

# 配列 p.50

- 配列: array

- ◆ 決まった個数の変数が並んだデータ構造(オブジェクト)
- ◆ 型の混在はできない

(1) 配列オブジェクトを保持するための変数を宣言

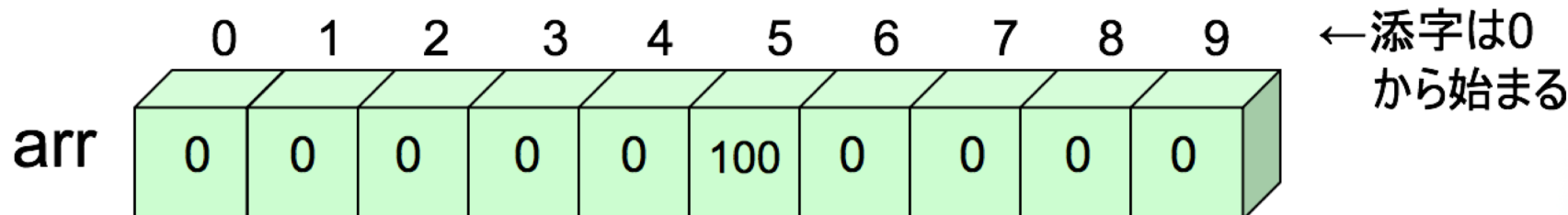
```
int[] arr;      または      int arr[];
```

(2) 配列オブジェクトを生成し変数に代入

```
arr = new int[10];
```

(3) 配列の要素に「もの」を格納

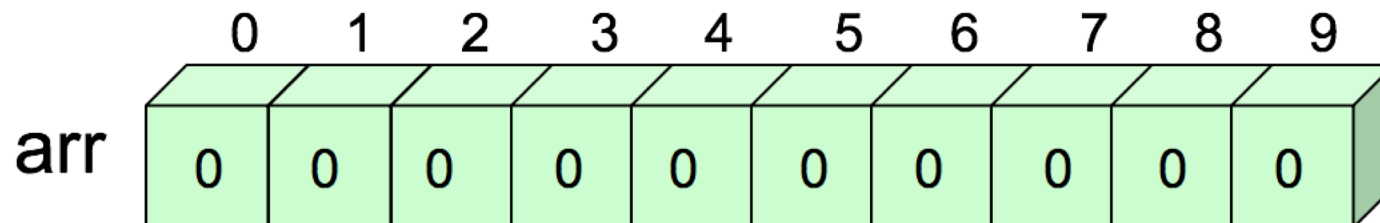
```
arr[5] = 100;      // 5は添字(index)と呼ぶ
```



# 配列の要素

- 配列オブジェクトを生成すると  
その要素はすべて初期化されている
  - ◆ 数値: 0
  - ◆ 文字: '¥0'
  - ◆ 論理値: false
  - ◆ クラス型: null

```
arr = new int[10];
```



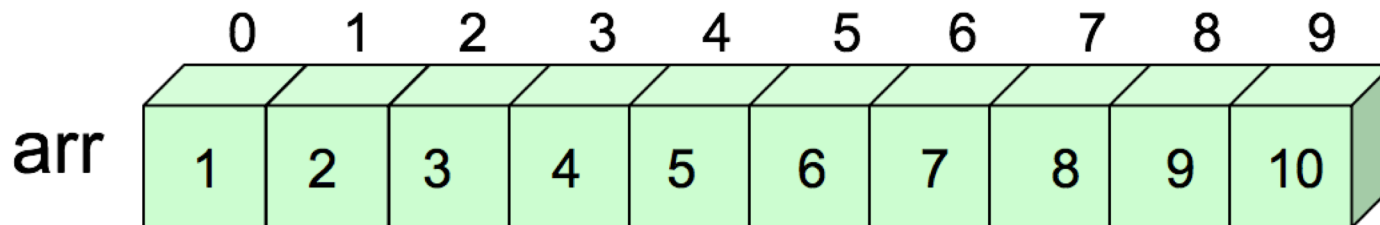
# length

- 配列オブジェクトは**インスタンス変数length**を持つ

```
arr = new int[10];
```

**arr.length** の値は 10

```
for(int i=0; i < arr.length; i++) {  
    arr[i] = i+1;           // 各要素に1から値を格納  
}
```



# for-each文 p.92

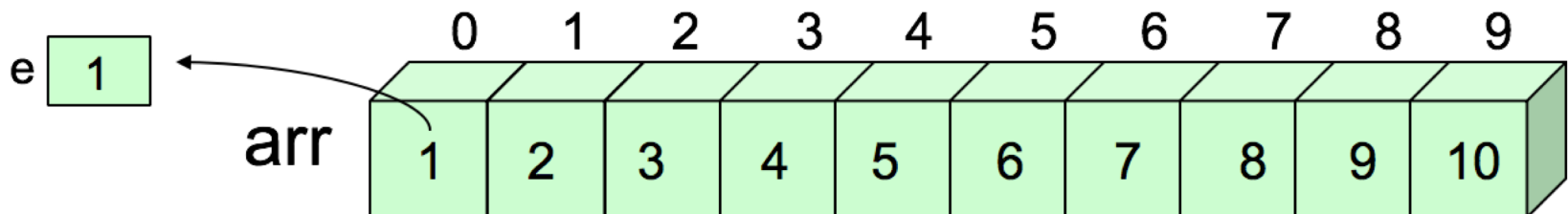
- 配列の要素を順に印字する例

```
for(int i=0; i < arr.length; i++) {  
    System.out.println(arr[i]);  
}
```



```
for(int e: arr) { // 配列オブジェクトの要素の値を順に局所変数eへ代入  
    System.out.println(e);  
}
```

※ eへの代入は、配列要素への代入にはならない！



# 配列の生成（初期値設定）

- 配列変数の宣言と同時に初期化を行える

```
int[] arr = {1,2,3,4,5,6,7,8,9,10};
```

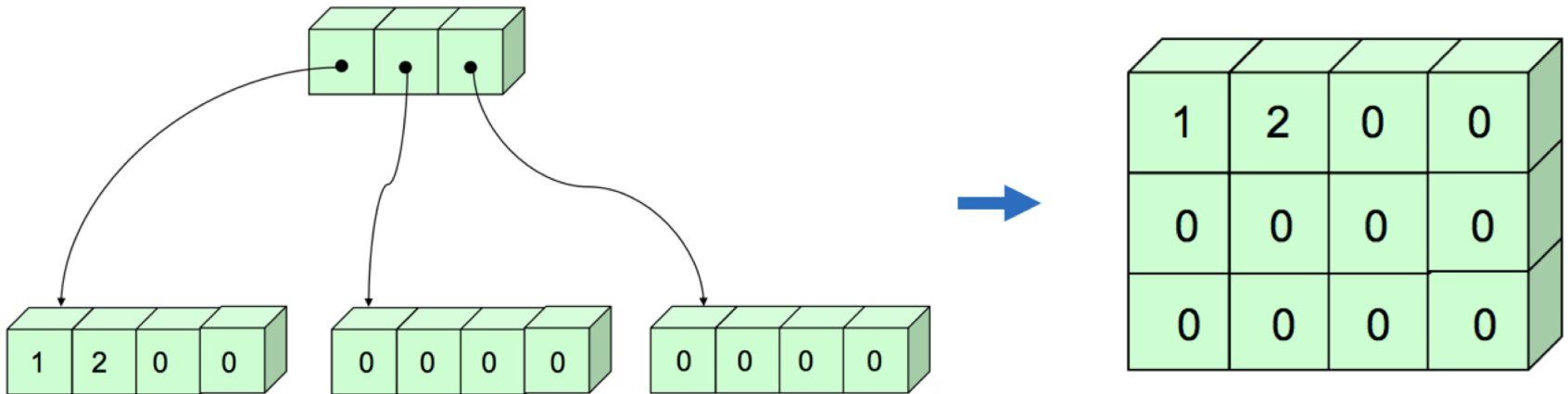
または

```
int[] arr = new int[] {1,2,3,4,5,6,7,8,9,10};
```

```
String[] words = {"Red", "Green", "Blue"};
```

# 2次元配列

```
int[][] a34 = new int[3][4];  
a34[0][0] = 1;  
a34[0][1] = 2;
```

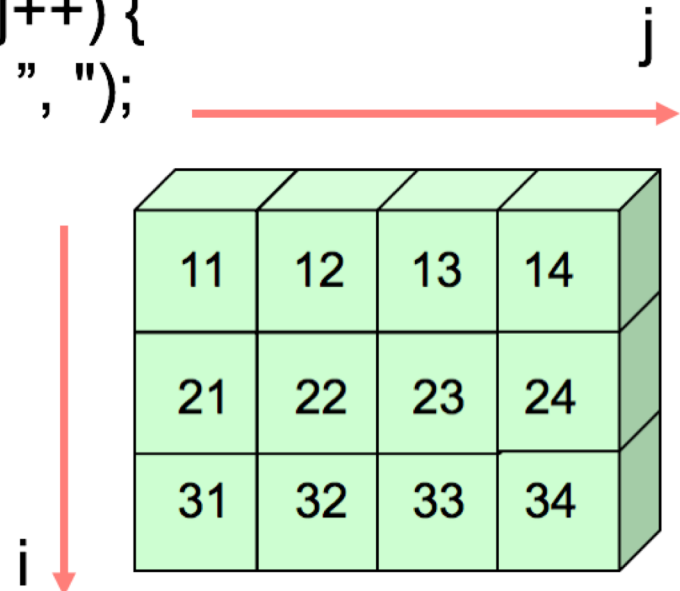


# 2次元配列

```
int[][] arr2 = { {11,12,13,14},  
                 {21,22,23,24},  
                 {31,32,33,34} };  
for (int i = 0; i < arr2.length; i++) {  
    for (int j = 0; j < arr2[i].length; j++) {  
        System.out.print(arr2[i][j] + ", ");  
    }  
    System.out.println();  
}
```

## 実行結果

```
11, 12, 13, 14,  
21, 22, 23, 24,  
31, 32, 33, 34,
```



# 「いびつな」配列も作れる

```
String[][] s = { { "a", "b", "c" },  
                 { "x", "y" } };  
for (int i = 0; i < s.length; i++) {  
    for (int j = 0; j < s[i].length; j++) {  
        System.out.print(s[i][j] + ", ");  
    }  
    System.out.println();  
}
```

## 実行結果

a, b, c,  
x, y,



# mainメソッドの引数

```
public class ArrayTest {  
    public static void main(String[] args) {  
        for(String s: args) {  
            System.out.print(s+", ");  
        }  
    }  
}
```

Javaアプリケーションの引数は  
String配列として受け渡される

## 実行結果

>java ArrayTest abc 123 xyz

abc, 123, xyz,

>java ArrayTest "abc 123" "x y z"

abc 123, x y z,

