Authors: Michael Berg and Jennifer Phan

Highlighted in Yellow : Old Code (Before)

Highlighted in Green : Refactored Code (After)

|  |
| --- |
| public void initialize(){ |
|  | //initialize variables |
|  | adjMatrix = new HashMap<BoardCell, Set<BoardCell>>(); |
|  | legend= new HashMap<Character, String>(); |
|  | board = new BoardCell[MAX\_BOARD\_SIZE][MAX\_BOARD\_SIZE]; |
|  | visited = new HashSet<>(); |
|  | targets = new HashSet<>(); |
|  |  |
|  | int currentRow = 0; |
|  | int currentCol = 0; |
|  | try { |
|  | loadRoomConfig(); |
|  | Scanner scanner = new Scanner(new File(boardConfigFile)); |
|  | while (scanner.hasNextLine()) { |
|  | String nextLine = scanner.nextLine(); |
|  | Scanner rowScanner = new Scanner(nextLine); |
|  | rowScanner.useDelimiter(","); |
|  | while (rowScanner.hasNext()) { |
|  | if(rowScanner.hasNextInt()) { |
|  | break; |
|  | } |
|  | String item = rowScanner.next(); |
|  |  |
|  | //doorways |
|  | if(item.length() == 2) { |
|  | if(item.endsWith("U")) { |
|  | board[currentRow][currentCol] = new BoardCell(item, DoorDirection.UP); |
|  | } |
|  | if(item.endsWith("D")) { |
|  | board[currentRow][currentCol] = new BoardCell(item, DoorDirection.DOWN); |
|  | } |
|  | if(item.endsWith("L")) { |
|  | board[currentRow][currentCol] = new BoardCell(item, DoorDirection.LEFT); |
|  | } |
|  | if(item.endsWith("R")) { |
|  | board[currentRow][currentCol] = new BoardCell(item, DoorDirection.RIGHT); |
|  | } |
|  | if(item.endsWith("N")) { |
|  | board[currentRow][currentCol] = new BoardCell(item, DoorDirection.NONE); |
|  | } |
|  | } |
|  | else { |
|  | board[currentRow][currentCol] = new BoardCell(item, DoorDirection.NONE); |
|  | } |
|  | currentCol++; |
|  | } |
|  | currentRow++; |
|  | if(currentCol > numColumns) {numColumns = currentCol;} |
|  | if(currentCol == 0) { |
|  | currentRow--; |
|  | } |
|  | currentCol = 0; |
|  | rowScanner.close(); |
|  | } |
|  | numRows = currentRow; |
|  |  |
|  |  |
|  | scanner.close(); |
|  | } catch (FileNotFoundException e) { |
|  | System.out.println("File not found"); |
|  | } |
|  | calcAdjacencies(); |
|  | } |

|  |
| --- |
| public void initialize(){ |
|  | //initialize variables |
|  | adjMatrix = new HashMap<BoardCell, Set<BoardCell>>(); |
|  |  |
|  | visited = new HashSet<>(); |
|  | targets = new HashSet<>(); |
|  |  |
|  | numRows = 0; |
|  | numColumns = 0; |
|  | try { |
|  | loadRoomConfig(); |
|  | loadBoardConfig(); |
|  | } catch (FileNotFoundException e) { |
|  | System.out.println("File not found"); |
|  | e.printStackTrace(); |
|  | } catch (BadConfigFormatException e) { |
|  | System.out.println(e.getMessage()); |
|  | e.printStackTrace(); |
|  | } |
|  | calcAdjacencies(); |
|  | } |