

Support de cours

Cours:

Initiation à la programmation (en Java)

Vidéo:

Init-JAVA-07-1-types

Concepts (extraits des sous-titres générés automatiquement) :

Façon générale. Programme final. Premier indice du tableau. Indice de la ligne. Sous-problèmes. Pion de sa couleur. Types nécessaires. Façon suivante. Colonne de son choix. Valeurs différentes. Cas de notre exemple. Ligne d'indice. Affectation de la façon suivante. Dernière leçon. Haut de la grille.



[vers la recherche de séquences vidéo](#)
(dans Initiation à la programmation (en Java).)



[vers la vidéo](#)

Center for Digital Education. Plus de matériel de soutien pédagogique ici :

<https://www.epfl.ch/education/educational-initiatives/cede/educational-technologies-gallery/boocs-en/>

Puissance 4 : introduction

Initiation à la programmation (Java)

Jamila Sam, Vincent Lepetit et Jean-Cédric Chappelier

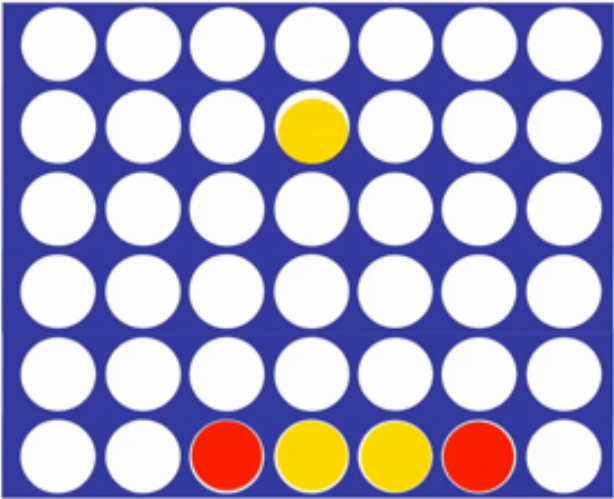
...

notes

résumé

0m 0s





Dans cette dernière leçon, nous allons voir comment aborder un projet plus ambitieux que ce que nous avons considéré jusqu'ici. C'est-à-dire comment décomposer ce projet en sous-problèmes plus faciles à résoudre jusqu'à ce que l'on sache écrire le programme final. Pour illustrer nos propos, nous avons choisi un jeu de puissance 4. Vous connaissez sans doute les règles du puissance 4. Cela se joue à deux joueurs avec une grille de sept colonnes ayant six cases chacune.

notes

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

résumé

.....

.....

.....

.....

.....

.....

0m 1s



```

| | | | | | |
| | | | | | |
| | |X| | | |
| |X|O|O|X|O|
| |O|X|X|X|O|
|O|O|X|X|O|X|
==1=2=3=4=5=6=7==

```

Joueur 0 : entrez un numero de colonne
5

```

| | | | | | |
| | | | | | |
| | |X|O| | |
| |X|O|O|X|O|
| |O|X|X|X|O|
|O|O|X|X|O|X|
==1=2=3=4=5=6=7==

```

Le joueur 0 a gagne !

Chaque joueur laisse tomber tout à tour un pion de sa couleur dans la colonne de son choix jusqu'à ce qu'un joueur réussisse à aligner quatre pions de sa couleur ou que la grille soit complètement remplie.

notes

résumé

0m 37s



1. N'essayez pas d'écrire tout le programme en une seule fois !
 - ▶ Décomposez le problème en sous-problèmes pour développer le programme par étapes ;
 - ▶ À chaque étape, testez le code développé.
2. Tout d'abord, identifiez les types nécessaires ;
3. Identifiez les méthodes qui portent sur ces types, écrivez-les et testez-les au fur et à mesure ;

Tout d'abord, une précision, nous n'allons pas développer une jolie interface graphique, ce qui nous intéresse ici c'est que notre programme suive la mécanique du jeu et nous allons nous contenter d'un affichage de la grille et des pions avec des caractères, c'est-à-dire quelque chose qui ressemblerait à ceci. De façon générale, comment devez-vous procéder quand vous êtes face à un problème un peu conséquent tel que celui-ci ? Surtout, il ne faut pas essayer d'écrire tout le programme en une seule fois, vous n'allez pas vous en sortir. Il faut savoir décomposer le problème en sous-problèmes pour écrire le programme par étape et à chaque étape, il faut tester le code que vous aurez écrit pour continuer sur des bases solides. Tout d'abord, il faut identifier les types nécessaires pour que le programme puisse fonctionner, c'est-à-dire pour pouvoir représenter les données dont le programme a besoin. Dans le cas de notre exemple, il s'agit essentiellement de pouvoir représenter la grille. Ensuite, il faudra identifier les méthodes

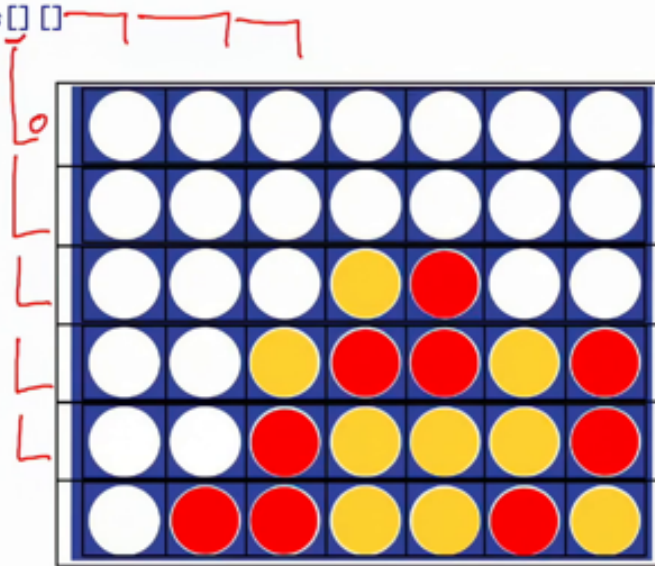
notes

résumé

0m 49s



`type_des_elements[] []`



qui portent sur ces types et les tester au fur et à mesure qu'elles sont codées. Enfin, quand une méthode est difficile à écrire, on peut introduire une méthode supplémentaire pour chacun des points un peu difficiles. J'ai dit qu'il fallait commencer par identifier les types. Alors, qu'est ce que ça veut dire ? On va commencer par identifier les données dont a besoin le programme et trouver comment représenter ces données. En l'occurrence, pour notre programme de puissance quatre, la donnée principale est la grille et les pions qu'elle contient ou plus exactement, on veut savoir pour chaque case ce que contient la case. Est-ce que c'est un pion rouge ou un pion jaune ou alors rien du tout ? La façon la plus simple et sans doute la plus pratique de représenter la grille dans notre programme Java est d'utiliser un tableau à deux dimensions où chacun des éléments du tableau à deux dimensions correspond à une case de la grille. Nous allons décidé que le premier indice du tableau correspond à l'indice de la ligne, où la ligne d'indice 0 correspond à la ligne tout en haut de la grille et que le deuxième indice correspond à l'indice de la colonne et là encore, il nous faut décider d'une convention

notes

résumé

2m 1s



type_des_elements?

Nous allons utiliser `int` et définir les constantes :

```
private final static int VIDE = 0;
private final static int JAUNE = 1;
private final static int ROUGE = 2;
```

et on va dire que la colonne la plus à gauche de la grille a un indice 0.

notes

résumé

3m 25s

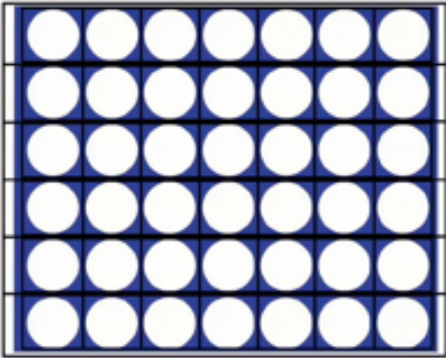


Types de notre programme

```
private final static int VIDE = 0;
private final static int JAUNE = 1;
private final static int ROUGE = 2;

...

int[] grille;
```



Il nous reste à choisir le type des éléments du tableau qui correspond à la grille. Nous allons utiliser ici le type int et définir les constantes de type int. Les trois constantes vont s'appeler VIDE, JAUNE et ROUGE. Elles vont être initialisées toutes les trois à des valeurs différentes ce qui fait que nous allons pouvoir nous servir de ces trois constantes pour donner des valeurs aux pions ou aux cases de notre grille. Il y a d'autres façons de procéder, notamment ce qu'on appelle les types énumérés mais nous n'allons pas les voir dans cette vidéo.

notes

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

résumé

.....

.....

.....

.....

.....

3m 30s



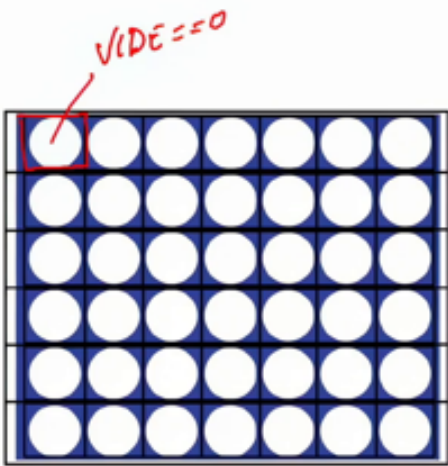
Types de notre programme

```
private final static int VIDE = 0;
private final static int JAUNE = 1;
private final static int ROUGE = 2;

...

int[] [] grille;

-> grille[0][0] = VIDE;
-> grille[2][3] = JAUNE;
```



Changer la grille va donc se faire de la façon suivante : par exemple, cette instruction-ci va mettre la valeur VIDE, c'est-à-dire 0 dans la case du tableau grille qui est sur la ligne 0 et sur la colonne 0, c'est-à-dire, si on respecte les conventions qu'on a décrites quelques transparents plus tôt, il s'agit de cette case-ci dans laquelle on va mettre la valeur VIDE (=0)

notes

résumé

4m 13s



notes

4m 49s

