Array

Q. What is array?

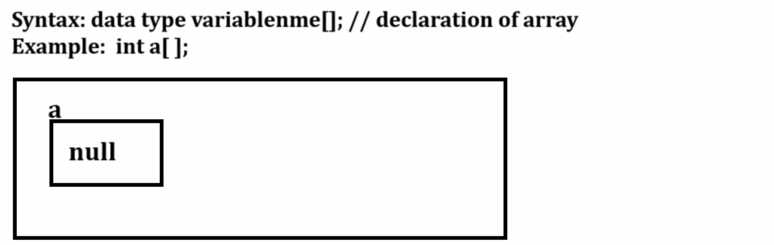
Array is a collection of similar type of data means when we have multiple data of same type then we can use array.

Example: Suppose we want to calculate addition of 100 values then we required to declare 100 variables so declaring 100 variable is not good approach or not possible at real time so we one facility name as array using array we can store 100 values in single variable.

How to use array in JAVA

If we want to use array in java we have two steps

1. Declare array: when we declare array then we not allocate memory of array just declare its reference variable and specify the array type and by default value of array reference variable is null.

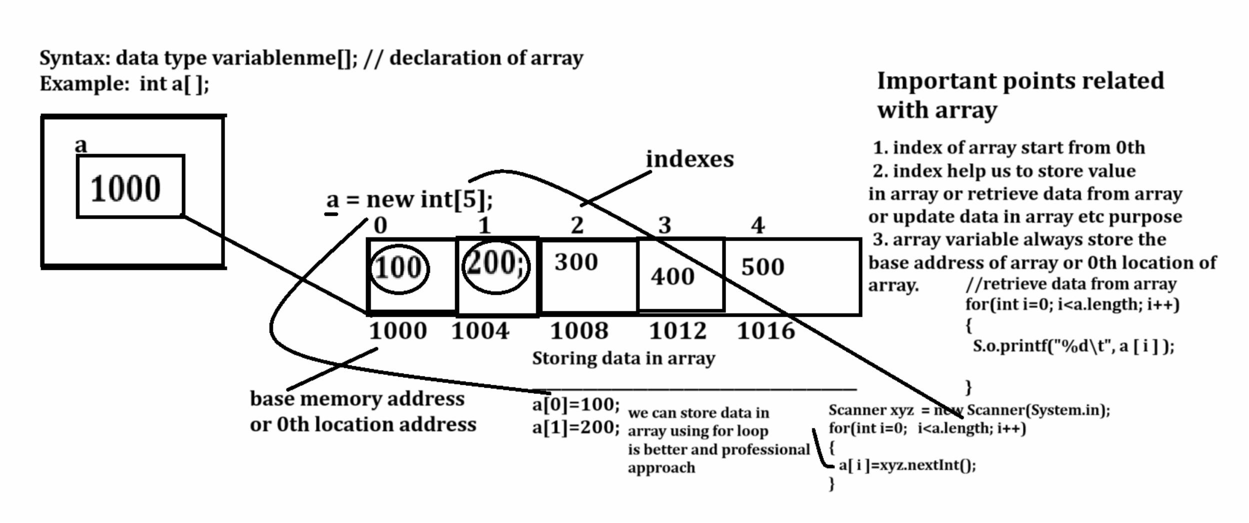


Syntax: data type variablenme[]; // declaration of array
Example: int a[ ];

2. Memory allocation of array: memory allocation decides size of the array means in this specify we allocate memory for array and its size.

Syntax: array variable = new datatype[size];

Example: a = new int[5]; //



Syntax: data type variablenme[]; // declaration of array
Example: int a[ ];

Important points related with array

1. index of array start from 0th
2. index help us to store value in array or retrieve data from array or update data in array etc purpose
3. array variable always store the base address of array or 0th location of array.

```
//retrieve data from array
for(int i=0; i<a.length; i++)
{
  S.o.printf("%d\t", a [ i ] );

}
```

```
a[0]=100;
a[1]=200;
```

we can store data in array using for loop is better and professional approach

```
Scanner xyz  = new Scanner(System.in);
for(int i=0;   i<a.length; i++)
{
 a[ i ]=xyz.nextInt();
}
```

Example with source code

```
import java.util.*;
public class ArrayApp
{
  public static void main(String x[])
  {
      Scanner xyz  = new Scanner(System.in);
                int a[];
                a=new int[5];
                System.out.println("Enter values in array");
                for(int i=0;i<a.length;i++)
                { a[i]=xyz.nextInt();
                }
                System.out.println("Display array values");
                for(int i=0; i<a.length;i++)
                { System.out.printf("%d\t",a[i]);
                }
  }
}
```

Example: WAP to create array of size 5 and find max value from array.

```
import java.util.*;
public class FindingMaxApp
{
  public static void main(String x[])
  {
                Scanner xyz  = new Scanner(System.in);
                 int a[]=new int[5];
                          System.out.println("Enter values in array");
                          for(int i=0;i<a.length;i++)
                          { a[i]=xyz.nextInt();
                          }
               int max=a[0];
    for(int i=0;i<a.length; i++)
    {
          if(a[i]>max) {
                                    max=a[i];
                                 }
                                 System.out.printf("%d\t",a[i]);
    }
               System.out.printf("\nMax vlaue is %d\n",max);
  }
```

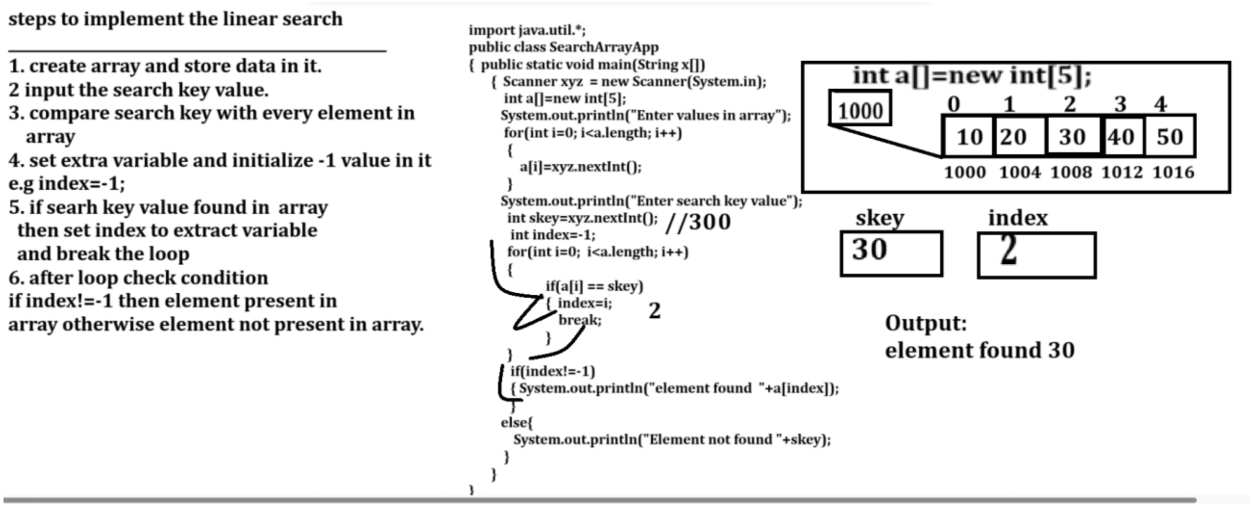
}

Example: WAP to input five values in array and input the one search key value and check search key present in array or not.

If you want to perform searching operation on array we have two approaches.

1. Linear Search
2. Binary Search

Linear Search: Linear Search operation means we compare search value or search key with every element in array and when element found we return its index otherwise return -1



steps to implement the linear search

1. create array and store data in it.
2 input the search key value.
3. compare search key with every element in array
4. set extra variable and initialize -1 value in it e.g index=-1;
5. if searh key value found in array then set index to extract variable and break the loop
6. after loop check condition if index!=-1 then element present in array otherwise element not present in array.

```
import java.util.*;
public class SearchArrayApp
{  public static void main(String x[])
     {  Scanner xyz  = new Scanner(System.in);
        int a[]=new int[5];
       System.out.println("Enter values in array");
        for(int i=0; i<a.length; i++)
        {
          a[i]=xyz.nextInt();
        }
       System.out.println("Enter search key value");
        int skey=xyz.nextInt();  //300
         int index=-1;
        for(int i=0;  i<a.length; i++)
        {
              if(a[i] == skey)
              { index=i;        2
                 break;
              }
        }
         if(index!=-1)
         { System.out.println("element found  "+a[index]);
         }
       else{
          System.out.println("Element not found "+skey);
        }
     }
}
```

Output:
element found 30

Example with source code

```
import java.util.*;
public class BinarySearchApp
{    public static void main(String x[])
           {  Scanner xyz  = new Scanner(System.in);
             int a[]=new int[5];
             System.out.println("Enter values in array");
             for(int i=0;i<a.length;i++)
             { a[i]=xyz.nextInt();
             }
             System.out.println("Enter search key");
             int skey=xyz.nextInt();
             int index=-1;
             for(int i=0;i<a.length;i++)
             {
                 if(a[i]==skey)
                              {  index=i;
                              }
                           System.out.printf("%d\t",a[i]);
             }
             if(index!=-1)
             { System.out.println("\nElement found "+a[index]);
             }
             else{
               System.out.println("\nElement not found");
             }
           }
}
```

Assignments

Q1. WAP to input five values in array and find min value from array

Q2. WAP to input five values in array and calculate sum of all elements

Q3. WAP to input 10 values in array and input search key and find the occurrence of search in key array

Example: 10 20 30 10 40 50 60 10 80 90

Input Search key: 10

Search key occur number of times: 3

Q4. WAP to input five element in array and reverse means swap value of index.

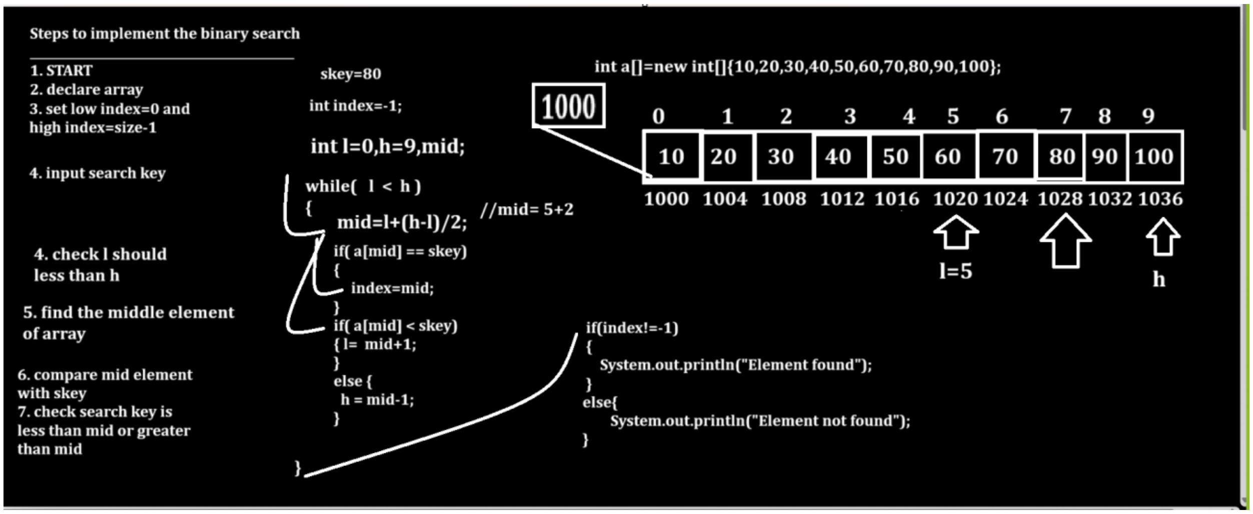
Now we want to discuss about the binary search

Q. What is binary search?

Binary search is a search algorithm used to find position of largest value within sorted.
it works by repeatedly dividing the large array in two parts and compare search key with middle value and if search key match with middle value then declare element found if search key is greater than middle then search value in right hand side and search key less than middle value then search value at left hand side.

Steps to implement the binary Search Algorithm

1. Divide the search space or array in two halves by find the middle index or mid
2. Compare the middle element of search space with key
3. If key is greater than middle element then search key in right hand side of array or section of array and if key is less than middle element the search key in left hand side of array
4. Repeat step 2 and 3 until array complete traverse or compare all element in array according to rules.
5. STOP.



Steps to implement the binary search

1. START
2. declare array
3. set low index=0 and high index=size-1

4. input search key

4. check l should less than h

5. find the middle element of array

6. compare mid element with skey
7. check search key is less than mid or greater than mid

```
skey=80

int index=-1;

int l=0,h=9,mid;

while(  l < h )
{
    mid=l+(h-l)/2;   //mid= 5+2
   if( a[mid] == skey)
   {
     index=mid;
   }
   if( a[mid] < skey)
   { l=  mid+1;
   }
   else {
     h = mid-1;
   }

}
```

```
int a[]=new int[]{10,20,30,40,50,60,70,80,90,100};
```

0	1	2	3	4	5	6	7	8	9
10	20	30	40	50	60	70	80	90	100
1000	1004	1008	1012	1016	1020	1024	1028	1032	1036

```
if(index!=-1)
{
   System.out.println("Element found");
}
else{
     System.out.println("Element not found");
}
```

Example with source code of binary search

```
import java.util.*;
public class BinarySearchApp
{
    public static void main(String x[])
            {
                        Scanner xyz  = new Scanner(System.in);
                        int a[]=new int[10];
                        int skey,l=0,h,mid,index=-1;
                        h=(a.length-1);
                        System.out.println("Enter values in array");
                        for(int i=0; i<a.length; i++)
                        {
                          a[i]=xyz.nextInt();
                        }
                        System.out.println("Enter search key value");
                        skey=xyz.nextInt(); //10
                        while(l<=h) //0<h
                        {
                          mid = l +(h-l)/2;
                          if(a[mid]==skey) //a[4]
                          { index=mid;
                              break;
                          }
                          if(a[mid]<skey)
                          {  l = mid+1;
                          }
                          else{
                            h = mid-1;
                          }
                        }
                        if(index!=-1)
                        { System.out.println("Element found "+skey);
                        }
                        else{
                          System.out.println("Element not found  "+skey);
                        }
            }
}
```

Example: Write program input five values in array and arrange in ascending order.

If we want to arrange element in ascending or in descending order we have some sorting technique provided by data structure

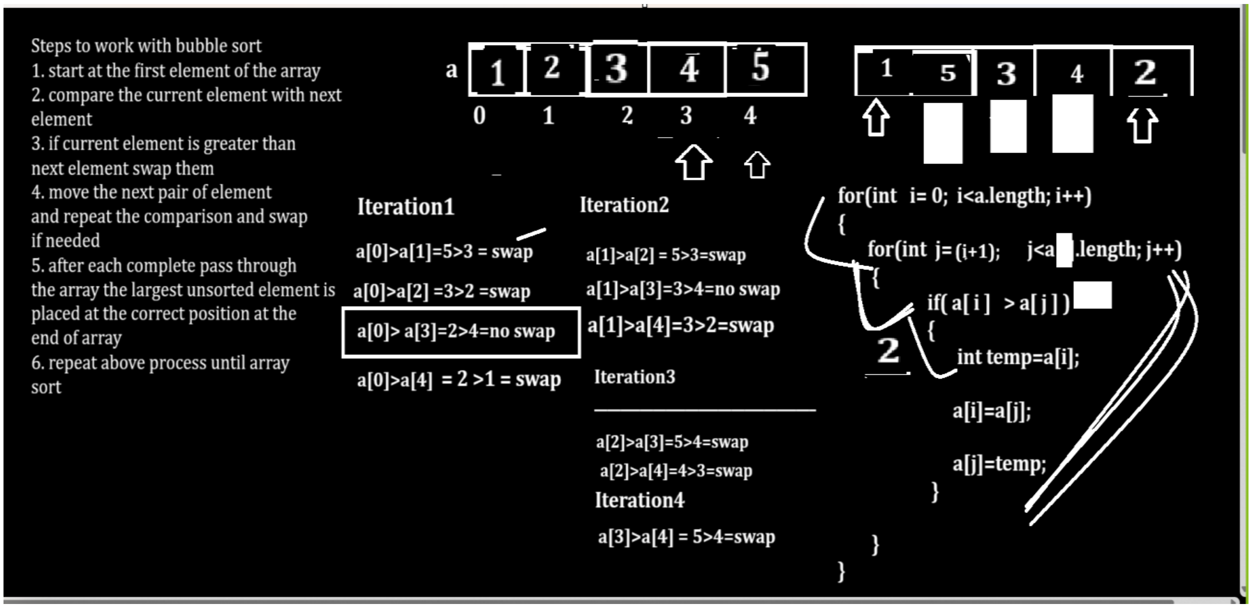
Types of sorting in data structure

1. Bubble Sort 2. Quick Sort 3. Insertion sort 4. Selection sort 5. Merge sort 6. Radix sort
7. Counting sort 8. Shell sort 9. Bitonic sort 10. Heap sort 11. Bucket sort 12. Comb sort etc

Now we want to discuss about the bubble sort

____

Bubble sort is algorithm which is used for sorting purpose and the working of bubble sort is compare the previous value with next value or previous element with next element and if previous element is greater than next element then perform swapping otherwise not and compare single element with all values in array.



Steps to work with bubble sort
1. start at the first element of the array
2. compare the current element with next element
3. if current element is greater than next element swap them
4. move the next pair of element and repeat the comparison and swap if needed
5. after each complete pass through the array the largest unsorted element is placed at the correct position at the end of array
6. repeat above process until array sort

Iteration1

a[0]>a[1]=5>3 = swap
a[0]>a[2] =3>2 =swap
a[0]> a[3]=2>4=no swap
a[0]>a[4] = 2 >1 = swap

Iteration2

a[1]>a[2] = 5>3=swap
a[1]>a[3]=3>4=no swap
a[1]>a[4]=3>2=swap

Iteration3

a[2]>a[3]=5>4=swap
a[2]>a[4]=4>3=swap

Iteration4

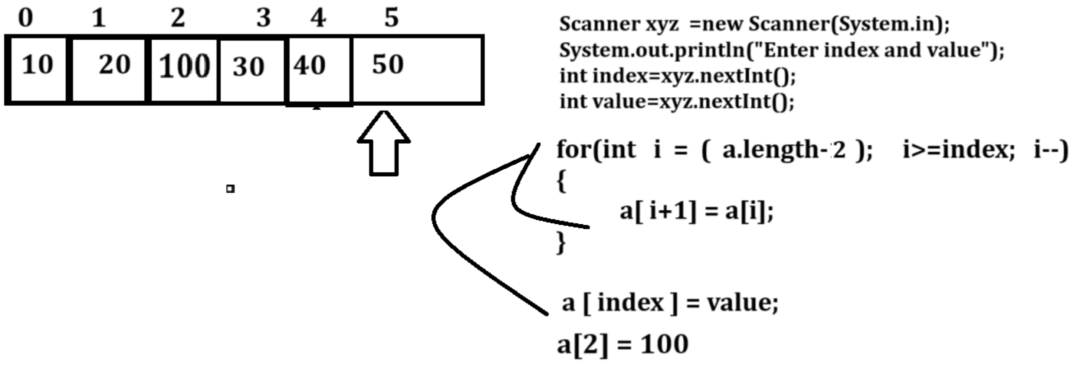
a[3]>a[4] = 5>4=swap

```
for(int  i= 0;  i<a.length; i++)
{
   for(int  j=(i+1);    j<a .length; j++)
   {
        if( a[ i ]  > a[ j ] )
        {
          int temp=a[i];

          a[i]=a[j];

          a[j]=temp;
        }

   }
}
```

Example with source code

```
import java.util.*;
public class BubbleSortApp
{
   public static void main(String x[])
   {
              Scanner xyz  = new Scanner(System.in);
              int a[]=new int[5];
              System.out.println("Enter five values in array");
              for(int i=0;i<a.length;i++)
              {
                a[i]=xyz.nextInt();
              }
              System.out.println("display array before sorting");
              for(int i=0; i<a.length; i++)
              {
                 System.out.printf("%d\t",a[i]);
              }
              //apply sorting using bubble sort technique.
              for(int i=0;i<a.length;i++)
              {
                for(int j=(i+1); j<a.length; j++)
                        {
                                   if(a[i]>a[j])
                                    {
                                  int temp=a[i];
                                               a[i]=a[j];
                                               a[j]=temp;
                                    }
                        }
              }
               System.out.println("\ndisplay array After sorting");
              for(int i=0; i<a.length; i++)
              {   System.out.printf("%d\t",a[i]);
              }
            }
}
```

Example: WAP to create array of size 6 and store 5 values in it and insert element in array on specified index.



```
Scanner xyz  =new Scanner(System.in);
System.out.println("Enter index and value");
int index=xyz.nextInt();
int value=xyz.nextInt();

for(int  i  =  (  a.length- 2  );    i>=index;   i--)
{
      a[ i+1] = a[i];
}

 a [ index ] = value;
a[2] = 100
```

Example with source code

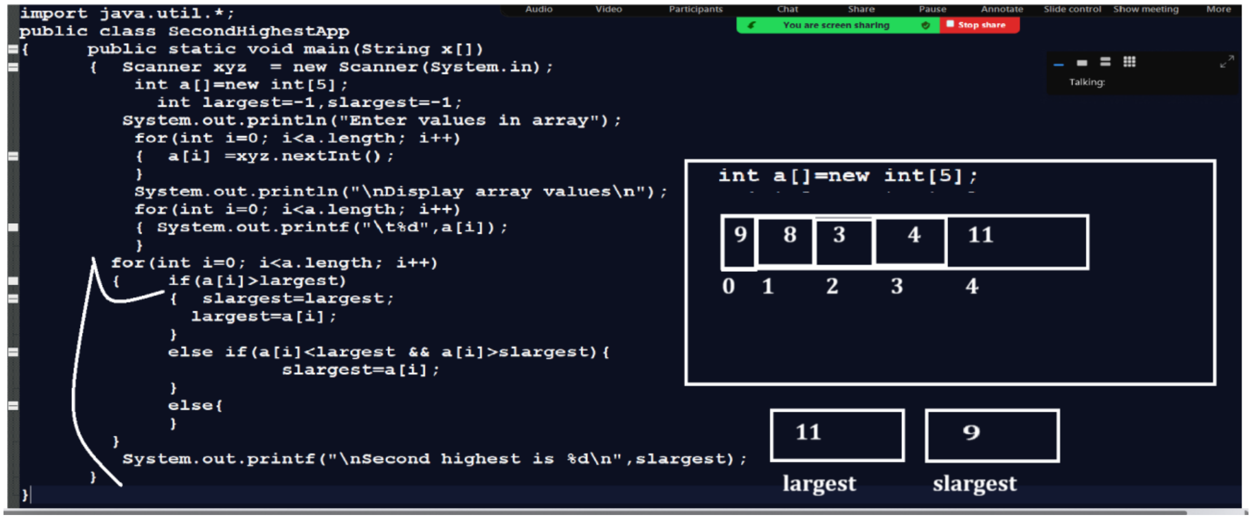
```
import java.util.*;
public class InsertEleApp
{
   public static void main(String x[])
           {
              Scanner xyz  = new Scanner(System.in);
                      int a[] = new int[6];
                      System.out.println("Enter the values in array");
                      for(int i=0; i<(a.length-1); i++)
                      {
                         a[i]=xyz.nextInt();
                      }
                      System.out.printf("\nBefore Inserting element\n");
                      for(int i=0;i<a.length;i++)
                      { System.out.printf("\na[%d] ---->%d\n",i,a[i]);
                      }
                      System.out.println("Enter index and value\n");
                       int index=xyz.nextInt();
                       int value=xyz.nextInt();

                       for(int i=(a.length-2); i>=index; i--)
                       {
                         a[i+1]=a[i];
                       }
                       a[index]=value;
                       System.out.printf("\nAfter Inserting element\n");
                      for(int i=0;i<a.length;i++)
                      { System.out.printf("\na[%d] ---->%d\n",i,a[i]);
```

```
                    }
          }
}
```

Example: WAP to find the second highest value from array.

```
import java.util.*;
public class SecondHighestApp
{
    public static void main(String x[])
           {  Scanner xyz  = new Scanner(System.in);
              int a[]=new int[5];
                     System.out.println("Enter values in array");
                     for(int i=0; i<a.length; i++)
                     {
                        a[i] =xyz.nextInt();
                     }
                     System.out.println("\nDisplay array values\n");
                     for(int i=0; i<a.length; i++)
                     { System.out.printf("\t%d",a[i]);
                     }
                     //write sorting logics
                     for(int i=0; i<a.length; i++)
                     {   for(int j=(i+1); j<a.length; j++)
                               {     if(a[i]>a[j])
                                                  {
                                                    int temp =a[i];
                                                           a[i]=a[j];
                                                           a[j]=temp;
                                                  }
                               }

                     }
                     System.out.println("\nDisplay array After sorting\n");
                     for(int i=0; i<a.length; i++)
                     { System.out.printf("\t%d",a[i]);
                     }
                     System.out.println("\nFind second highest value "+a[a.length-2]);

           }

}
```

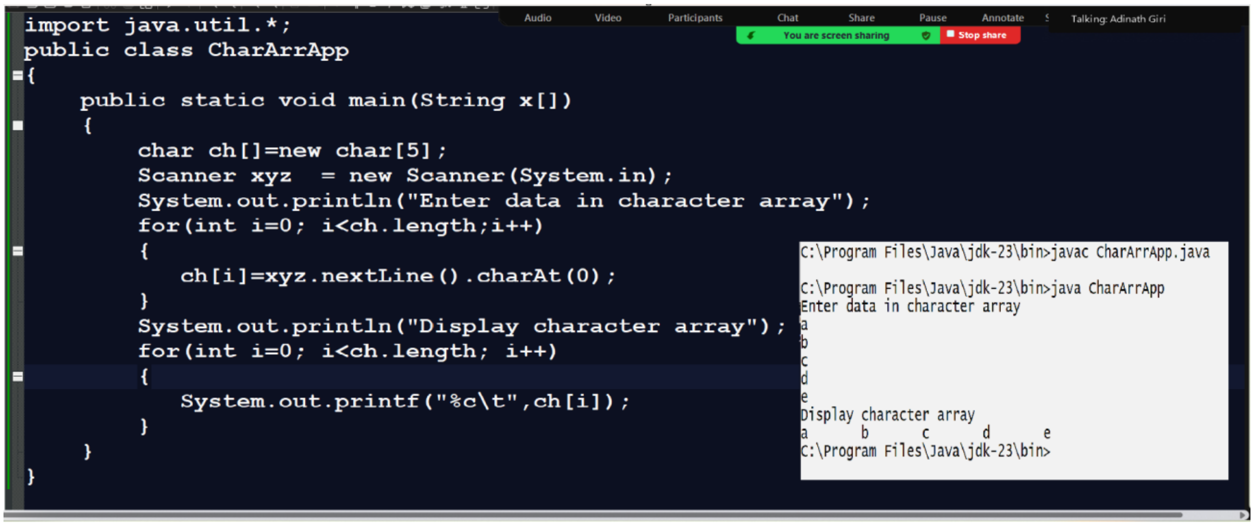
Example: Second highest logic without sorting



```
import java.util.*;
public class SecondHighestApp
{       public static void main(String x[])
        {  Scanner xyz  = new Scanner(System.in);
             int a[]=new int[5];
               int largest=-1,slargest=-1;
           System.out.println("Enter values in array");
            for(int i=0; i<a.length; i++)
            {  a[i] =xyz.nextInt();
            }
            System.out.println("\nDisplay array values\n");
            for(int i=0; i<a.length; i++)
            { System.out.printf("\t%d",a[i]);
            }
         for(int i=0; i<a.length; i++)
         {      if(a[i]>largest)
                {  slargest=largest;
                  largest=a[i];
                }
                else if(a[i]<largest && a[i]>slargest){
                          slargest=a[i];
                }
                else{
                }
         }
          System.out.printf("\nSecond highest is %d\n",slargest);
        }
}
```

Example: WAP to input 5 values in array and check array is anagram array or not

Example: WAP to create character array and store data in it and display it.



```
import java.util.*;
public class CharArrApp
{
    public static void main(String x[])
    {
        char ch[]=new char[5];
        Scanner xyz  = new Scanner(System.in);
        System.out.println("Enter data in character array");
        for(int i=0; i<ch.length;i++)
        {
            ch[i]=xyz.nextLine().charAt(0);
        }
        System.out.println("Display character array");
        for(int i=0; i<ch.length; i++)
        {
            System.out.printf("%c\t",ch[i]);
        }
    }
}
```

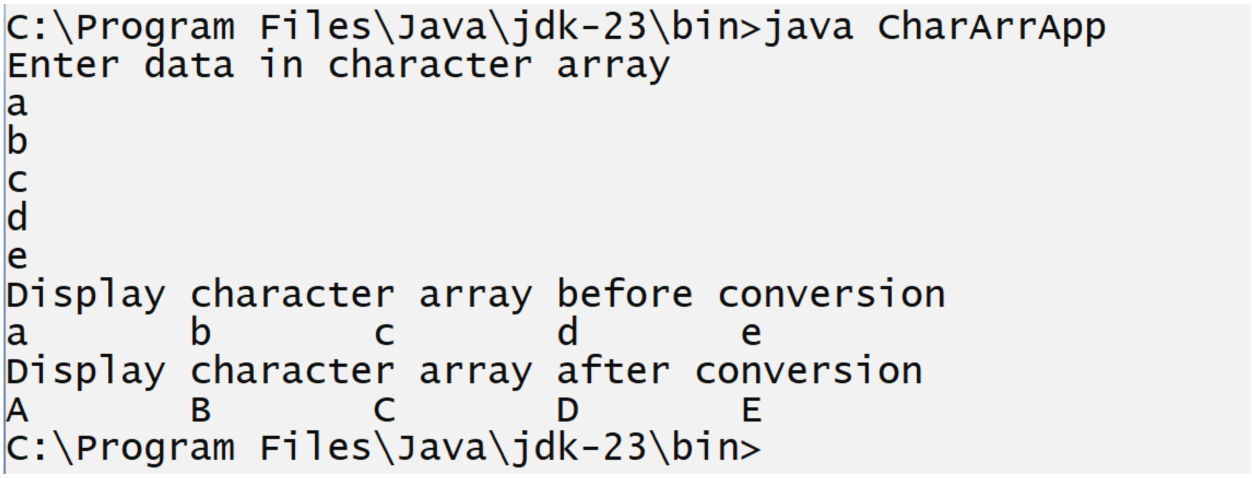
```
C:\Program Files\Java\jdk-23\bin>javac CharArrApp.java

C:\Program Files\Java\jdk-23\bin>java CharArrApp
Enter data in character array
a
b
c
d
e
Display character array
a       b       c       d       e
C:\Program Files\Java\jdk-23\bin>
```

Example : convert lower case character array in to upper case

```
import java.util.*;
public class CharArrApp
{
   public static void main(String x[])
          {
                        char ch[]=new char[5];
                        Scanner xyz  = new Scanner(System.in);
                        System.out.println("Enter data in character array");
```

```
                    for(int i=0; i<ch.length;i++)
                    {
                      ch[i]=xyz.nextLine().charAt(0);
                    }
                    System.out.println("Display character array before conversion");
                    for(int i=0; i<ch.length; i++)
                    {
                      System.out.printf("%c\t",ch[i]);
                    }
                    //apply conversion logics
                    for(int i=0; i<ch.length; i++)
                    {
                                if(ch[i]>=97 && ch[i]<=122)
                                {
                                            ch[i]=(char)((int)ch[i]-32);
                                }
                    }
                    System.out.println("\nDisplay character array after conversion");
                    for(int i=0; i<ch.length; i++)
                    {
                      System.out.printf("%c\t",ch[i]);
                    }
          }
}
```

Output

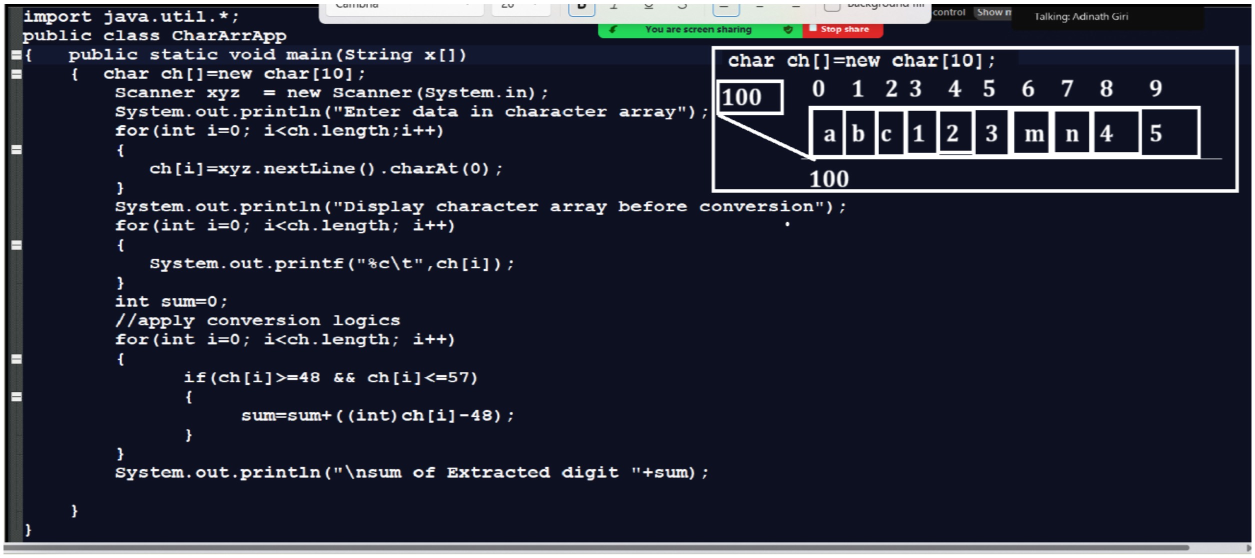


```
C:\Program Files\Java\jdk-23\bin>java CharArrApp
Enter data in character array
a
b
c
d
e
Display character array before conversion
a       b       c       d       e
Display character array after conversion
A       B       C       D       E
C:\Program Files\Java\jdk-23\bin>
```

Example: WAP to input character array of size 10 and extract digit from and calculate its sum

abc123mno45pqr

Output: 1+2+3+4+5=15



```
import java.util.*;
public class CharArrApp
{   public static void main(String x[])
    {  char ch[]=new char[10];
       Scanner xyz  = new Scanner(System.in);
       System.out.println("Enter data in character array");
       for(int i=0; i<ch.length;i++)
       {
          ch[i]=xyz.nextLine().charAt(0);
       }
       System.out.println("Display character array before conversion");
       for(int i=0; i<ch.length; i++)
       {
          System.out.printf("%c\t",ch[i]);
       }
       int sum=0;
       //apply conversion logics
       for(int i=0; i<ch.length; i++)
       {
             if(ch[i]>=48 && ch[i]<=57)
             {
                  sum=sum+((int)ch[i]-48);
             }
       }
       System.out.println("\nsum of Extracted digit "+sum);

    }
}
```

Example: WAP to create two character of size and compare character array without using inbuilt function

First character array: abcde
Second character array: abcmno
Output: arrays not equal

First character array: abcd
Second character array: abcd
Output: array is equal.

Example with source code

```
import java.util.*;
public class CharCompApp
{
   public static void main(String x[])
          {
             char first[]=new char[5];
                    char second[]=new char[5];
                    Scanner xyz  = new Scanner(System.in);
                    System.out.println("Enter first character array");
                    for(int i=0; i<first.length;i++)
                    {
                      first[i]=xyz.nextLine().charAt(0);
                    }
                    System.out.println("Enter second character array");
```

```
                for(int i=0; i<second.length;i++)
                {
                  second[i]=xyz.nextLine().charAt(0);
                }
                //apply comparision logics
                boolean flag=true;
                for(int i=0; i<first.length;i++)
                {
                   if(first[i]!=second[i])
                        { flag=false;
                          break;
                        }
                }
                if(flag)
                { System.out.println("Both Strings are same");
                }
                else
                { System.out.println("Both Strings are not same");
                }
        }
}
```

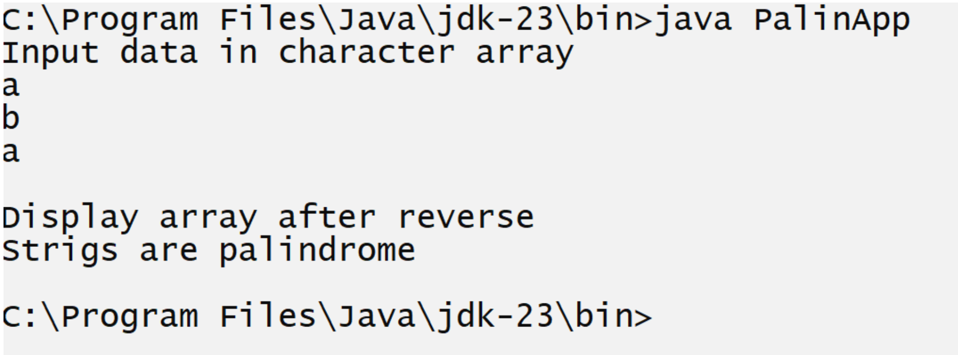
Example: WAP to input character array and check character array is palindrome or not
Input: c
Output: aba - palindrome string

Input :abc
Output: cba

Example with source code

```
import java.util.*;
public class PalinApp
{
  public static void main(String x[])
  {
    char first[]=new char[3];
          char temp[]=new char[3];
          Scanner xyz  = new Scanner(System.in);
          System.out.println("Input data in character array");
          for(int i=0; i<first.length;i++)
          {
            first[i]=xyz.nextLine().charAt(0);
          }
          //copy original array in temporary array
          for(int i=0;i<first.length;i++)
          {
            temp[i]=first[i];
          }
          //reverse the original array
          int mid=(first.length)/2;
          int end=(first.length-1);
          for(int start=0; start<mid; start++)
          {
            char t =first[start];
                  first[start]=first[end];
                  first[end]=t;
                  end--;
          }
          boolean flag=true;
          System.out.println("\nDisplay array after reverse");
          for(int i=0;i<first.length;i++)
          {   if(first[i]!=temp[i])
                   {   flag=false;
                                break;
                   }
          }
          if(flag)
          { System.out.println("Strigs are palindrome");
          }
          else
          { System.out.println("Strings are not palindrome");
          }
  }
}
```

Output



```
C:\Program Files\Java\jdk-23\bin>java PalinApp
Input data in character array
a
b
a

Display array after reverse
Strigs are palindrome

C:\Program Files\Java\jdk-23\bin>
```

Example with source code

```
import java.util.*;
public class PalinApp
{
  public static void main(String x[])
  {
     char first[]=new char[3];
              char temp[]=new char[3];
              Scanner xyz  = new Scanner(System.in);
              System.out.println("Input data in character array");
              for(int i=0; i<first.length;i++)
              {
                first[i]=xyz.nextLine().charAt(0);
              }
              //copy original array in temporary array
              for(int i=0;i<first.length;i++)
              {
                temp[i]=first[i];
              }
              //reverse the original array
              int mid=(first.length)/2;
              int end=(first.length-1);
              for(int start=0; start<mid; start++)
              {
                char t =first[start];
                        first[start]=first[end];
                        first[end]=t;
                        end--;
              }
              boolean flag=true;
              System.out.println("\nDisplay array after reverse");
              for(int i=0;i<first.length;i++)
              {
                 if(first[i]!=temp[i])
                        {
                                 flag=false;
                                        break;
                        }
              }
              if(flag)
              { System.out.println("Strigs are palindrome");
              }
```

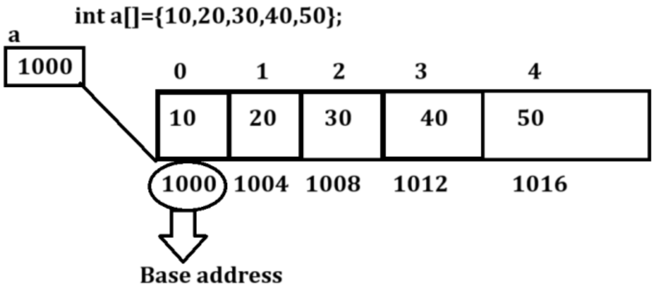
```
            else
            { System.out.println("Strings are not palindrome");
            }
    }
}
```

How to pass array as parameter to the function

If we want to pass array as parameter to the function we need to pass base address of array as parameter in function.

Q. What is base address?

0th location address memory address called as base address.



Q. How we get base address of array?

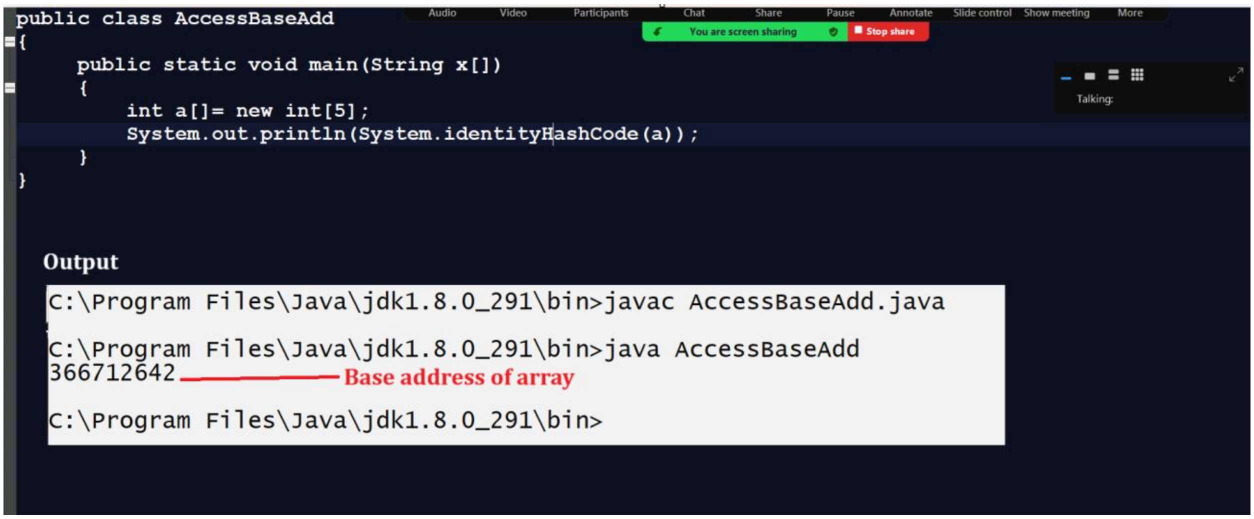
If we want to access the base address of array we can use array without subscript.

Syntax: arrayname; //base address

Example: int a[]=new int[5];

a ; // base address

Example with source code

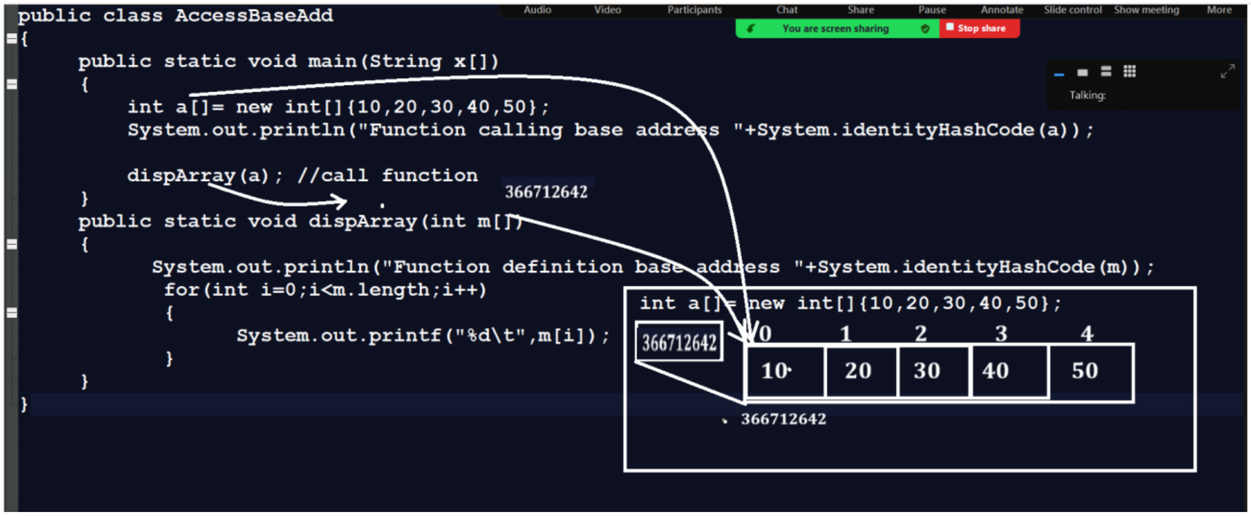


```
public class AccessBaseAdd
{
     public static void main(String x[])
     {
         int a[]= new int[5];
         System.out.println(System.identityHashCode(a));
     }
}
```

Output

```
C:\Program Files\Java\jdk1.8.0_291\bin>javac AccessBaseAdd.java

C:\Program Files\Java\jdk1.8.0_291\bin>java AccessBaseAdd
366712642

C:\Program Files\Java\jdk1.8.0_291\bin>
```

Example: WAP to create function name as void dispArra(int a[]) and this function can accept array as parameter and display it.



```
public class AccessBaseAdd
{
     public static void main(String x[])
     {
          int a[]= new int[]{10,20,30,40,50};
          System.out.println("Function calling base address "+System.identityHashCode(a));

          dispArray(a); //call function
     }
     public static void dispArray(int m[])
     {
            System.out.println("Function definition base address "+System.identityHashCode(m));
             for(int i=0;i<m.length;i++)
             {
                    System.out.printf("%d\t",m[i]);
             }
     }
}
```

Example with source code

public class AccessBaseAdd

{   public static void main(String x[])

{ int a[]= new int[]{10,20,30,40,50};

System.out.println("Function calling base address "+System.identityHashCode(a));

dispArray(a); //call function

}

public static void dispArray(int m[])

{   System.out.println("Function definition base address "+System.identityHashCode(m));

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

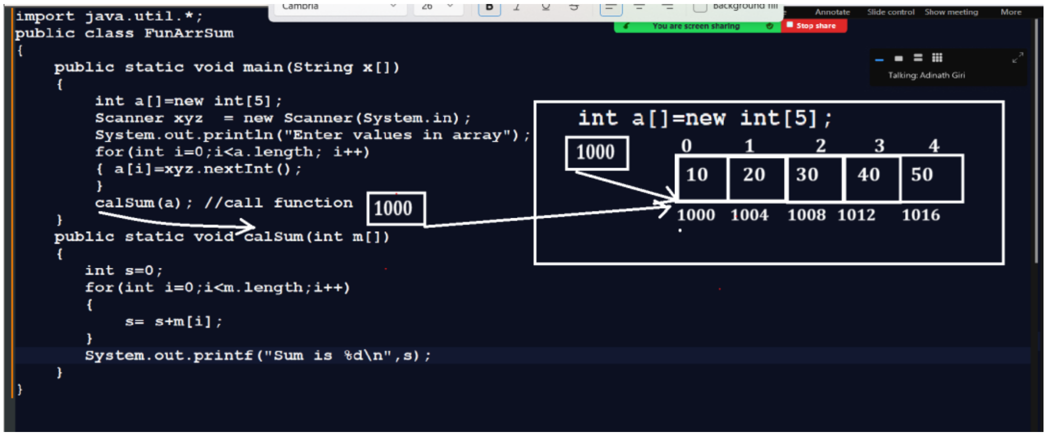
{ System.out.printf("%d\t",m[i]);

}

}

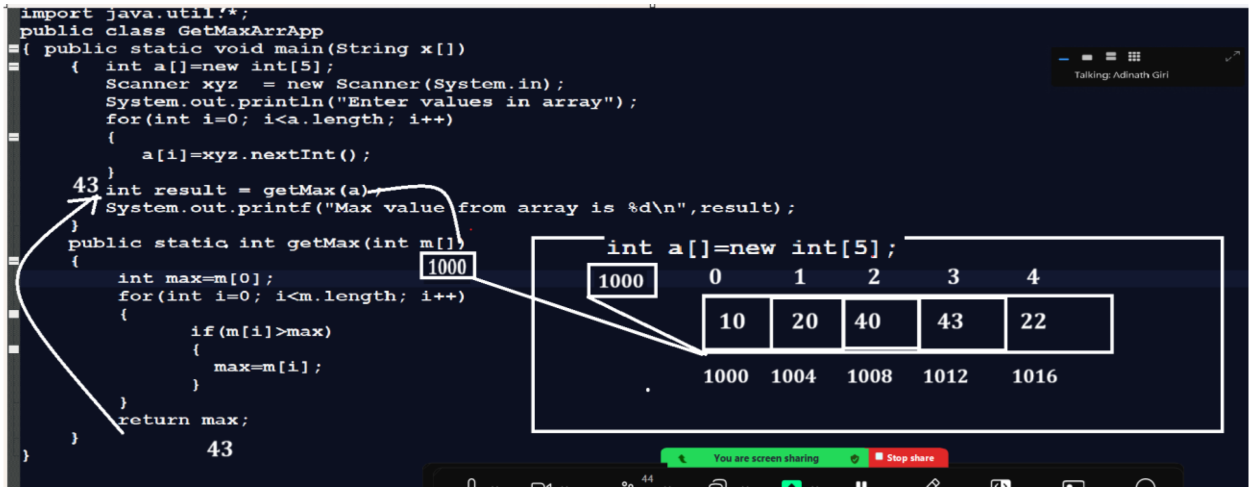
}

Example: WAP to create function name as void calSum(int a[]) we can pass array as parameter in calSum() function and calculate sum of all elements and display it.



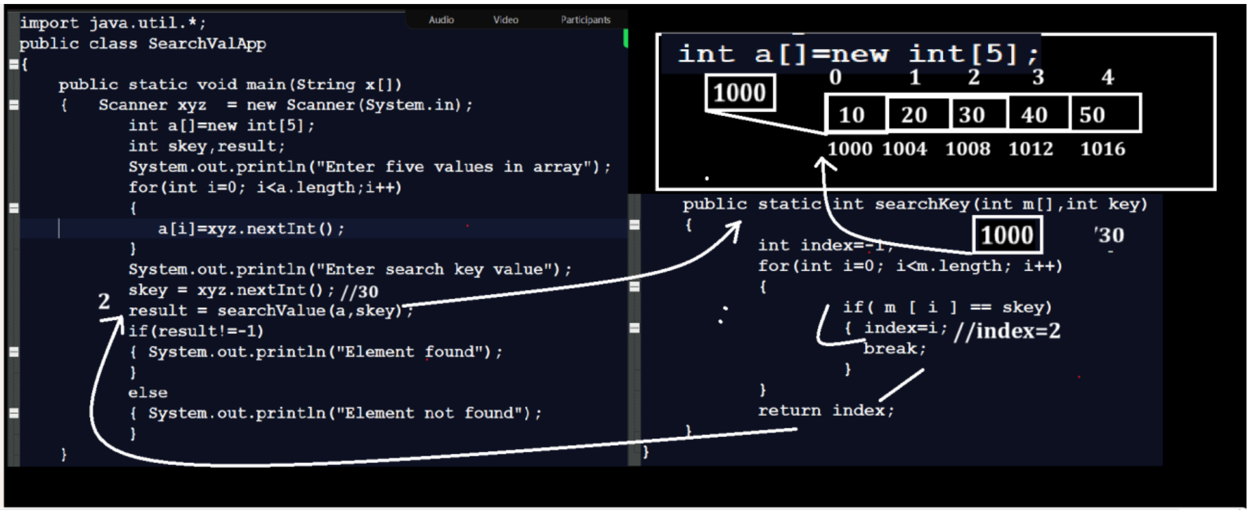
```
import java.util.*;
public class FunArrSum
{
    public static void main(String x[])
    {
        int a[]=new int[5];
        Scanner xyz  = new Scanner(System.in);
        System.out.println("Enter values in array");
        for(int i=0;i<a.length; i++)
        { a[i]=xyz.nextInt();
        }
        calSum(a); //call function
    }
    public static void calSum(int m[])
    {
       int s=0;
       for(int i=0;i<m.length;i++)
       {
           s= s+m[i];
       }
       System.out.printf("Sum is %d\n",s);
    }
}
```

Example: WAP to create function name as int getMax(int a[]) this function can accept array as parameter and find the max value and return it.



```
import java.util.*;
public class GetMaxArrApp
{ public static void main(String x[])
    {  int a[]=new int[5];
       Scanner xyz  = new Scanner(System.in);
       System.out.println("Enter values in array");
       for(int i=0; i<a.length; i++)
       {
          a[i]=xyz.nextInt();
       }
       int result = getMax(a);
       System.out.printf("Max value from array is %d\n",result);
    }
    public static int getMax(int m[])
    {
        int max=m[0];
        for(int i=0; i<m.length; i++)
        {
              if(m[i]>max)
              {
                max=m[i];
              }
        }
        return max;
    }
}
```

Example: WAP to create function name as int searchValue(int a[],int key): this function can accept array and search key as parameter and search element in array and when element found in array return its index and otherwise return -1.



```
import java.util.*;
public class SearchValApp
{
    public static void main(String x[])
    {   Scanner xyz  = new Scanner(System.in);
           int a[]=new int[5];
           int skey,result;
           System.out.println("Enter five values in array");
           for(int i=0; i<a.length;i++)
           {
              a[i]=xyz.nextInt();
           }
           System.out.println("Enter search key value");
           skey = xyz.nextInt(); //30
           result = searchValue(a,skey);
           if(result!=-1)
           { System.out.println("Element found");
           }
           else
           { System.out.println("Element not found");
           }
    }
}

public static int searchKey(int m[],int key)
{
        int index=-1;
        for(int i=0; i<m.length; i++)
        {
              if( m [ i ] == skey)
              { index=i; //index=2
                break;
              }
        }
        return index;
}
}
```

Example with source code

import java.util.*;

public class SearchValApp

{

public static void main(String x[])

{ Scanner xyz = new Scanner(System.in);

int a[]=new int[5];

int skey,result;

```
         System.out.println("Enter five values in array");
         for(int i=0; i<a.length;i++)
         {
          a[i]=xyz.nextInt();
         }
         System.out.println("Enter search key value");
         skey = xyz.nextInt();
         result = searchValue(a,skey);
         if(result!=-1)
         { System.out.println("Element found");
         }
         else
         { System.out.println("Element not found");
         }
    }
    public static int searchValue(int m[],int key)
    {
      int index=-1;
         for(int i=0; i<m.length; i++)
         {
           if( m [ i ] == key)
                 { index=i;
                  break;
                 }
         }
         return index;
    }
}
```