1. Classe GameEntity :

C'est une classe de base qui représente une entité de jeu générique. Toutes les entités de jeu, telles que la balle, la raquette et les briques, héritent de cette classe. Elle a des méthodes pour obtenir la position, déplacer l'entité et la supprimer du canevas.

2. Classe BouncingBall :

Cette classe hérite de GameEntity et représente la balle du jeu. La balle rebondit sur les côtés du canevas et peut entrer en collision avec d'autres entités. Elle a une méthode update qui ajuste la position de la balle en fonction de sa vitesse et de sa direction.

3. Classe ControlledObject :

C'est la classe représentant la raquette contrôlée par le joueur. Elle hérite également de GameEntity et a une méthode supplémentaire set\_ball qui permet de lier la raquette à la balle.

4. Classe ObstacleBrick :

Représente une brique dans le jeu. La couleur de la brique dépend du nombre de « hits » qu'elle peut prendre avant de disparaître. La méthode hit est appelée lorsque la brique est touchée, réduisant le nombre de hits restants.

5. Classe InteractiveGame :

C'est la classe principale qui gère l'ensemble du jeu. Elle hérite de tk.Frame et crée une interface graphique Tkinter. Elle initialise le canevas, la balle, la raquette et les briques. La méthode game\_loop est appelée de manière récursive pour mettre à jour le jeu à chaque itération.

6. Méthodes setup\_game, add\_ball, add\_brick, draw\_text, update\_lives\_text, start\_game, check\_collisions :

Ces méthodes sont utilisées pour initialiser le jeu, ajouter des éléments au canevas, afficher du texte, mettre à jour le texte des vies, démarrer le jeu, vérifier les collisions, etc.