## Arrays in Java

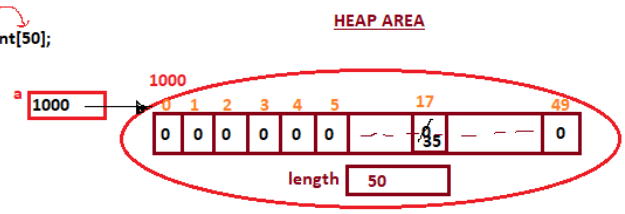
### Traditional/Variable Approach of Storing the data:

```
int a;   int i;   int q;   int y;
int b;   int j;   int r;   int z;
int c;   int k;   int s;   int aa;
int d;   int l;   int t;   int ab;
int e;   int m;   int u;   int ac;
int f;   int n;   int v;   .....
int g;   int o;   int w;   .....
int h;   int p;   int x;   int ax;
```

i. Creation is tedious & inefficient
ii. Accessing is extremely difficult
iii. Length of the code increases
iv. Suitable to store only small amount of data

### Modern/Array Approach of Storing of data:

```
int[] a = new int[50];
```




```
a[17] = 35;

a[41] = 100;

S.o.p(a.length); //50
```

i. Creation is Simple
ii. Accessing is Easy
iii. Length of the code reduces
iv. Suitable to store large volumes of data

Types of Arrays

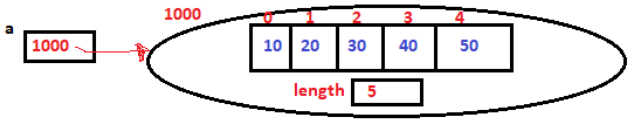
- Regular/ Rectangular Arrays
  - 1D
  - 2D
  - 3D
  - 4D
  - ...
  - ...
  - ...
  - ND
- Irregular/ Jagged Arrays
  - 2D
  - 3D
  - 4D
  - ...
  - ...
  - ...
  - ND

## 1D Regular Array

**Scenario:** To Store the marks of 5 Students

**Declaration:** int[] a = new int[5];

**Memory Map:**




**Size:** In a 1D Array,
length => No. of columns

a.length => 5 => No. of columns

**Code:**

```
S.o.p("Enter the marks:");
a[0] = scan.nextInt();

S.o.p("Enter the marks:");
a[1] = scan.nextInt();

S.o.p("Enter the marks:");
a[2] = scan.nextInt();

S.o.p("Enter the marks:");
a[3] = scan.nextInt();

S.o.p("Enter the marks:");
a[4] = scan.nextInt();
```

Loops

1. for
2. while
3. do-while
4. for-each

a. Initialization
b. Termination Condition
c. Update Statement

```
for(int i=0;   i<=4;        ++i )
{                  a.length-1
    S.o.p("Enter the marks:");
    a[ i ] = scan.nextInt();
}
```

2D Regular Array

Multi-Dimensional Arrays in Java are "Array of Arrays"

**Scenario: To Store the marks of 5 Students each in 3 Classrooms**

**Declaration:** `int[][] a = new int[3][5];`

Classrooms/ Rows

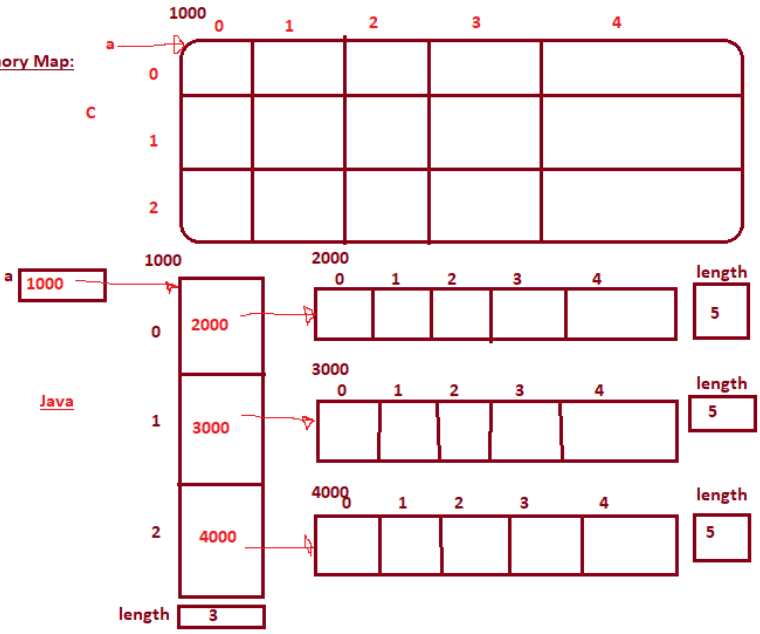
Students/ Columns

a.length => 3

a[0].length => 5

In a 2D Array,
a.length ==> No.of Rows
a[i].length ==> No. of Columns in the ith Row

**Memory Map:**



3D Regular Array

Scenario: To Store the marks of 5 Students each in 3 Classrooms each in 2 Schools

Declaration:

int[][][] a = new int[2][3][5];

Schools/Blocks

Classrooms/Rows

Students/Columns

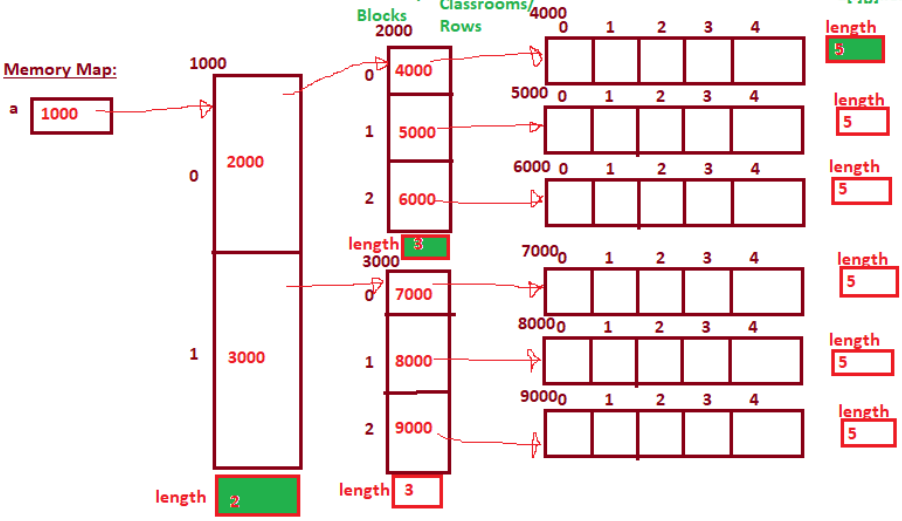
Size:

a.length ==> 2 ==> No. of Blocks

a[i].length ==> 3 ==> No. of Rows in the ith Block

a[i][j].length ==> 5 ==> No. of Columns in the ith Block jth Row

Memory Map:



## Assignment

## 4D Regular Array

**Scenario: To Store the marks of 5 Students each in 3 Classrooms each in 2 Schools each in 2 Cities**