Snakes and Ladders Documentation

Java/Spring Boot/Maven Backend

maciel.frank.snakesladders.Controller.GameBoardController.java

Description
This defines the endpoints in which our front end will interact with

Method	Description
loadGameData()	This will load up the initial board
advanceGameData()	Advances in the game and performs a check if the game is over
playerPositions()	Returns player positions to the front end as a string containing an array
playerPosition[1-4]()	Returns the integer position of the respective (1-4) player to the front end
rollNumbers()	Returns the numbers rolled to the front end
winnerExists()	Returns the integer value of the player who won (or 0 if none)
currentPlayer()	Returns the integer value of the current player
endGame()	Sets the board to null
error()	Returns an error

maciel.frank.snakesladders.Links.SnakesLaddersLinks.java

Description
This contains constant strings which relate to the links that exist in our endpoint

maciel.frank.snakesladders.Model.DieObject.java

Description
A Die object with a roll number and a number of sides. Used for the dice in the game. Contains a method to roll the dice internally as well as return the number rolled.

maciel.frank.snakesladders.Model.GameBoardObject.java

Description
The main object for the application. Contains the current player, a player list of PlayerObjects, two dice, a snake list of SnakeObjects, ladder list of LadderObjects, a dice sum as well as a winner.

maciel.frank.snakesladders.Model.LadderObject.java

Description
A ladder for the game contains a top and a bottom.

maciel.frank.snakesladders.Model.SnakeObject.java

Description
A snake for the game contains a head and a tail

maciel.frank.snakesladders.Model.PlayerObject.java

Description
A player for the game which contains a playerID and currentPosition, as well as methods to increment and decrement the player position.

maciel.frank.snakesladders.Service.Abstract.GameBoardService.java

Description
An abstract class which defines the important methods for the implementation of the game board service

maciel.frank.snakesladders.GameBoardServiceImplementation.java

Description
Performs the “business” logic on the game board object, and is used in the GameBoardController in order to simplify and modularize data being throughput

Method	Description
getGameBoard()	Returns the game board at its current state
diceRoll()	Rolls the two dice and returns their sum
playerPositions()	Iterates through playerList and adds their positions to an array; returns this array.
advanceInGame()	Calls iterateAdvance() & proceeds to check if a double was rolled; if so it remains on the current player, if not it moves to the next player
iterateAdvance()	Rolls the dice, increments the current player’s position by the dice sum, checks if the player goes over position 100 and backtracks, checks if a player hits a ladder bottom or a snake head.
checkLadder()	Iterates over the ladder list and determines if the current position matches any ladder bottom, if so return the ladder top, if not return -1
checkSnake()	Iterates over the snake list and determines in the current position matches any snake head, if so return the snake bottom, if not return -1
setGameBoard()	Used to end the game and set the board to null; it sets the board.
getRollNumbers()	Fetches the two numbers rolled, and returns them as an array
checkIfGameOver()	Checks if there was a winner, and sets this winner in the game board object
setSnakeList()	Sets up our snake list with the provided game board data
setLadderList()	Sets up our ladder list with the provided game board data