Session 05:

JAVA Arrays and Strings

# Describe an array

# Explain declaration, initialization, and instantiation of a single-dimensional array

# Explain declaration, initialization, and instantiation of a multi-dimensional array

# Explain the use of loops to process an array

# Describe an array

Arrays are used to store multiple values in a single variable, instead of declaring separate variables for each value.

To declare an array, define the variable type with **square brackets**:

String[] cars;

We have now declared a variable that holds an array of strings. To insert values to it, we can use an array literal - place the values in a comma-separated list, inside curly braces:

String[] cars = {"Volvo", "BMW", "Ford", "Mazda"};

To create an array of integers, you could write:

int[] myNum = {10, 20, 30, 40};

# Explain declaration, initialization, and instantiation of a single-dimensional array

## Access the Elements of an Array

class Array1{

public static void main(String args[]){

int a[]=new int[5];//phan khai bao va khoi tao

a[0]=10;//Phan khoi tao

a[1]=20;

a[2]=70;

a[3]=40;

a[4]=50;

//in mang

for(int i=0;i<a.length;i++)//length la thuoc tinh cua mang

System.out.println(a[i]);

}}

class Testarray1{

public static void main(String args[]){

int a[]={33,3,4,5};//khai bao, khoi tao va khoi tao

//in mang

for(int i=0;i<a.length;i++)//length la thuoc tinh cua mang

System.out.println(a[i]);

}}

class Testarray2{

static void min(int arr[]){

int min=arr[0];

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

if(min>arr[i])

min=arr[i];

System.out.println(min);

}

public static void main(String args[]){

int a[]={33,3,4,5};

min(a);//Truyen mang toi phuong thuc

}}

## Loop Through an Array

public class TestArray3 {

public static void main(String[] args) {

double[] BK49 = {1.9, 2.9, 3.4, 3.5};

// In tat ca cac phan tu mang

for (double element: BK49) {

System.out.println(element);

}

}

}

## Loop Through an Array with For-Each

package edu.doannhg.basic;

public class ArrayDemo {

public static void hamDaoNguoc(int[] list) {

int[] result = new int[list.length];

// vong lap de dao nguoc mang

for (int i = 0, j = result.length - 1; i < list.length; i++, j--) {

result[j] = list[i];

}

// vong lap foreach de hien thi cac phan tu trong mang dao nguoc

for (int a: result) {

System.out.print(a+ " ");

}

}

public static void main(String [] args){

int [] arr = {9, 8, 7, 6, 5, 4, 3, 2, 1};

hamDaoNguoc(arr);

}

}

# Explain declaration, initialization, and instantiation of a multi-dimensional array

int[][] arr=new int[3][3];//3 hang va 3 cot

class Testarray3{

public static void main(String args[]){

//khai bao va khoi tao mang 2 chieu

int arr[][]={{1,2,3},{2,4,5},{4,4,5}};

//in mang hai chieu

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

for(int j=0;j<3;j++){

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

}

System.out.println();

}

}}

# Explain the use of loops to process an array

### Sort

public static void main(String[] args) {

    int n, temp, max = 100;

    Scanner scanner = new Scanner(System.in);

    // khai báo và cấp phát bộ nhớ cho mảng A

    int A[] = new int[max];

    // nhập số phần tử của mảng

    // kiểm tra nếu n <= 2 hoặc n > max - 1

    // thì phải nhập lại

    do {

        System.out.println("Nhập số phần tử của mảng: ");

        n = scanner.nextInt();

    } while (n <= 2 || n > max-1);

    System.out.println("Nhập giá trị cho các phần tử của mảng: ");

    for (int i = 0; i < n; i++) {

        System.out.print("A[" + i + "] = ");

        A[i] = scanner.nextInt();

    }

    // in ra màn hình mảng số ban đầu

    System.out.println("Mảng ban đầu: ");

    for (int i = 0; i < n; i++) {

        // in các phần tử trên cùng 1 dòng

        System.out.print(A[i] + "\t");

    }

    // sắp xếp các phần tử trong mảng

    // vòng lặp for sẽ duyệt i và j

    // i chạy từ 0 đến n - 2, j chay từ i + 1 đến n - 1

    // nếu phần tử tại chỉ số j nhỏ hơn phần tử tại i

    // thì sẽ hoán đổi vị trí 2 phần tử này cho nhau

    for (int i = 0; i < n - 1; i++) {

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

            if (A[j] < A[i]) {

                temp = A[i];

                A[i] = A[j];

                A[j] = temp;

            }

        }

    }

    // in ra màn hình mảng số sau khi sắp xếp

    System.out.println("\nMảng sau khi sắp xếp: ");

    for (int i = 0; i < n; i++) {

        System.out.print(A[i] + "\t");

    }

}

### Search

public static void main(String[] args) {

    int n, max = 100, i;

    int[] A = new int[max];

    Scanner scanner = new Scanner(System.in);

    do {

        System.out.println("Nhập số phần tử của mảng: ");

        n = scanner.nextInt();

    } while (n <= 0 || n > max - 1);

    System.out.println("Nhập giá trị cho các phần tử của mảng: ");

    for (i = 0; i < n; i++) {

        System.out.print("A[" + i + "] = ");

        A[i] = scanner.nextInt();

    }

    System.out.println("Nhập số nguyên k cần tìm: ");

    int k = scanner.nextInt();

    i = 0;

    // tiến hành tìm kiếm phần tử

    // nếu không tìm thấy giá trị tại i bằng với số nguyên k

    // thì tăng i lên 1

    while ((i < n) && (A[i] != k)) {

        i++;

    }

    // Nếu i lớn hơn số phần tử của mảng - 1

    // thì thông báo không tìm thấy phần tử

    // ngược lại hiển thị ra i

    if (i > n-1) {

        System.out.println("Trong mảng không có phần tử nào chứa giá trị cần tìm.");

    } else {

        System.out.println("Phần tử đầu tiên có giá trị bằng " + k + " tại vị trí = " + i);

    }

}

Bài Tập

1. Java - Tìm và in ra màn hình số lượng số nguyên lẻ, số lượng số nguyên chẵn trong mảng

public static void main(String[] args) {

    int n;

    int soLuongSoNguyenLe = 0, soLuongSoNguyenChan = 0;

    Scanner scanner = new Scanner(System.in);

    do {

        System.out.println("Nhập vào số phần tử của mảng: ");

        n = scanner.nextInt();

    } while (n <= 0);

    int A[] = new int[n];

    System.out.println("Nhập các phần tử cho mảng: ");

    // A.length: trả về độ dài của mảng

    for (int i = 0; i < n; i++) {

        System.out.print("Nhập phần tử thứ " + i + ": ");

        A[i] = scanner.nextInt();

    }

    // tính số lượng số nguyên lẻ, chẵn trong mảng

    for (int i = 0; i < n; i++) {

        if (A[i] % 2 == 0) {

            soLuongSoNguyenChan++;

        } else {

            soLuongSoNguyenLe++;

        }

    }

    System.out.println("Số lượng số nguyên chẵn trong mảng = " + soLuongSoNguyenChan);

    System.out.println("Số lượng số nguyên lẻ trong mảng = " + soLuongSoNguyenLe);

}

1. Viết chương trình thực hiện các công việc sau: Nhập một mảng có n phần tử nguyên từ bàn phím *(n > 0*). Sau đó nhập vào số nguyên k và xóa các phần tử trong mảng có giá trị bằng k.

### **Yêu cầu kỹ thuật:** Chương trình phải kiểm tra n nhập vào: nếu n <= 0 thì cho nhập lại số phần tử cho đến khi thỏa mãn điều kiện.

public static void main(String[] args) {

    int n, i, c;

    Scanner scanner = new Scanner(System.in);

    do {

        System.out.println("Nhập vào số phần tử của mảng: ");

        n = scanner.nextInt();

    } while (n <= 0);

    int A[] = new int[n];

    System.out.println("Nhập các phần tử cho mảng: ");

    for (i = 0; i < n; i++) {

        System.out.print("Nhập phần tử thứ " + i + ": ");

        A[i] = scanner.nextInt();

    }

    System.out.println("Nhập số nguyên k: ");

    int k = scanner.nextInt();

    // xóa phần tử k ra khỏi mảng

    // vòng lặp for sẽ khởi tạo c = i = 0

    // và duyệt i từ 0 đến n

    // nếu phần tử tại vị trí i khác với số nguyên k

    // thì gán phần tử tại i cho phần tử tại k

    // sau đó tăng c lên 1

    for (c = i = 0; i < n; i++) {

        if (A[i] != k) {

            A[c] = A[i];

            c++;

        }

    }

    n = c;  // lúc này số phần tử trong mảng sẽ bằng c

    // hiển thị các phần tử trong mảng sau khi xóa k

    System.out.println("Mảng còn lại sau khi xóa phần tử " + k + " là: ");

    for (i = 0; i < n; i++) {

        System.out.print(A[i] + "\t");

    }

}

1. Viết chương trình nhập vào một mảng số nguyên và sau đó tách các số chẵn và lẻ trong mảng đó sang hai mảng khác.

public static void main(String[] args) {

    int n, c, d;

    Scanner scanner = new Scanner(System.in);

    do {

        System.out.println("Nhập vào số phần tử của mảng: ");

        n = scanner.nextInt();

    } while (n <= 0);

    int A[] = new int[n];

    int ch[] = new int[n];  // mảng chứa các phần tử là số chẵn

    int le[] = new int[n];  // mảng chứa các phần tử là số lẻ

    System.out.println("Nhập các phần tử cho mảng: ");

    for (int i = 0; i < n; i++) {

        System.out.print("Nhập phần tử thứ " + i + ": ");

        A[i] = scanner.nextInt();

    }

    // c: số phần tử của mảng ch

    // d: số phần tử của mảng le.

    c = d = 0;

    for (int i = 0; i < n; i++) {

        // nếu phần tử tại vị trí i chia hết cho 2

        // thì gán phần tử đó cho phần tử tại vị trí c của mảng ch

        // ngược lại thì gán phần tử đó

        // cho phần tử tại vị trí d của mảng le

        if (A[i] % 2 == 0) {

            ch[c] = A[i];

            c++;

        } else {

            le[d] = A[i];

            d++;

        }

    }

    System.out.println("Các phần tử của mảng ch là: ");

    for (int i = 0; i < c; i++) {

        System.out.print(ch[i] + "\t");

    }

    System.out.println("\nCác phần tử của mảng le là: ");

    for (int i = 0; i < d; i++) {

        System.out.print(le[i] + "\t");

    }

}

1. Viết chương trình nhập vào một mảng các số nguyên bất kỳ và sau đó hiển thị các phần tử chia hết cho 5 trong mảng đó.

public static void main(String[] args) {

    int n;

    Scanner scanner = new Scanner(System.in);

    do {

        System.out.println("Nhập vào số phần tử của mảng: ");

        n = scanner.nextInt();

    } while (n <= 0);

    int A[] = new int[n];

    System.out.println("Nhập các phần tử cho mảng: ");

    for (int i = 0; i < n; i++) {

        System.out.print("Nhập phần tử thứ " + i + ": ");

        A[i] = scanner.nextInt();

    }

    // tìm và hiển thị các phần tử trong mảng chia hết cho 5

    System.out.println("Các phần tử chia hết cho 5 là: ");

    for (int i = 0; i < n; i++) {

        if (A[i] % 5 == 0) {

            System.out.print(A[i] + "\t");

        }

    }

}

1. Viết chương trình đếm số lần xuất hiện của 1 phần tử bất kỳ được nhập từ bàn phím

public static void main(String[] args) {

    int n;

    // biến đếm số phần tử được nhập từ bàn phím có trong mảng

    int count = 0;

    Scanner scanner = new Scanner(System.in);

    do {

        System.out.println("Nhập vào số phần tử của mảng: ");

        n = scanner.nextInt();

    } while (n < 0);

    int A[] = new int[n];

    System.out.println("Nhập các phần tử cho mảng: ");

    for (int i = 0; i < n; i++) {

        System.out.print("Nhập phần tử thứ " + i + ": ");

        A[i] = scanner.nextInt();

    }

    // Đếm số lần xuất hiện của 1 phần tử được nhập từ bàn phím

    System.out.println("Nhập vào 1 số nguyên bất kỳ: ");

    int number = scanner.nextInt();

    for (int i = 0; i < n; i++) {

        if (A[i] == number) {

            count++;

        }

    }

    System.out.println("Số phần tử " + number + " có trong mảng = " + count);

}

1. Viết chương trình nhập vào một mảng số nguyên có n phần tử và thực hiện các công việc sau:

* Xuất giá trị các phần tử của mảng.
* Tìm phần tử có giá trị lớn nhất, nhỏ nhất.
* Đếm số phần tử là số chẵn.
* Sắp xếp mảng tăng dần.

public static void main(String[] args) {

    int n;

    Scanner scanner = new Scanner(System.in);

    do {

        System.out.println("Nhập vào số phần tử của mảng: ");

        n = scanner.nextInt();

    } while (n < 0);

    int A[] = new int[n];

    System.out.println("Nhập các phần tử cho mảng: ");

    for (int i = 0; i < n; i++) {

        System.out.print("Nhập phần tử thứ " + i + ": ");

        A[i] = scanner.nextInt();

    }

    // Xuất giá trị các phần tử của mảng

    System.out.println("\nMảng ban đầu: ");

    for (int i = 0; i < n; i++) {

        System.out.print(A[i] + "\t");

    }

    // Tìm phần tử có giá trị lớn nhất, nhỏ nhất

    int max = A[0]; // khởi tạo phần tử lớn nhất là phần tử đầu tiên

    int min = A[0]; // khởi tạo phần tử bé nhất là phần tử đầu tiên

    for (int i = 0; i < n; i++) {

        if (A[i] < min) {

            min = A[i];

        }

        if (A[i] > max) {

            max = A[i];

        }

    }

    System.out.print("\nPhần tử lớn nhất trong mảng là " + max);

    System.out.print("\nPhần tử nhỏ nhất trong mảng là " + min);

    // Đếm số phần tử là số chẵn

    int soPhanTuChan = 0;

    for (int i = 0; i < n; i++) {

        if (A[i] % 2 == 0) {

            soPhanTuChan++;

        }

    }

    System.out.println("\nSố phần tử chẵn có trong mảng = " + soPhanTuChan);

    // Sắp xếp măng tăng dần

    // Java cung cấp cho chúng ta một thư viện có sẵn

    // để sắp xếp mảng, đó là thư viện sort của lớp Arrays

    // mặc định thư viện này sẽ sắp xếp mảng tăng dần

    Arrays.sort(A);

    System.out.println("\nMảng sau khi sắp xếp: ");

    for (int i = 0; i < n; i++) {

        System.out.print(A[i] + "\t");

    }

}

1. Viết chương trình nhập  các phần tử cho mảng sao cho không có phần tử nào trùng nhau.

public static void main(String[] args) {

    int n, i = 0;

    Scanner scanner = new Scanner(System.in);

    do {

        System.out.println("Nhập vào số phần tử của mảng: ");

        n = scanner.nextInt();

    } while (n < 0);

    int A[] = new int[n];

    System.out.println("Nhập các phần tử cho mảng: ");

    do {

        int flag=0;

        System.out.print("Nhập phần tử thứ " + i + ": ");

        A[i] = scanner.nextInt();

        for(int j = i-1; j >= 0; j--) {

            if (A[i] == A[j]) {

                // nếu phần tử đứng sau bằng phần tử đứng trước

                // thì flag = 1

                // thì thoát khỏi vòng lặp for

                System.out.println("Phần tử đã tồn tại. Mới nhập lại!");

                flag=1;

                break;

            }

        }

        if(flag == 1) {

            // sau khi ra khỏi vòng lặp for

            // mà flag = 1

            // thì quay lại thực hiện thân vòng lặp do - while

            continue;

        } else {

            // ngược lại tăng i lên 1

            // và quay lại thực hiện thân vòng lặp do - while

            i++;

        }

     } while (i<n);

    // Xuất giá trị các phần tử của mảng

    System.out.println("Mảng vừa nhập: ");

    for (i = 0; i < n; i++) {

        System.out.print(A[i] + "\t");

    }

}