**Description**

This Python script simulates a basic version of a Monopoly game, extending its gameplay with a few additional rules and a broader board setup. It incorporates essential Monopoly elements such as property purchasing, rent payment, community chest cards, and basic player movements determined by dice rolls. The script is designed for multiple players (2-8) and ends when only one player is not bankrupt, or a maximum number of rounds are reached.

Here are the main components and features of the script:

**1. Classes and Initialization:**

- `Player`: Stores each player's name, balance, properties, and board position.

- `Monopoly`: Sets up the game board, manages properties, and controls the flow of the game.

**2. Game Setup:**

- The board is defined with a mix of purchasable and non-purchasable spaces.

- Players are set up with their initial balances and names entered interactively.

- Community chest cards are prepared and shuffled.

**3. Gameplay Functions:**

- Dice are rolled to determine movement.

- Players can land on different types of spaces, triggering specific actions like drawing cards, paying taxes, or deciding whether to purchase properties.

- Rent is paid to property owners, and properties are bought from the bank if unowned and the player affords it.

- Community Chest and potential Chance card effects are applied immediately to the players.

**4. End Game Conditions:**

- The game continues until one player remains solvent or a pre-set limit of 100 rounds is reached.

- Outputs inform players of each turn's outcomes and overall game status, including bankruptcy or the decision not to purchase a property.

**5. Interactive Inputs:**

- The game relies heavily on user inputs for the number of players, player names, and decisions to buy properties during the game.

This script would make a solid foundation for a more complex Monopoly simulation, where you could add features like Chance cards, more specific property management, and detailed player strategies. If you have specific functionalities or rules you’d like to integrate, I can help modify or expand the script accordingly!