

# *Puissance 4 - Consignes de Projet*

## *Objectif*

Créer un jeu de Puissance 4 en Python avec une interface graphique `tkinter` permettant l'interaction utilisateur et la détection de victoire.

## *Structure du Programme*

### 1. Création de la grille (`afficher_grille`)

**Objectif :** Représenter visuellement le plateau de jeu.

- Parcourt chaque case et affiche un cercle blanc sur un fond gris.
- Utilise un Canvas `tkinter` pour un affichage clair et structuré.
- Met à jour l'affichage après chaque coup pour refléter les nouvelles positions des jetons.

### 2. Gestion des jetons (`interagir_jeu via event.x`)

**Objectif :** Placer un jeton à l'endroit sélectionné par le joueur.

- Détecte le clic de souris et identifie la colonne sélectionnée.
- Parcourt la colonne de bas en haut pour trouver la première case vide.
- Place le jeton du joueur actuel et passe au joueur suivant.

### 3. Vérification de la victoire (`verifier_victoire`)

**Objectif :** Déterminer si un joueur a gagné.

- Recherche un alignement de 4 jetons dans toutes les directions :
  - Horizontalement (lignes)
  - Verticalement (colonnes)
  - Diagonales
- Peut être optimisée avec une sous-fonction pour éviter la duplication du code.
- Utilisation possible de tuples pour simplifier la vérification des combinaisons gagnantes.

### 4. Affichage du gagnant (`fenetre_congrats`)

**Objectif :** Informer du résultat et proposer une action.

- Affiche un message indiquant le vainqueur avec sa couleur.
- Ouvre une nouvelle fenêtre (`Toplevel`) pour signaler la victoire.
- Propose deux options :
  - Quitter le jeu (`Fermer`)
  - Relancer une partie (`Recommencer`) en réinitialisant la grille

## *Éléments `tkinter` utilisés*

- `Tk()`, `Canvas()`, `create_oval()`, `bind("<Button-1>", interagir_jeu)` pour l'interface.
- `Toplevel()`, `Label()`, `Button()` pour la fenêtre de victoire.