

# Exercices sur les tableaux à deux dimensions

## Exercice 4.3.1 *Tableau de vente*

On va considérer un tableau à deux dimensions qui regroupe les informations relatives aux ventes de voitures dans une concession. Une première dimension sert à représenter les différents modèles de voitures (une colonne pour chaque modèle). Une deuxième dimension sert à représenter les ventes d'un vendeur de l'entreprise (une ligne par vendeur). Une case contient le nombre de voitures d'un modèle donné vendu par un vendeur X.

Par exemple :

	twingo	clio	mégane	vel satis
andré	0	3	2	0
ingemar	2	3	0	1
jean-jérôme	1	1	1	1
Cindy	5	1	0	0
Joey	1	1	2	0

### Question 1

On suppose qu'il y a 4 modèles et 5 vendeurs. Ecrivez un programme qui crée le tableau des ventes et lit au clavier les données permettant de le remplir.

### Question 2

Ecrivez un programme qui donne le nombre d'exemplaires vendus pour chacun des modèles.

### Question 3

On donne le prix de chaque modèle dans un second tableau à une seule dimension. Ecrire un programme qui calcule le chiffre d'affaire généré par chacun des vendeurs, c'est à dire le total de ses ventes exprimé en euros.

## Exercice 4.3.2 *occupation de salle*

Une salle de réunion peut être utilisée par différents employés d'une entreprise. La réservation se fait par plage d'une heure, de 8H00 du matin à 19H00. Chaque plage d'une heure commence à l'heure pile (par exemple, il y a une plage 9H00-10H00 mais il n'y a pas de plage 9H15-10H15).

---

Un tableau de booléens à deux dimensions est utilisé pour représenter si la salle est occupée (valeur true) ou disponible (valeur false) pendant une semaine. Une dimension est utilisée pour coder les jours ouvrables de 0 (lundi) à 4 (vendredi). L'autre dimension est utilisée pour les plages horaires de 0 (8H00-9H00) à 10 (18H00-19H00). Chaque case correspond à la réservation de la salle pour une plage d'un jour donné.

---

```
boolean[][] occupee =  
    { {false,true,true,true,false,true,true,false,true,false},  
      {false,true,true,true,false,true,true,false,true,false},  
      {false,false,true,true,false,false,true,false,true,false},  
      {true,true,false,true,false,true,true,false,true,true},  
      {false,true,false,true,false,true,true,false,false,false}};
```

---

### Question 1 *affichage*

écrire un programme qui déclare la variable `occupee` comme ci-dessus et qui affiche l'occupation de la salle de façon intelligible (par exemple : salle occupée le mardi de 9H00 à 10H00).

### Question 2 *taux d'occupation*

écrire un programme qui calcule le taux d'occupation de la salle, c'est à dire le nombre de plages réservées divisé par le nombre total de plages.

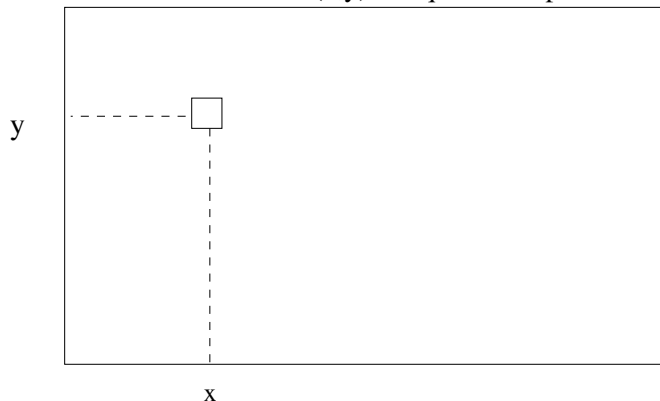
### Question 3 *reservation*

écrire un programme qui permet à l'utilisateur de réserver une plage. Il faut vérifier que la salle est libre au moment demandé.

## Exercice 4.3.3 *image*

### Question 1

On veut représenter une photo en noir et blanc comme un ensemble de points, avec pour chaque point un niveau de gris codé par un entier compris entre 0 (pour noir) et 255 (blanc). Les points ont des coordonnées cartésiennes (x,y) indiquant leur position sur l'image.



Ecrivez un programme (méthode `main`) qui contienne successivement :

- 
- la saisie d’une image au clavier. Pour cela, on pourra demander à l’utilisateur de rentrer le niveau de gris de chaque point. Il faut vérifier que ce niveau de gris est bien compris entre 0 et 255.
  - le calcul du nombre de points blancs sur cette image et le pourcentage des points qui sont blancs. Ce pourcentage pourra être de type entier ou de type nombre à virgule.
  - l’éclaircissement de l’image obtenu en ajoutant 30 à chaque niveau de gris, sans dépasser toutefois le nombre de 255 (par exemple 100 sera transformé en 130, 255 restera 255 et 240 deviendra 255).

## **Question 2**

On peut représenter une image en couleur en donnant pour chaque point un niveau d’intensité pour une des couleurs fondamentales : rouge, vert et bleu. Ce niveau sera de 0 pour pas du tout de cette couleur, 255 pour le maximum de la couleur. On obtiendra les différentes nuances par mélange de couleur (par exemple, du rouge et du vert pour le jaune). Comment peut-on représenter une image en couleur dans une variable Java en n’utilisant que des constructions Java vues dans le cours NFA001 ? On demande une explication en 5 lignes maximum.