

JAVA BEGINNER

ARRAY



Abstract of New Technology

Website <http://www.antkh.com>

Tel 010 / 016 66 66 53

Prepared By Tum Sakal

Tel 087 36 31 30

Mail tumsakal.ts@gmail.com

Facebook [sak kal](#)

ARRAY

- Array គឺជាប្រភេទមួយនៃ Variable ដែលមានលទ្ធភាពរក្សាទុកសំណុំនៃតម្លៃ វិទិន្នន័យច្រើននៅក្នុងខ្លួនវាតែមួយ។

Fruit List:

Apple	Banana	Mango	Orange	Lemon
-------	--------	-------	--------	-------

Student List:

Id	Name	Gender	Phone
STD001	Dara	Male	012 99 88 77
STD002	Nita	Female	016 77 88 99

ARRAY

- Array មានលក្ខណៈពិសេសបីគឺ៖
 - ធាតុនីមួយៗមានប្រភេទទិន្នន័យដូចគ្នា
 - Zero-Base Index
 - នឹង Fixed Size

Fruit List:

Apple	Banana	Mango	Orange	Lemon
0	1	2	3	4

Student List:

	0	1	2	3
	Id	Name	Gender	Phone
0	STD001	Dara	Male	012 99 88 77
1	STD002	Nita	Female	016 77 88 99

TYPE OF ARRAY

- Array មានពីរប្រភេទគឺ:
 - Single Dimensional Array
 - Multi-Dimensional Array: 2D, 3D, ...ND

SINGLE DIMENSIONAL ARRAY

- Declare Array:

```
data_type[] name;
```

- Instantiate Array:

```
data_type[] name = new data_type[N];
```

- Initialize Array:

```
data_type[] name = new data type[]{v1,v2, ..., vN};
```

```
data_type[] name = {v1,v2, ..., vN};
```

ACCESSING ARRAY ELEMENT

- ដើម្បីប្រើប្រាស់ធាតុណាមួយរបស់ **Array** អ្នកត្រូវស្គាល់លេខទីតាំងរបស់ធាតុនោះជាមុនសិន

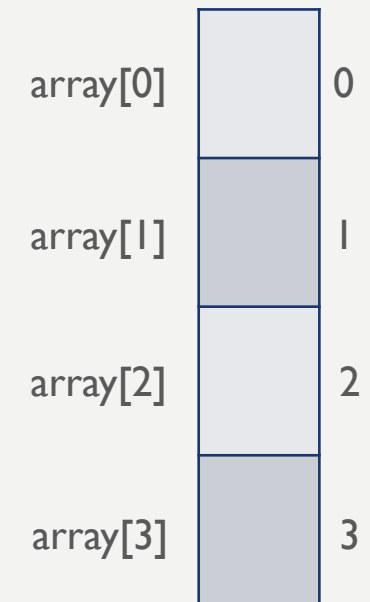
`array[index]`

- ផ្តល់តម្លៃ:

`array[index] = value;`

- ចាប់យកតម្លៃ:

`variable = array[index];`



ARRAY LENGTH

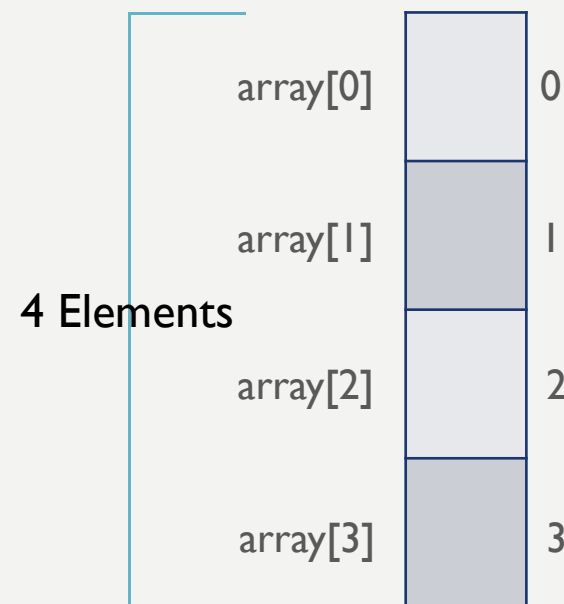
- Length សំដៅទៅលើចំនួនធាតុសរុបរបស់ Array
- ទាញយកចំនួនធាតុរបស់ Array:

Return

int

Field

array.Length

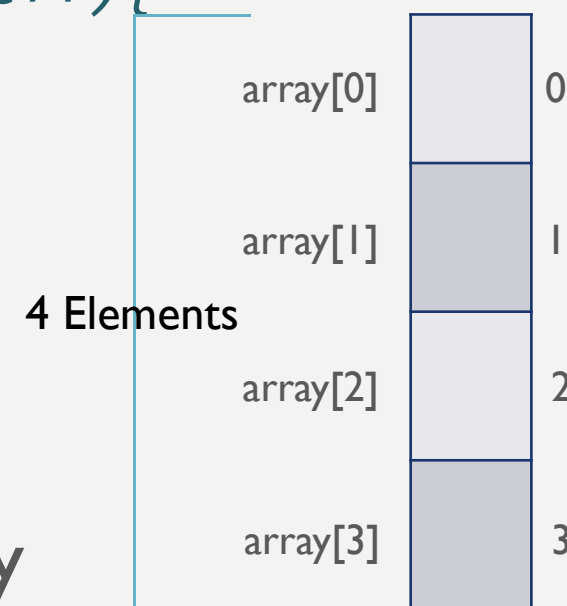


FOR LOOP

- “for” loop ទៅកាន់ធាតុនីមួយៗរបស់ Array តាមរយៈ index

```
for(int i = 0; i < array.Length; i++){  
    array[i]  
}
```

- Variable “i” មានតម្លៃចាប់ពី 0 ទៅដល់
(array.length – i)
- “array[i]” នឹងចង្អុលទៅកាន់ធាតុរបស់ array
ទៅតាមតម្លៃ “i” នៃ For loop

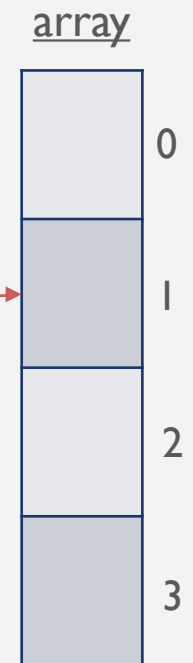


ENHANCED FOR LOOP

- “Enhanced For Loop” loop ទៅកាន់ធាតុនិមួយៗរបស់ Array ដោយផ្ទាល់ៗ

```
for(data_type variable : array){  
    statements...  
}
```

- Variable តំណាងអោយធាតុនិមួយៗរបស់ Array ពីដើម រហូតដល់ធាតុចុងក្រោយនៃ Array



2D ARRAY

- 2D Array ប្រើសំរាប់រក្សាទុកទិន្នន័យក្នុងទំរង់ជា table ដែលមានសំនុំទិន្នន័យគិតជា row និងបែងចែកទិន្នន័យទៅតាម Column។

Student List:

4 Columns					
		<u>Id</u>	<u>Name</u>	<u>Gender</u>	<u>Phone</u>
		0	1	2	3
2 rows	0	STD001	Dara	Male	012 99 88 77
	1	STD002	Nita	Female	016 77 88 99

2D ARRAY

- Declare array:

`data_type[][] name;`

- Instantiate array:

`data_type[][] name = new data_type[r][c];`

- Initialize array:

`data_type[][] name = new data_type[][]{ {v1, ...}, {v1, ...} };`

`data_type[][] name = { {v1, ...}, {v1, ...} };`

ACCESSING ELEMENT

- Access row

array[row_index]

- Access array element in row

array[row_index][col_index]

	0	1	2	3
0	STD001	Dara	Male	012 99 88 77
1	STD002	Nita	Female	016 77 88 99

array[0]

array[1][3]

ARRAY LENGTH

- ចំនួនធាតុរបស់ Array: *array.Length*
- ចំនួនធាតុរបស់ Row : *array[row_index].Length*

	0	1	2	3
0	STD001	Dara	Male	012 99 88 77
1	STD002	Nita	Female	016 77 88 99

array.Length -> 2

array[0].Length -> 4

FOR LOOP

- Loop ទៅកាន់ជួរដេកនិមួយៗ:

```
for(int i = 0; i < array.Length; i++){
```

```
...
```

```
}
```

- Loop ទៅកាន់ធាតុនិមួយៗ នៃជួរដេកនិមួយៗ:

```
for(int i = 0; i < array.Length; i++){
```

```
    for(int j = 0; j < array[i].Length; j++){
```

```
        ...
```

```
    }
```

```
}
```

FOREACH LOOP

- Loop ទៅកាន់ជួរដេកនិមួយៗ:

```
for(dataType[] row : array){  
    ...  
}
```

- Loop ទៅកាន់ធាតុនិមួយៗ នៃជួរដេកនិមួយៗ:

```
for(dataType[] row : array){  
    for(datatype element : row){  
        ...  
    }  
}
```


ARRAYS – UTILITY CLASS

- java.util.Arrays ជា Class មួយដែលសំរួលដល់ការប្រើប្រាស់ Array និងមាន Static Methods ជាច្រើនដែលអាចប្រើសំរាប់
 - Search
 - Sort
 - Copy
 - Fill
 - Convert to String

SEARCHING

Return

int

int

Method

binarySearch(T[] array, T key)

binarySearch(T[] array, int from_index,
int to_index, T key)

*** T អាចជា byte, short, int, long, float, double រឺជា char

SORTING

Return

Method

void

sort(T[] array)

void

sort(T[] array, int from_index, int to_index)

*** T អាចជា byte, short, int, long, float, double រឺជា char

COPYING

Return

Method

T[]

copyOf(T[] original, int new_length)

T[]

copyOfRange(T[] original, int from_index, int to_index)

*** T អាចជា byte, short, int, long, float, double រឺជា char

FILLING

Return

Method

Void

fill(T[] array, T value)

Void

fill(T[] array, int from_index, int to_index, T value)

*** T អាចជា byte, short, int, long, float, double រឺជា char

TO STRING

Return

String

Method

toString(T[] array)

*** T អាចជា byte, short, int, long, float, double រឺជា char