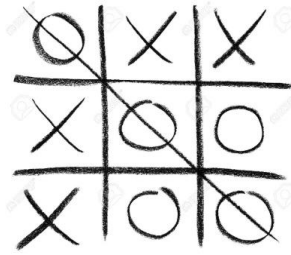


Mini-projet Tic-Tac-Toe



Date d'échéance : Avant le jeudi 10 Janvier 2019.

Mission : Écrire un programme java qui simule le jeu Tic-Tac-Toe à la console en utilisant le Scanner.

Concepts : Tableaux 2D, String, Notions de classes et objets.

Objectifs :

1. Comprendre l'utilisation des classes et objets.
2. Utilisation des tableaux 2D.
3. Utilisation et manipulation de String.
4. Comprendre l'utilisation des nombres aléatoire en Java.

2 niveaux de difficulté : (choisir la difficulté à effectuer)

1. **Moyen** : Implémenter une IA qui permet de jouer contre l'ordinateur. (+5 points)
2. **Facile** : Compléter les fonctions du projet (+15 points)

Description du projet :

Voir le lien suivant :

<https://fr.wikipedia.org/wiki/Tic-tac-toe>

Tester le jeu sur Google

<https://www.google.com/search?q=tic+tac+toe>

Implémentation de la classe TicTacToe :

Les caractères de TicTacToe doivent être toujours une de ces marques (caractères) :

- ' - ' : case vide
- ' X ' : la marque du joueur X
- ' O ' : La marque du joueur O

Lors de l'initialisation de la table de jeu, celle-ci doit être vide (les cases vides portent le caractère ' - ')

```
La table de jeu actuelle:  
- - -  
- - -  
- - -
```

Après le 1^{er} message qui demande au joueur X de choisir une case

```
Le joueur X, Saisissez une ligne et une colonne vide pour placer votre marque!  
[
```

Le joueur X doit entrer les coordonnées de la cellule qu'il souhaite remplir (exemple ligne 1 et colonne 1)

```
Le joueur X, Saisissez une ligne et une colonne vide pour placer votre marque!  
1  
1  
]
```

La table de jeu actuelle doit devenir comme suit :

```
[La table de jeu actuelle:  
X - -  
- - -  
- - -
```

Ensuite c'est le tour du joueur O :

```
Le joueur O, Saisissez une ligne et une colonne vide pour placer votre marque!
```

Le joueur O doit entrer les coordonnées de la cellule qu'il souhaite remplir (exemple ligne 2 et colonne 2). La table de jeu devient alors comme suit :

```
Le joueur O, Saisissez une ligne et une colonne vide pour placer votre marque!  
2  
2  
[La table de jeu actuelle:  
X - -  
- O -  
- - -
```

Si le joueur choisit une cellule non vide ou dépasse la taille du tableau, le programme demande au joueur de choisir la bonne cellule vide

```
La table de jeu actuelle:
X  -  -
-  -  -
-  -  -

Le joueur O, Saisissez une ligne et une colonne vide pour placer votre marque!
1
1
Le joueur O, Saisissez une ligne et une colonne vide pour placer votre marque!
1
4
Le joueur O, Saisissez une ligne et une colonne vide pour placer votre marque!
1
2
La table de jeu actuelle:
X  O  -
-  -  -
-  -  -
```

La partie est gagnée par le joueur qui réussit à occuper une ligne ou une colonne ou une diagonale avec sa marque. Exemple :

```
Le joueur X, Saisissez une ligne et une colonne vide pour placer votre marque!
3
1
Current board layout:
X  O  X
X  O  O
X  -  -

X Wins!
```

Travail à faire :

Le programme contient 2 classes (Main.java et TicTacToe.java)

Attention : ne pas modifier le fichier Main.java

- Compléter les fonctions de la classe TicTacToe.java suivantes (**modifiez que les fonctions qui apparaissent dans le tableau**) :

public void initializeBoard()	Permet de remplir la table de jeu avec des '-'
public void printBoard()	Affiche la table de jeu actuelle
public boolean isBoardFull()	teste si la table est pleine ou pas (retourne un booléen)
private boolean checkRowsForWin()	Teste si les lignes sont gagnantes (retourne un booléen)
private boolean checkColumnsForWin()	Teste si les colonnes sont gagnantes (retourne un booléen)
private boolean checkDiagonalsForWin()	Teste si les diagonales sont gagnantes (retourne un booléen)
public boolean placeMark(int row, int col)	Place la marque du joueur actuelle dans la cellule qu'il a choisie

Critère de notations :

- Un programme qui ne compile pas – 0 points
- Un programme 24h en retard – 50% de la note maximale
- Un programme 48h en retard – 0 points

Travail à rendre :

Vous devez rendre un rapport .pdf et les 2 fichiers (Main.java et TicTacToe.java) fonctionnels.