OOP Group Project

Project Topic: Monopoly
Team: Gayane Miranyan, Liparit Igityan, Hrach Sargsyan

Board Class: This class represents the game board.

Fields: It has a field step to indicate the current position on the board.

Player Class: This class represents a player in the game.

Fields: It has fields such as moneyAmount to keep track of the player's money, step to denote the current position on the board, round to track the number of rounds played, and name to identify the player.

Methods: Methods include increaseMoney() and decreaseMoney() for managing money, throwDice() to roll the dice, and move() to update the player's position on the board.

Dice Class: This class represents a dice used in the game.

Fields: It has a field value to store the current roll value of the dice.

Bank Class: This class simulates the bank in the game.

Fields: It keeps track of the total moneyAmount held by the bank.

Methods: Provides methods like giveMoney() and receiveMoney() to facilitate transactions between players and the bank.

Property Class: This is an abstract class representing properties on the game board.

Fields: It has fields like price to denote the cost of purchasing the property and mortgagePrice calculated as half of the property's price.

Subclasses: Subclasses include House and Hotel. House class has its own subclasses - BlueHouse, RedHouse, GreenHouse, and OrangeHouse.

CornerPosition Class: This class represents special corner positions on the game board.

Methods: It provides methods like payJailMoney() for paying bail, goToStart() to move a player to the start position, and goToJail() to send a player to jail.

Card Class: This is an abstract class representing cards in the game.

Subclasses: Subclasses are ChanceCard and CommunityChestCard cards.

Fields: It has a field cardTypes to store different types of cards.

Methods: It includes a method getRandomCard() to retrieve a random card from cardTypes.