```

2-D ARRAYA

```
public class Prog1 {

    public static void main(String[] args) {
        int [][]a= {{1,2,3},{4,5,6},{7,8,9}};
        System.out.println("TO PRINT  ELEMENTS OF AN ARRAY");
        System.out.println("-------------------------------------");
        for (int r=0;r<a.length;r++) {
            for(int c=0;c<a[0].length;c++) {
                System.out.print(a[r][c]+" ");
            }
            System.out.println();
        }

        System.out.println("SUBTRACTING ALL ELEMENTS IN AN ARRAY BY 1");
        System.out.println("------------------------------------------");
            for (int r=0;r<a.length;r++) {
                for(int c=0;c<a[0].length;c++) {
                    System.out.print((a[r][c]-1)+" ");
                }
                System.out.println();
            }

     System.out.println("TO PRINT  DIAGONAL ELEMENTS OF AN ARRAY");
      System.out.println("----------------------------------------");
            for (int r=0;r<a.length;r++) {
                  for(int c=0;c<a[0].length;c++) {
                        if(r==c||r+c==2) {
                              System.out.print(a[r][c]+" ");
                        }
                        else {
                              System.out.print("  ");
                        }
                  }
                  System.out.println();
            }
      System.out.println("TO PRINT ELEMENTS ABOVE DIAGONAL");
      System.out.println("-------------------------------------");
            for (int r=0;r<a.length;r++) {
                  for(int c=r;c<a[0].length;c++) {
                        System.out.print((a[r][c])+" ");
                  }
                  System.out.println();
            }

      System.out.println("TO PRINT ELEMENTS BELOW DIAGONAL");
      System.out.println("---------------------------------");
            for (int r=0;r<a.length;r++) {
                  for(int c=0;c<=r;c++) {
                        System.out.print((a[r][c])+" ");
                  }
                  System.out.println();
            }
      System.out.println("TO GET SUM OF ALL ELEMENTS IN AN ARRAY");
      System.out.println("---------------------------------------");
            int sum=0;
            for (int r=0;r<a.length;r++) {
                  for(int c=0;c<a[0].length;c++) {
                        sum=sum+a[r][c];
                  }
            }
            System.out.println(sum);


System.out.println("TO GET SUM OF ALL DIAGONAL ELEMENTS IN AN ARRAY");
System.out.println("------------------------------------------------");
            int sum1=0;
            for (int r=0;r<a.length;r++) {
                  for(int c=0;c<a[0].length;c++) {
                        if(r==c||r+c==2) {
                        sum1=sum1+a[r][c];
                  }
                  }
            }
            System.out.println(sum1);

            System.out.println("TO SWAP THE FIRST AND LAST ROW");
            System.out.println("-------------------------------");
```

```
                  int r1=0;
                  int rn=2;
                  for(int c=0;c<a[0].length;c++) {
                        int temp=a[r1][c];
                        a[r1][c]=a[rn][c];
                        a[rn][c]=temp;
                  }
                  for(int r=0;r<a.length;r++) {
                        for(int c=0;c<a[0].length;c++) {
                              System.out.print(a[r][c]+" ");
                        }
                        System.out.println();
                  }

            System.out.println("TO SWAP FIRST COLUMN TO LAST COLUMN");
            System.out.println("--------------------------------------");
                  int c1=0;
                  int cn=2;
                  for(int r=0; r<a.length;r++) {
                        int temp=a[r][c1];
                        a[r][c1]=a[r][cn];
                        a[r][cn]=temp;
                  }
                  for(int r=0;r<a.length;r++) {
                        for(int c=0;c<a[0].length;c++) {
                              System.out.print(a[r][c]+" ");
                        }
                        System.out.println();
                  }
                  System.out.println("TRANSPOSE OF MATRIX");
                  System.out.println("-----------------------");
                  int [][]b=new int[3][3];
                  for(int r=0;r<a.length;r++) {
                        for(int c=0;c<a[0].length;c++) {
                              b[r][c]=a[c][r];
                        }
                        //System.out.println();
                  }
                  for(int r=0;r<a.length;r++) {
                        for(int c=0;c<a[0].length;c++) {
                              System.out.print(b[r][c]+" ");
                        }
       System.out.println();
            }
      }
}
```

STRINGS PROGRAMS

THE SUM OF DIGITS PRESENT IN THE STRING

```
public class AddDigits {
    public static void main(String[] args) {
        String s = "my123name45";
        char[] c = s.toCharArray();
        int sum = 0;
        for (int i = 0; i < c.length; i++) {
            if (c[i] > '0' && c[i] < '9') {
                sum = sum + (c[i] - 48);
            }
        }
System.out.println("the sum of digits present in the String is : "+sum);
    }

}
```

TO FIND THE ASCI VALUE OF GIVEN CHARACTER

```
public class Asci {

    public static void main(String[] args) {
        char c = 'A';
        int tem = c; // widenning (implicite type casting)

        System.out.println("the asci value of " + c + " is : " + tem);

    }
```

CAPITALIZE EACH WORD FIRST LETTER

```
public class CaptiEacWordFstLett {

    public static void main(String[] args) {
        String s = "trust no one, be only one";
        char[] c = s.toCharArray();
        for (int i = 0; i < c.length; i++) {
            int k = i;
            while (i < c.length && c[i] != ' ') {
                i++;
            }
            if (c[k] >= 'a' && c[k] <= 'z') {
                c[k] = (char) (c[k] - 32);
            }
        }
        String res = new String(c);
        System.out.println(s);
        System.out.println(res);
    }
}----------------------------------------------------------------------------------------------------------------------------------
-
```

TO CHECK THE GIVEN CHAR IS VOWLE OR NOT

```
public class CharVowelORnot {

    public static void main(String[] args) {
        char c = 'E';
    if (c == 'a' || c == 'e' || c == 'i' || c == 'o' || c == 'u' ||
     c == 'A' || c == 'E' || c == 'I' || c == 'O'|| c == 'U') {
            System.out.println("the given char " + c + " is vowel");
        } else {
        System.out.println("the given char " + c + " is not vowel");
        }
        System.out.println("----------------------------------------");
        System.out.println("TO CHECK THE GIVEN CHAR IS ALPHABET");

        if (c >= 'a' && c <= 'z' || c >= 'A' && c <= 'Z') {
            System.out.println(c + " it is a character");
        } else {
            System.out.println(c + " it is not a character");
        }
    }
}
```

WAJP TO CHECK THE GIVEN STRING CONTAINS VOWELS ARE NOT IF CONTAINS PRINT "HI" ELSE PRINT "BYE".

```
public class CheckConteVowels {

    public static void main(String[] args) {
        String s = "abecidoUs";
        char[] c = s.toCharArray();
        int acount = 0, ecount = 0, icount = 0, ocount = 0, ucount = 0;
        for (int i = 0; i < c.length; i++) {
            char choice = c[i];
            switch (choice) {
            case 'a':
                acount++;
                break;
            case 'e':
                ecount++;
                break;
            case 'i':
                icount++;
                break;
            case 'o':
                ocount++;
                break;
            case 'u':
                ucount++;
                break;
            }
        }
if (acount >= 1 && ecount >= 1 && icount >= 1&& ocount >= 1 && ucount >= 1) {
            System.out.println("HI");
        } else {
            System.out.println("BYE");
        }
    }}
```

TO CONVERT UPPER TO LOWER

```
public class ConUpptoLow {

    public static void main(String[] args) {
        String s = "SHAshi";
        String res = "";
        String res1 = "";
        String res2 = "";
        char[] c = s.toCharArray();
        for (int i = 0; i < c.length; i++) {
            // TO CONVERT UPPER CHAR TO LOWER CHAR
            if (c[i] >= 'A' && c[i] <= 'Z') {
                res = res + (char) (c[i] + 32);
            } else {
                res = res + c[i];
            }
            // TO CONVERT LOWER CHAR TO UPPER CHAR
            if (c[i] >= 'a' && c[i] <= 'z') {
                res1 = res1 + (char) (c[i] - 32);
            } else {
                res1 = res1 + c[i];
            }
            // TO CONVERT LOWER CHAR TO UPPER CHAR AND UPPER TO LOWER
            if (c[i] >= 'a' && c[i] <= 'z') {
                res2 = res2 + (char) (c[i] - 32);
            } else {
                res2 = res2 + (char) (c[i] + 32);
            }
        }
    System.out.println("original string : " + s);
    System.out.println("after conversion upper to lower : " + res);
    System.out.println("after conversion lower to upper : " + res1);
    System.out.println("after conversion LowToUp and UpTOLow : " + res2);
    }

}
```

TO COUNT THE NUMBER OF WORDS IN THE GIVEN SENTENCE

```
public class CounNumOFWords {

    public static void main(String[] args) {
        String s = "my name is bond";
        // s=s+" ";
        char[] c = s.toCharArray();
        int count = 1; // count=0;
        for (int i = 0; i < c.length; i++) {
            if (c[i] == ' ') {
                count++;
            }
        }
        System.out.println("sentence is : " + s);
System.out.println("the number of words in given sentence is : " + count);
    }}
```

TO COUNT NUMBER OF VOWELS IN THE GIVEN STRING

```
public class Countvowels {

      public static void main(String[] args) {
            String s="JaiNTR";
       int count=0;
       char [] c=s.toCharArray();
       for(int i=0;i<c.length;i++) {
         if(c[i]=='a'||c[i]=='e'||c[i]=='i'||c[i]=='o'||c[i]=='u'||
                     c[i]=='A'||c[i]=='E'||c[i]=='I'||c[i]=='O'||c[i]=='U') {
               count++;
         }
       }
       System.out.println(count);
       System.out.println("---------------");
       System.out.println("BY USING SWITCH ");
       int count1=0;
       for(int j=c.length-1;j>=0;j--) {
       char a=c[j];
       switch (a) {
      case 'a':
      case 'e':
      case 'i':
      case 'o':
      case 'u':
      case 'A':
      case 'E':
      case 'I':
      case 'O':
      case 'U': count1++;
      break;
      }
      }
       System.out.println(count1);
      }
}
```

1a2b3c4d5e➔abbcccddddeeeee

```
public class PrintLetters {

      public static void main(String[] args) {
            String s = "1a2b3c4d5e";
            char[] c = s.toCharArray();
            String res = "";
            for (int i = 0; i < c.length; i=i+2) {
                  int n = c[i] - 48;
                  for (int j = 0; j < n; j++) {
                        char r = c[i + 1];
                        res = res + r;
                  }
            }
            System.out.println(res);
      }
}
```

TO REMOVE ALL OTHER CHARACTERS EXPET ALPHABET

```
public class RemoveOtherChar {

      public static void main(String[] args) {
            String s = "asmj*&%ghj";
            String res = "";
            char[] c = s.toCharArray();
            for (int i = 0; i < c.length; i++) {
            if (c[i] >= 'a' && c[i] <= 'z' || c[i] >= 'A' && c[i] <= 'Z') {
                        res = res + c[i];
                  }
            }
            System.out.println(res);
      }

}
```

TO REMOVE SPACE IN THE GIVEN STRING

```
public class RemoveSpace {

      public static void main(String[] args) {
            String s="I     AM     GOD";
            String res="";
            String res1="";
            char []c=s.toCharArray();
            for(int i=0;i<c.length;i++) {
                  if(c[i]!=' ') {
                        res=res+c[i];
                  }
                  if(c[i]!=' '||c[i]==' '&&c[i+1]!=' ') {
                        res1=res1+c[i];
                  }
            }
          System.out.println("original string : "+s);
System.out.println("-----------------------------------------");

          System.out.println("TO REMOVE TOTAL SPACES IN A GIVEN STRING");
          System.out.println("changed String : "+res);
System.out.println("-----------------------------------------");

          System.out.println("TO REMOVE EXTRA SPACES IN A GIVEN STRING");
          System.out.println("changed String : "+res1);


      }

}
```

TO REVERSE THE GIVEN STRING

```
public class Reverse {

      public static void main(String[] args) {
            String s= "ssgss";
            String res="";
```

```
        char [] c=s.toCharArray();
        for (int i=c.length-1; i>=0; i--) {
            res=res+c[i];
        }
        System.out.println("the given string is "+s);
        System.out.println("reverse of string");
    System.out.println(res);



    System.out.println("the given string is palindrome or not");
    if(res.equals(s)) {
      System.out.println("it is a palindrome");
    }
    else {
      System.out.println("not palindrome");
    }
    }

}
```

TO REVERSE EACH WORD IN A GIVEN STRING

```
public class ReversEachWord {

    public static void main(String[] args) {
        String s = "revers each word";
        char[] c = s.toCharArray();
        String res = "";
        for (int i = 0; i < c.length; i++) {
            int k = i;
            while (i < c.length && c[i] != ' ') {
                i++;
            }
            for (int j = i - 1; j >= k; j--) {
                res = res + c[j];
            }
            if (i < c.length) {
                res = res + c[i];
            }
        }
        System.out.println(res);
    }

}
```

TO SWAP FIRST AND LAST LETTER IN THE GIVEN STRING

```
public class SwapFirnLastLett {

    public static void main(String[] args) {
        String s = "donot stop";
        char[] c = s.toCharArray();
        for (int i = 0; i < c.length; i++) {
            int k = i;
            while (i < c.length && c[i] != ' ') {
                i++;
            }
            char temp = c[k];
            c[k] = c[i - 1];
            c[i - 1] = temp;
        }
        String res = new String(c);
        System.out.println(s);
        System.out.println(res);
    }

}
```

ARRAYS

WAJP TO INSERT AN ELEMENT AT SPECIFIED INDEX POSITION

```
import java.util.Arrays;

public class InsertEle {

    public static void main(String[] args) {
        int[] a = { 1, 2, 3, 4, 5, 6 };
        int index = 2;
        int element = 50;
        for (int i = a.length - 1; i > index; i--) {
            a[i] = a[i - 1];
        }
        a[index] = element;
        System.out.println(Arrays.toString(a));
    }

}
```

WAJP TO FIND LARGEST ELEMENT IN THE GIVEN ARRAY

```
public class Largest {

    public static void main(String[] args) {
        int[] a = { 2, 6, 9, 5 };
        int max = a[0];
        int min = a[0];
        for (int i = 0; i < a.length; i++) {
            if (a[i] > max) {
                max = a[i];
            }
            if (a[i] < min) {
                min = a[i];
            }
        }
System.out.println("the largest element in the given array is : " + max)
    System.out.println("the least element in the given array is : " + min);
    }
}
```

WAJP TO DELETE AN ELEMENT AT SPECIFIED INDEX POSITION

```
import java.util.Arrays;

public class RemoveEle {

	public static void main(String[] args) {
		int[] a = { 2, 5, 9, 6, 4 };
		int index = 1;
		for (int i = index; i < a.length - 1; i++) {
			a[i] = a[i + 1];
		}
		System.out.println(Arrays.toString(a));
	}

}
```

WAJP TO REVERSE THE GIVEN STRING

```
import java.util.Arrays;

public class ReverseEle {

	public static void main(String[] args) {
		int[] a = { 1, 2, 3, 4 };
		/*
		 * int[] temp = new int[a.length]; int n = 0; for (int i =
a.length - 1; i >= 0;
		 * i--) { temp[n] = a[i]; n++; }
		 */
		for (int i = 0, j = a.length - 1; i < a.length / 2; i++, j--) {
			int temp = a[i];
			a[i] = a[j];
			a[j] = temp;
		}
		System.out.println(Arrays.toString(a));
		// System.out.println(Arrays.toString(temp));
	}

}
```

WAJP TO SUM OF ALL ELEMENTS IN THE GIVEN ARRAY

```
public class SumOEle {

	public static void main(String[] args) {
		int[] a = { 1, 2, 3, 4, 5 };
		int sum = 0;
		for (int i = 0; i < a.length; i++) {
			sum = sum + a[i];
		}
System.out.println("the sum of elements present in an array is : " + sum);
	}
}
```

WAJP TO FIND 1ST AND 2ND LARGEST ELEMENTS IN THE GIVEN ARRAY

```
public class FstSenLarg {

     public static void main(String[] args) {
          int []a= {10,60,55,96,70};
          int max=0;
          int smax=0;
          for (int i=0; i<a.length; i++) {
               if(a[i]>max) {
                    smax=max;
                    max=a[i];
               }
               else if(a[i]>smax) {
                    smax=a[i];
               }
          }
          System.out.println(max);
            System.out.println(smax);
          int min=a[0];
          int smin=a[0];
          for (int j=1; j<a.length; j++) {
               if(a[j]<min) {
                    smin=min;
                    min=a[j];
               }
               else if(a[j]<smin) {
                    smin=a[j];
               }
          }

  System.out.println(min);
  System.out.println(smin);
     }

}
```

WAJP TO SEPARATE EVEN AND ODD ELEMENTS IN THE GIVEN ARRAY

```
package arrays;

import java.util.Arrays;
```

```java
public class Seperste {

    public static void main(String[] args) {
        int[] a = { 3, 5, 2, 7, 8, 76 };
        int evencount = 0;
        int oddcount = 0;
        for (int i = 0; i < a.length; i++) {
            if (a[i] % 2 == 0) {
                evencount++;
            } else {
                oddcount++;
            }
        }
        int[] even = new int[evencount];
        int[] odd = new int[oddcount];
        int n = 0;
        int k = 0;
        for (int i = 0; i < a.length; i++) {
            if (a[i] % 2 == 0) {
                even[n] = a[i];
                n++;
            } else {
                odd[k] = a[i];
                k++;
            }
        }
        System.out.println(Arrays.toString(a));
        System.out.println(Arrays.toString(even));
        System.out.println(Arrays.toString(odd));
    }

}
```

WAJP TO FIND MISSING NUMBER ELEMENTS IN THE GIVEN ARRAY

```java
public class Missing {

    public static void main(String[] args) {
        int[] a = { 1, 3, 4, 5 };
        int sumOfa = 0;
        for (int i = 0; i < a.length; i++) {
            sumOfa = sumOfa + a[i];
        }
        int n = 5;
        int totalsum = n * (n + 1) / 2;
        System.out.println("missing number is : " + (totalsum - sumOfa));
    }

}
```

WAJP TO SORT THE ELEMENTS IN THE GIVEN ARRAY

```
import java.util.Arrays;

public class Sort {

      public static void main(String[] args) {
            int[] a = { 1, 5, 16, 25, 30 };
            System.out.println("TO SORT GIVEN ARRAY IN DECRESING ORDER");
            for (int i = 0; i < a.length; i++) {
                  for (int j = 0; j < a.length; j++) {
                        if (a[i] > a[j]) {
                              int temp = a[i];
                              a[i] = a[j];
                              a[j] = temp;
                        }
                  }
            }
            System.out.println(Arrays.toString(a));
System.out.println("---------------------------------------------------------
--");
            System.out.println("TO SORT GIVEN ARRAY IN INCRESING ORDER");
                        for (int i = 0; i < a.length; i++) {
                              for (int j = 0; j < a.length; j++) {
                                    if (a[i] < a[j]) {
                                          int temp = a[i];
                                          a[i] = a[j];
                                          a[j] = temp;
                                    }
                              }
                        }
                        System.out.println(Arrays.toString(a));
      }
```

WAJP TO CONVERT DECIMAL NUMBER TO BINERY FORM

```
public class Decimal2Binery {

      public static void main(String[] args) {
            int a = 19;
            String res = "";
            while (a != 0) {
                  int rem = a % 2;
                  res = res + rem;
                  a = a / 2;
            }
            for (int i = res.length() - 1; i >= 0; i--) {
                  System.out.print(res.charAt(i));
            }
      }

}
```

WAJP TO BINERY TO DECINAL FORM

```java
public class Binery2Decimal {

	public static void main(String[] args) {
		int a = 1111;
		int sum = 0;
		int n = 0;
		while (a != 0) {
			int rem = a % 10;
			sum = (int) (sum + rem * Math.pow(2, n));
			n++;
			a = a / 10;
		}
		System.out.println(sum);
	}

}
```

WAJP TO PRINT ALTERNATE POSITIVE AND NEGITIVE ELEMENTS IN THE GIVEN ARRAY

```java
import java.util.Arrays;

public class PosNeg {

	public static void main(String[] args) {
		int[] a = { 10, -8, -3, 5, 8, -4 };
		int[] res = new int[a.length];
		int x = 0;
		int y = 1;
		for (int i = 0; i < a.length; i++) {
			if (a[i] >= 0) {
				res[x] = a[i];
				x = x + 2;
			}
		}
		for (int j = 1; j < a.length; j++) {
			if (a[j] < 0) {
				res[y] = a[j];
				y = y + 2;
			}

		}
		System.out.println(Arrays.toString(res));

	}

}
```

WAJP TO SAPERATE 0'S AND 1'S ELEMENTS IN THE GIVEN ARRAY

```
import java.util.Arrays;

public class ZeroOne {

     public static void main(String[] args) {
          int[] a = { 1, 0, 0, 1, 0, 1, 1, 0 };
          int[] res = new int[a.length];
          int x = 0;

          for (int i = 0; i < a.length; i++) {
               if (a[i] == 0) {
                    res[x] = res[x] + a[i];
                    x++;
               }
          }
          for (int j = 0; j < a.length; j++) {
               if (a[j] == 1) {
                    res[x] = res[x] + a[j];
                    x++;
               }

          }

          System.out.println(Arrays.toString(res));
//        Arrays.sort(a);
//        System.out.println(Arrays.toString(a));

     }

}
```

WAJP TO SEPARETE POSITIVE AND NEGITIVE ELEMENTS IN THE GIVEN ARRAY

```
import java.util.Arrays;

public class SepPOSnNEG {

     public static void main(String[] args) {
          int []a= {1,-10,30,-20,8,-3,6};
          int p=0,n=0;
          for (int i = 0; i < a.length; i++) {
               if(a[i]<0)
                    n++;
               else
                    p++;
          }
       int [] pos=new int[p];
       int []neg=new int[n];
       int x=0, y=0;
       for(int i=0; i<a.length; i++) {
        if (a[i]>=0) {
             pos[x]=a[i];
             x++;
        }else {
             neg[y]=a[i];
             y++;
        }
       }
       System.out.println(Arrays.toString(pos));
       System.out.println(Arrays.toString(neg));
       int []res=new int[a.length];
       int z=0;
       for (int i = 0; i < neg.length; i++) {
          res[z]=neg[i];
          z++;
     }
       for (int i = 0; i < pos.length; i++) {
          res[z]=pos[i];
          z++;
     }
       System.out.println(Arrays.toString(res));
     }

}
```

WAJP TO SORT THE ELEMENTS IN THE GIVEN ARRAY

```
import java.util.Arrays;

public class Sort {

      public static void main(String[] args) {
            int[] a = { 1, 5, 16, 25, 30 };
            System.out.println("TO SORT GIVEN ARRAY IN DECRESING ORDER");
            for (int i = 0; i < a.length; i++) {
                  for (int j = 0; j < a.length; j++) {
                        if (a[i] > a[j]) {
                              int temp = a[i];
                              a[i] = a[j];
                              a[j] = temp;
                        }
            }
        }
        System.out.println(Arrays.toString(a));
    System.out.println("---------------------------------------------------
--------");
        System.out.println("TO SORT GIVEN ARRAY IN INCRESING ORDER");
                for (int i = 0; i < a.length; i++) {
                    for (int j = 0; j < a.length; j++) {
                        if (a[i] < a[j]) {
                            int temp = a[i];
                            a[i] = a[j];
                            a[j] = temp;
                        }
                    }
                }
                System.out.println(Arrays.toString(a));
    }

}
```