



មជ្ឈមណ្ឌលកូរ៉េ សហវ័រ អេច អ ឌី

Korea Software HRD Center

កម្មវិធីបង្រៀន Java Programming ជាភាសាខ្មែរ

Online Java Training Course

Advisor: Dr. Kim Tae Kyung



www.kshrd.com.kh

ជំពូកទី៥

Arrays

មេរៀនទី១: Array 2D និង 3D

មេរៀនទី២: Fixed Size Array

មេរៀនទី៣: Dynamic Array

មេរៀនទី១៖ **Array 2D និង 3D**

១.១. អ្វីជា Array?

១.២. ប្រភេទនៃ Array

១.៣. ការប្រកាស Array

១.៤. Array ក្នុង foreach Loops

១.៥. Passing Arrays to Methods

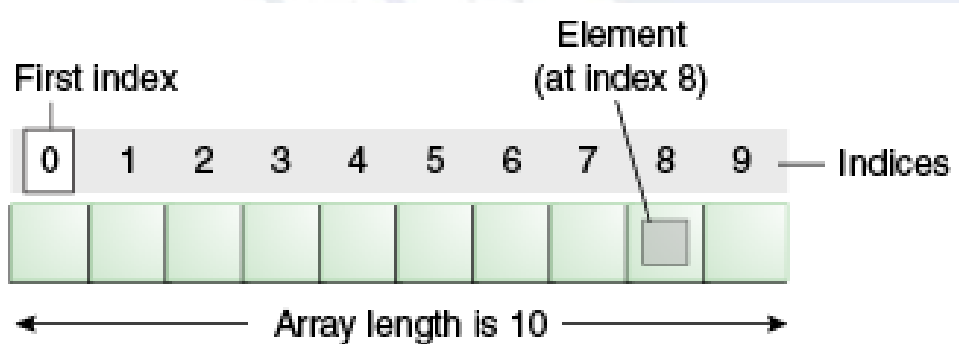
១.៦. Returning an Array From a Method

១.៧. The Arrays Class

១.១. អ្វីជា Array?

□ **Array** គឺជា **Container Object** ដែលវា **Store** ឬ ក្លោប **Fixed-Size Collection** នៃ **Element** ទៅកាន់ប្រភេទដូចគ្នា ឬ យើងអាចនិយាយថា វាជា សំណុំអញ្ញាត ដែលមាន ប្រភេទទិន្នន័យដូចគ្នា ក្រោមឈ្មោះរួមមួយ ដោយខុសគ្នា រវាងលេខ **Index** ។ របៀបប្រកាស **Data Type** ជា **Array** នៅក្នុង **Java** ។

Software World



Real World



១.២. ប្រភេទនៃ **Array**

□ **Java Array** មានពីរប្រភេទ៖

☞ **Single Dimensional (1D) Array:**

ឧទាហរណ៍:

```
int a[] = new int[5]; // 1 Dimensional
```

☞ **Multidimensional Array (2D, 3D,...):**

ឧទាហរណ៍:

```
int arr[][]; // 2 Dimensional
```

```
int[][][] array3D = new int[x][y][z]; // 3 Dimensional
```

១.២. ប្រភេទនៃ **Array**

ឧទាហរណ៍ នៃ 1 Dimensional Array:

```
package array;
public class SingleDmArray {
    public static void main(String args[]){
        int a[]=new int[5];//declaration, instantiation
        a[0]=10;    //initialization
        a[1]=20;
        a[2]=30;
        a[3]=70;
        a[4]=50;
        for(int i=0;i<a.length;i++)
            System.out.println(a[i]);
    }
}
```

Result:

10
20
30
70
50

១.៣. ការប្រកាស **Array**

- យើងអាច ប្រកាស **Array** តាមរបៀបដូចខាងក្រោម៖

រូបមន្ត៖

```
dataType[ ] arrayRefVar;
```

```
dataType [ ][ ]arrayRefVar;
```

```
dataType arrayRefVar[ ][ ][ ];
```

```
dataType [ ]arrayRefVar[ ];
```

```
dataType arrayRefVar[ ] = {33,3,4,5};
```

១.៤. Array ក្នុង foreach Loops

- ក្នុង **JDK 1.5** បានណែនាំយើងអោយស្គាល់នូវ **Loop** ថ្មីមួយដែលត្រូវបានហៅថា **foreach Loop** ដែលអាចអោយ យើងធ្វើការបង្ហាញតម្លៃរបស់ **Array** ជាបន្តបន្ទាប់គ្នាដោយមិនចាំបាច់ប្រើ **Index Variable** ។ ឧទាហរណ៍ខាងក្រោមនេះ បានប្រើ **Method** ឈ្មោះថា **printArray** ដើម្បីបង្ហាញធាតុរបស់ **array int** មួយ ។

ឧទាហរណ៍ :

```
package array;
public class DisplayArr {
    public static void main(String args[]){
        double[] myList = {1.9, 2.9, 3.4, 3.5};

        //print all the array elements
        for(double element: myList){
            System.out.println(element);
        }
    }
}
```

Result:

1.9
2.9
3.4
3.5

១.៥. Passing Arrays to Methods

- អ្នកមិនត្រឹមតែ អាចបោះពុម្ពជាប្រភេទ **Primitive Type** ទៅអោយ **Method** ប៉ុណ្ណោះទេ តែអ្នក ក៏អាចបោះពុម្ពជាប្រភេទ **Array** បានផងដែរ ។

ឧទាហរណ៍ខាងក្រោមនេះ បានប្រើ **Method** ឈ្មោះថា **printArray** ដើម្បីបង្ហាញ
លេខ រាប់ **array int** មួយ។

```
public class DisplayArr {  
    public static void printArray(int[] array){  
        for (int i = 0; i < array.length; i++){  
            System.out.print(array[i] + ":");  
        }  
    }  
    public static void main(String args[]){  
        printArray(new int[]{3, 1, 2, 6,  
    }  
}
```

លទ្ធផល

```
3 : 1 : 2 : 6 : 4  
: 2 :
```

១.៦. Returning an Array From a Method

- **Method** ក៏អាចធ្វើការ **Return** ជា **Array** ដែរ ។ ឧទាហរណ៍ **Method** ខាងក្រោមនឹងធ្វើការ **Return Array** មួយដែលវា ផ្ទុយ (Reverse) ជាមួយ **Array** មួយទៀត ។

ឧទាហរណ៍:

```
public static int[] reverse(int[] list) {  
    int[] result = new int[list.length];  
    for (int i = 0, j = result.length - 1;  
         i < list.length; i++, j--) {  
        result[j] = list[i];  
    }  
    return result; //return array  
}
```

១.៧. The Arrays Class

- **java.util.Arrays class** មាន **Static Method** ផ្សេងៗគ្នាជាច្រើន សម្រាប់ធ្វើការ តម្រៀប (**Sorting**) ស្វែងរក (**Searching Array**), ប្រៀបធៀប (**Comparing Array**) ហើយនឹងបញ្ចូលធាតុ **Array (filling array element)**។

SN	Methods with Description
1	public static int binarySearch(Object[] a, Object key) : ស្វែងរក Array ជាក់លាក់ នៃ Object (Byte, Int , double, etc.) សម្រាប់តម្លៃជាក់លាក់ ដោយការប្រើប្រាស់ Binary Search Algorithm ។
2	public static boolean equals(long[] a, long[] a2) : Returns ពិតកាលណា Array of Longs មាន តម្លៃស្មើទៅនឹង Array ផ្សេងទៀត ។ Array ទាំងពីរនឹងស្មើគ្នាកាលណា Array ទាំងពីរមានធាតុស្មើគ្នា ។
3	public static void fill(int[] a, int val) : Assigns តម្លៃជាក់លាក់ទៅអោយគ្រប់ធាតុនៃ Array Int ។ Method តែមួយត្រូវបានប្រើប្រាស់រាល់គ្រប់ Primitive Data Types (Byte, short, Int etc.) ទាំងអស់ ។
4	public static void sort(Object[] a) : តម្រៀបតម្លៃជាក់លាក់នៃ Array of Objects តាមលំដាប់កើន ដោយអាស្រ័យទៅលើធាតុនៃ Array ។ Method តែមួយត្រូវបានប្រើប្រាស់រាល់គ្រប់ Primitive Data Types (Byte, short, Int etc.) ទាំងអស់។

សមាជិក

ក្រុមអ្នកស្រាវជ្រាវ



ល. ស៊ីម សុភារត្នី
simroth@ymail.com



ល. សែត យុត
syuth89@gmail.com



ល. អ៊ាប សុជាតិ
eamssocheatsmart@gmail.com



ល. ស្លេ: អុលវី
slehulvy@gmail.com



ល. សៃ វិចិត្រ
sai.vichet70@gmail.com

ក្រុមផលិតវីដេអូ



ល. ឈុន បញ្ចារតន៍
chhunpanharath@gmail.com



ល. ហង្ស បូរី
houngboreyrupp@gmail.com



ល. ព្រាប វិទ្ធី
Itpreap.vuthy@gmail.com



ល. ហួ ឈុនឡេង
huochhunleng@yahoo.com

មេរៀនបន្ទាប់ និងធ្វើការបង្ហាញពី Modifier Types