

DESCRIPTION DU MODÈLE :

- La classe **jeu** permet d'initialiser une partie par la méthode **initialiser**, qui créera un plateau avec 42 cases, ainsi que 2 joueurs (1 humain et une IA, ou 2 humains).
 - La méthode **affichageGraphique** est appelée à chaque tour pour mettre à jour l'affichage du plateau.
 - La méthode **determinerVictoire** est aussi appelée à chaque tour. Cette dernière analyse les cases du plateau pour déterminer si les conditions de victoire sont réunies (4 cases de la même couleur).
 - La méthode **afficherVainqueur** est appelée lorsque **determinerVictoire** est vraie. Elle met fin à la partie en affichant le nom du vainqueur.
- La classe **Plateau** contient les dimensions du Puissance 4. Il y a donc deux champs qui sont sa largeur et sa hauteur.
- La classe **Colonne** contient seulement une méthode.
 - La méthode **premiereCaseLibre** retourne la première case libre de la colonne
- La classe **Pion** contient un champ couleur (de type string).
- La classe **Case** contient une coordonnée (défini par une position en X et Y) et un statut pour indiquer si la case est déjà utilisée ou non.
 - La méthode **caseUtilisee** vérifie si la case est libre.
- La classe **Joueur** permet d'afficher le nom du joueur, le score, et le nombre de pions restants.
 - La méthode **renseignerNom** permet d'assigner un nom à un joueur ou l'IA.
 - La méthode **afficherNom** récupère le nom entré par le ou les joueurs et les affichera pendant la partie.
 - La méthode **compterPions** calcul le nombre de pions restant à placer.
 - La méthode **compterScore** calcul le nombre de victoire d'un joueur.
 - La méthode **placerPion** utilise un événement OnTouch, qui détermine la colonne dans laquelle placer le pion.
- Les classes **Humain** et **Ordinateur** héritent de la classe **Joueur**.

Déroulement d'un tour :

Le joueur “actif” choisit une colonne dans laquelle placer le pion. La liste des cases de la colonne est parcourue de bas en haut jusqu’à la dernière ligne du plateau. La première case avec un statut libre est choisie. On affecte à cette case la couleur du joueur appelant la méthode `placerPion`.

On appelle la méthode `affichageGraphique`, qui met à jour l’affichage du plateau.

On vérifie ensuite si les conditions de victoire sont présentes :

- on parcourt les colonnes une par une,
- si une case est occupée et de couleur particulière, la valeur de victoire est initialisée à 1, avec le nom du joueur propriétaire,
- si la case suivante est occupée et de la même couleur, on implémente encore la valeur de victoire, sinon celle-ci revient à 1, et si la case est libre, on passe la valeur à 0,
- si la valeur de victoire atteint 4, la méthode confirme la victoire et renvoie le joueur vainqueur,
- tant que la victoire n’est pas confirmée, on continue à parcourir les colonnes, puis on fait de même avec les lignes, puis avec les diagonales.

Si la victoire est confirmée, on lance l’affichage du vainqueur.

Avant de passer au prochain tour, on parcourt toutes les cases pour vérifier qu’il y en a au moins une de libre, sinon, la partie prend fin sans vainqueur.

Si la partie n’est pas finie, on change l’attribut “actif” des joueurs, pour que le second joueur joue à son tour, et on incrémente le tour du jeu.

Si la partie est terminée, on incrémente l’attribut score du gagnant s’il y en a un, puis on initialise une nouvelle partie.

Le joueur perdant commencera à jouer en premier sur cette nouvelle partie. Si égalité en fin de partie, la détermination du joueur qui commence en premier sera aléatoire.