

Chapitre 10

Collectionner un nombre fixe d'objets

1. Les tableaux à 1 dimension

1.1 Déclarer un tableau

- Syntaxe

Indique au compilateur que la variable `nomDuTableau` représente un tableau

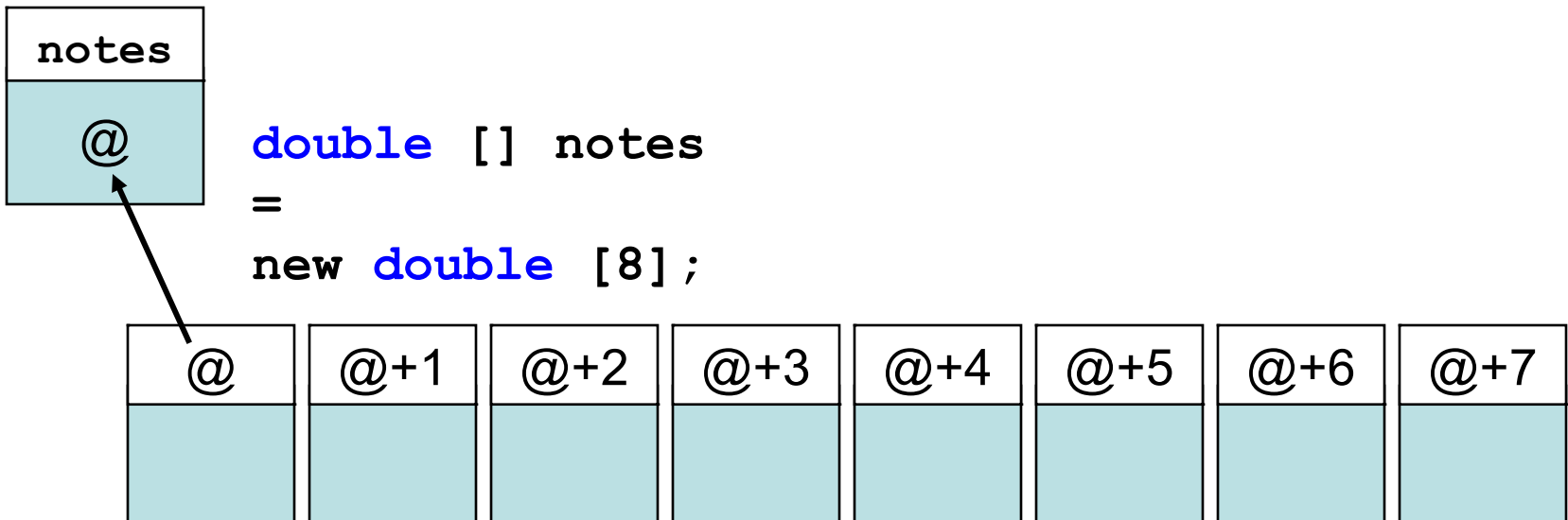
```
TypeDuTableau [] nomDuTableau;  
nomDuTableau = new TypeDuTableau [tailleDuTableau]
```

L'opérateur `new`

- réserve autant de cases mémoire consécutives qu'il est indiqué entre `[]`
- détermine l'adresse de la première case du tableau et la stocke dans `nomDuTableau`

- Exemple : déclarer un tableau de 8 notes

```
double [] notes;  
notes = new double [8];
```



- Remarques

- Il est possible de ne pas fixer définitivement la taille du tableau avant compilation en plaçant une variable entre les [] au lieu d'une valeur numérique :

```
double [] notes;
```

```
int nbNotes;
```

```
Scanner clavier = new Scanner(System.in);
```

```
System.out.println("Combien de notes à saisir?");
```

```
nbNotes = clavier.nextInt();
```

```
notes = new double [nbNotes];
```

- Remarques (suite)

- Les tableaux sont des objets
- La taille d'un tableau peut être connue à l'aide de la propriété `length`

```
System.out.print("Nombre de notes=" +  
                notes.length) ;
```

```
// affiche à l'écran Nombre de notes = 8
```

- Équivalence entre les deux déclarations suivantes:

```
double [] notes = new double [8] ;
```

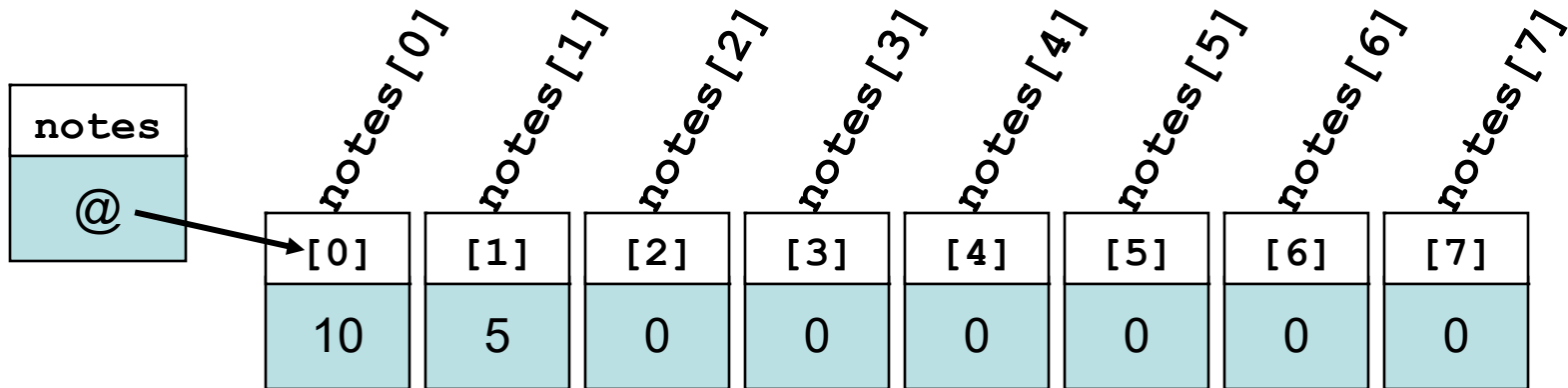
et

```
double [] notes ;
```

```
notes = new double [8] ;
```

1.2 Manipuler un tableau

- Accéder aux éléments d'un tableau
 - Première case du tableau : `nomDuTableau[0]`
 - Accès à la nième case : `nomDuTableau[n]`



– Exemple :

`notes[0] = 10;`

`notes[1] = 5;`

↖
Indice du tableau

- Accéder aux éléments d'un tableau
 - Remarques :
 - La première case est **numérotée 0 et non 1**
 - L'indice du tableau varie donc entre 0 et length-1
 - Exemple de programme (extrait) :

```
System.out.print("Combien de notes à saisir?");  
int nombre = clavier.nextInt();  
notes = new double [nombre];  
for (int i = 0; i < notes.length; i++)  
{  
    System.out.print("Entrer la note n°"+(i+1)+" : ");  
    notes[i]= clavier.nextDouble();  
}
```

- Initialiser un tableau

- Possibilité d'initialiser directement un tableau

```
double [] notes = {10, 5, 10, 18, 8, 13, 5, 9};
```

- La valeur `notes.length` prend automatiquement la valeur 8

- Les tableaux et les opérations arithmétiques

- La somme, la soustraction, la division ou la multiplication directes de deux tableaux sont des opérations impossibles.

Correcte

```
int [] tab1 = new int[10] ;  
int [] tab2 = new int[10];  
int [] somme = new int[10];  
for (i = 0 ; i < 10 ; i++)  
    somme[i] = tab1[i] + tab2[i];
```

Impossible

```
int [] tab1 = new int[10] ;  
int [] tab2 = new int[10];  
int [] somme = new int[10];
```

```
somme = tab1 + tab2;
```


2. Tableaux à 2 dimensions

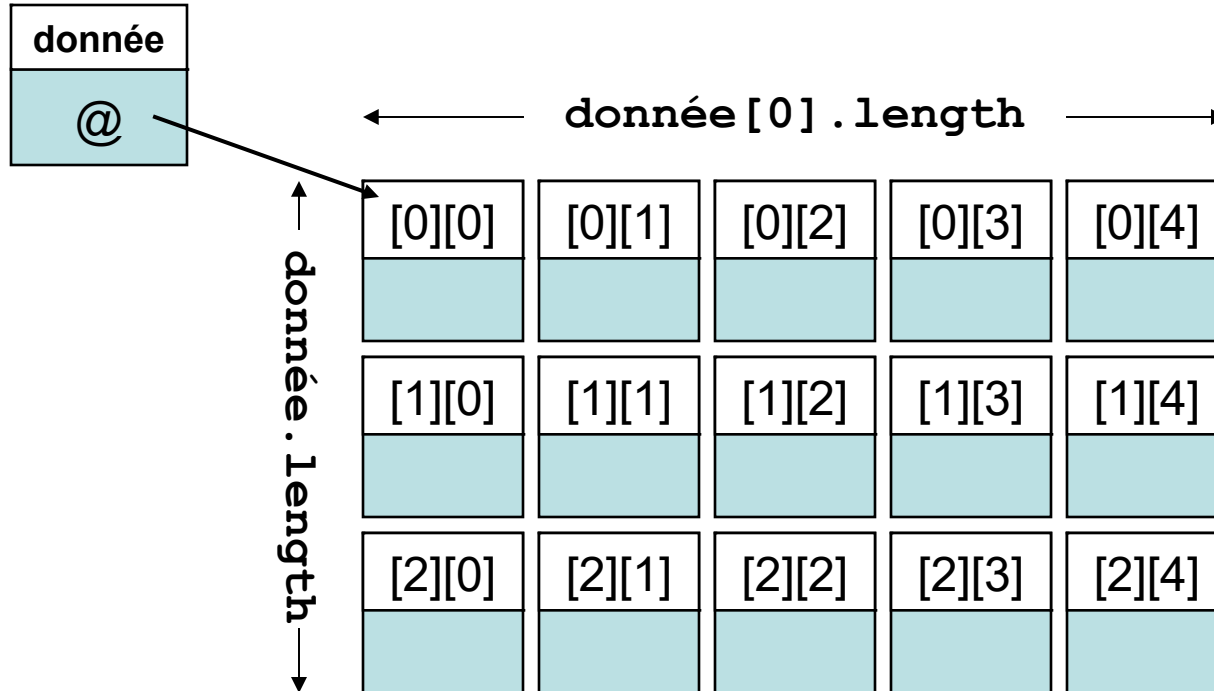
- Il est possible de travailler avec des tableaux de deux, trois, voire n dimensions.
- Exemple pour un tableau à deux dimensions

Déclaration :

```
int [][] donnée = new int [3][5];
```

- Le deuxième crochet permet d'indiquer qu'il y a une deuxième dimension
- [3] [5] indiquent respectivement le nombre de lignes puis de colonnes
- Il y a donc $3 * 5 = 15$ cases de type **int** de déclarées

- `donnée.length` permet de connaître le nombre de lignes.
- `donnée[0].length` permet de connaître le nombre de colonnes.



- Accéder aux éléments d'un tableau
 - Utiliser un indice pour les lignes et un indice pour les colonne :

```
for (int i=0; i<donnée.length; i++)  
    for (int j=0; j<donnée[0].length; j++)  
        donnée [i][j] = uneValeur;
```