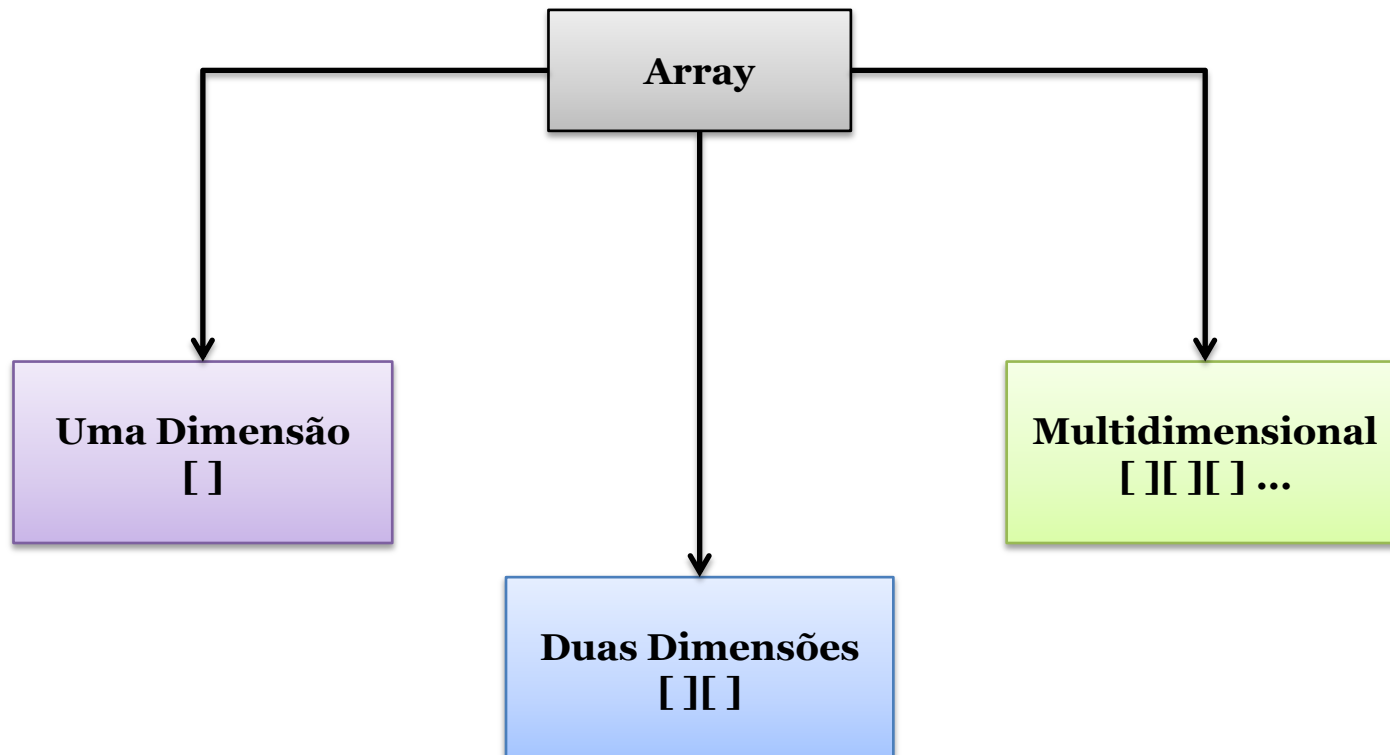


Java Orientado a Objetos

Array



Array ?



Declarando Array

Inicialização

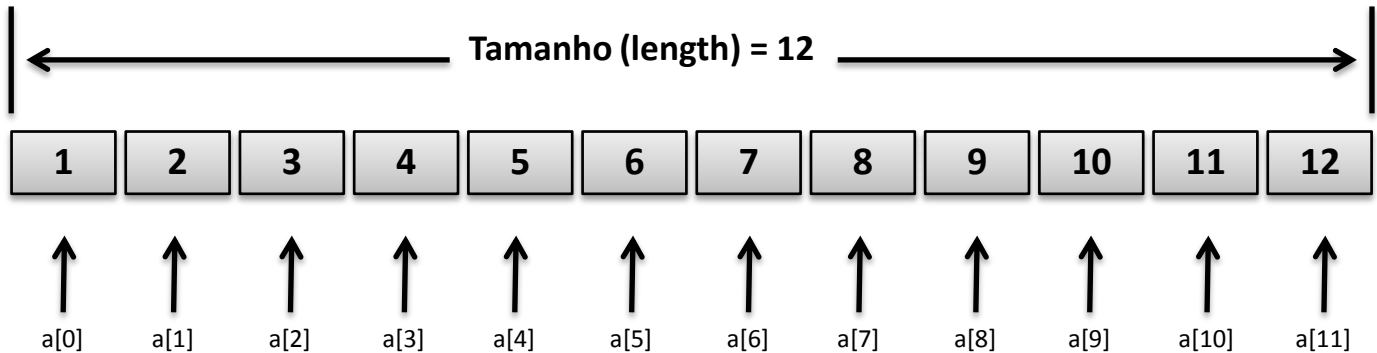
```
int a[] = new int[12];
```

||

```
int []a = {1,2,3,4,5,6,7,8,9,10,11,12};
```

Valores

Índices

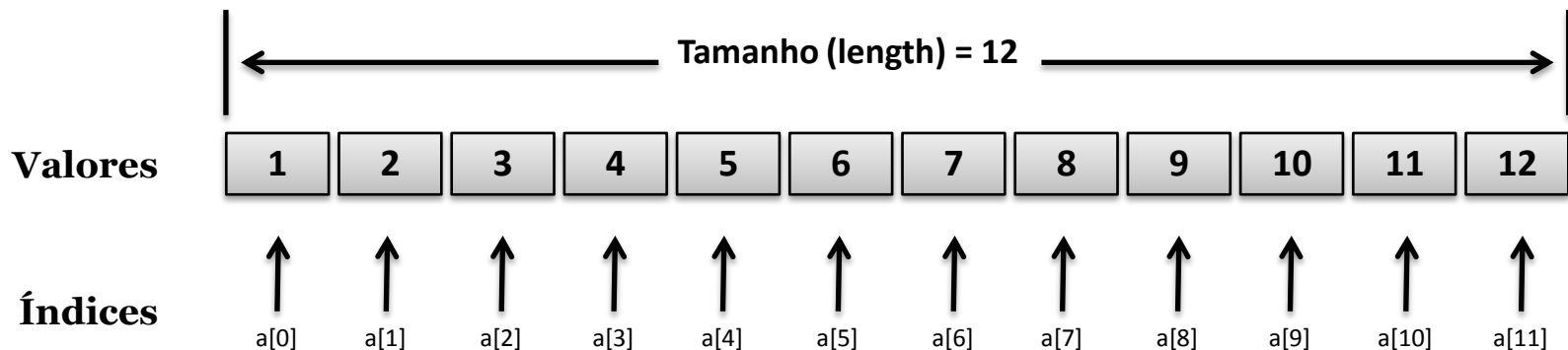


→ *“Pode guardar somente papel”
(Um único tipo de dado, anteriormente definido)*



Acessando um elemento do Array

```
public class Matriz {  
    public static void main(String[] args) {  
  
        int[] a = {1,2,3,4,5,6,7,8,9,10,11,12};  
        System.out.println(a[0]); //acessando o primeiro elemento do array  
        System.out.println(a[6]); //acessando o elemento do indice 6  
        System.out.println(a[a.length -1 ]); //acessando o último elemento  
  
        for (int i : a) { //percorre e imprime todos os elementos do array  
            System.out.println(a[i]);  
        }  
    }  
}
```

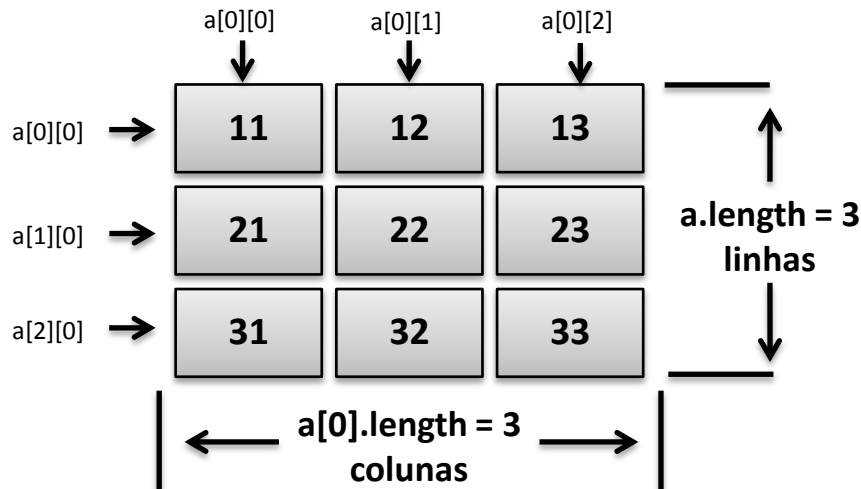


Arrays Multidimensionais

Inicialização

```
int a[][] = new int[3][3];
```

```
int [][]a = {{11,12,13},{21,22,23},{31,32,33}};
```



*“Pode guardar somente papel”
(Um único tipo de dado, anteriormente definido)*

Acessando um elemento de um Array Multidimensional

```
public static void main(String[] args) {  
  
    int[][] matriz = { { 11, 12, 13 }, { 21, 22, 23 }, { 31, 32, 33 } }; //constroi matriz  
  
    System.out.println(matriz.length); // imprime numero de linhas  
  
    System.out.println(matriz[0].length); // imprime numero de colunas  
  
    System.out.println(matriz[0][0]); // acessa elemento na linha [0] e coluna [0]  
  
    System.out.println(matriz[2][2]); // acessa elemento na linha [2] e coluna [2]  
  
    for (int linha = 0; linha < matriz.length; linha++) { // percorre todas linhas  
        for (int coluna = 0; coluna < matriz[linha].length; coluna++) { // percorre todas colunas  
            System.out.print(matriz[linha][coluna] + " "); // imprime matriz  
        }  
        System.out.println("\n");  
    }  
}
```

		a[0][0]	a[0][1]	a[0][2]
		↓	↓	↓
a[0][0] →	11	12	13	
a[1][0] →	21	22	23	
a[2][0] →	31	32	33	



Percorrendo Arrays com Enhanced-for

```
public static void main(String[] args) {  
    int[] a = {1,2,3,4,5,6,7,8,9,10,11,12};  
    for (int i=0; i < a.length; i++) {  
        //percorre array  
        System.out.println(a[i]);  
        //imprime todos os elementos do array  
    }  
}
```

```
public static void main(String[] args) {  
    int[] a = {1,2,3,4,5,6,7,8,9,10,11,12};  
    for (int i : a) {  
        //usando Enhanced-for  
        System.out.println(a[i]);  
        //imprime todos os elementos do array  
    }  
}
```



Manipulando Arrays com java.util.Arrays



```
public static void main(String[] args) {  
    int[] a = { 8, 4, 1, 9, 2, 5, 12, 3, 6, 10, 7, 11 };  
    System.out.println(Arrays.toString(a));  
    Arrays.sort(a);  
    System.out.println(Arrays.toString(a));  
    System.out.println("a[" + Arrays.binarySearch(a, 6) + "]");  
}
```

