

Semaine 4 : utilisation d'une API (bataille navale)

L'objectif de ce TD-TP est de réaliser le jeu de la bataille navale en mode texte en se servant des classes fournies, dans un paquetage que vous nommerez `maBatailleNavale`.

Vous connaissez sûrement le jeu de la bataille navale : dans ce jeu de société à deux joueurs, les joueurs doivent placer des « navires » sur une grille de dimensions 10x10 tenue secrète et tenter de « toucher » les navires adverses. Le gagnant est celui qui parvient à torpiller complètement les navires de l'adversaire avant que tous les siens ne le soient.

Dans le travail demandé, on ne se préoccupe pas de la première phase du jeu dans laquelle les joueurs placent secrètement leurs navires sur la grille, on s'intéresse plutôt à la phase principale de jeu dans laquelle, tour à tour, les joueurs sélectionnent une case de la grille adverse masquée afin de découvrir s'ils ont « touché », voire « coulé » un navire adverse (quand toutes les cases du navire sont touchées), ou bien si leur coup est « à l'eau ». Pour simplifier, on considèrera uniquement le cas de deux joueurs « aléatoires » jouant l'un contre l'autre. Un certain nombre de classes sont fournies (binaire et documentation), dans un paquetage nommé `batailleNavale`, qui réalisent l'essentiel des traitements pour ce jeu.

	1	2	3	4	5	6	7	8	9	10
A	■	■						■		
B								■		
C		■	■	■	■			■		
D								■		
E				■				■		
F				■				■		
G				■				■		
H								■		
I							■	■		
J		■	■	■						

1. Commencez par explorer la documentation pour vous familiariser avec les classes fournies.
2. Spécialisez la classe abstraite `batailleNavale.Joueur` en une classe `maBatailleNavale.JoueurRandom`, dans laquelle un coup est tiré aléatoirement et joué sur le plateau.
3. Définissez ensuite une classe `maBatailleNavale.BatailleNavaleTexte` dont l'unique méthode `main()` initialise les composants nécessaires, réalise la boucle principale du jeu, et affiche le gagnant à la fin de la partie.
4. Si le temps le permet, intéressez-vous à la définition d'un joueur humain.

