SudokuMaker.java

Objective: To create a Sudoku puzzle using two-dimensional arrays and recursion.

Background:
Sudoku is a number puzzle using a 9-by-9 grid. The game has been around since the 19th century, but only caught on in a big way after the year 2000. It has become so popular, it shows up daily in newspapers and online publications, and there are numerous apps available to play the game. A completed puzzle is shown below.

4	**2**	**3**	6	**9**	7	**8**	1	5
6	9	**1**	5	**3**	**8**	4	7	2
5	8	7	**4**	2	1	6	3	9
3	1	9	8	7	5	**2**	6	**4**
2	5	**6**	1	4	9	**3**	8	7
7	4	**8**	3	6	2	5	9	**1**
9	6	4	2	1	**3**	7	5	**8**
1	3	5	**7**	**8**	4	**9**	2	**6**
8	7	**2**	9	**5**	6	**1**	**4**	3

The objective is to fill the grid with the digits 1 through 9 in every column and row without repetition, as well as each internal 3-by-3 grid without repetition. The 3-by-3 grids are denoted with dark lines. There are approximately 5.47 x 10^9 unique solution grids.

Algorithm:
You will implement a "brute-force" method for creating a Sudoku solution that uses recursion and backtracking. The idea is to start in the upper-left corner (0,0) and fill in numbers 1 to 9 at random in a row-major order. A number works in the location if it is not duplicated in the row, in the column, and in the 3-by-3 grid. If the number is not duplicated, then the routine recursively calls itself on the next grid location. When a number is duplicated, the program tries other numbers in random order in that grid location. If all the numbers 1 through 9 fail, then it discards this partial solution and recursively rolls back to a previous grid location to try another number. Eventually, the program will fill the grid, row by row, until it reaches the lower right corner and terminates with a solution.

You will use a two-dimensional array of **int**'s for the **puzzle**. Initialize the **puzzle** to 0's to represent the grid locations which do not have a solution in them.

Assignment:

1. Download **SudokuMaker.zip** file from Mr Greenstein's web site and unzip. It will create a **SudokuMaker** directory to do your work. In the directory you will find the **SudokuMaker.java** starter program. The program contains the `printPuzzle()` method.

2. Use the algorithm to write the `createPuzzle()` method.

When completed, the program should print out a solution grid. Every run of the program should produce a different result. Several sample runs are below.

```
% java SudokuMaker

Sudoku puzzle

 +-----------+-----------+-----------+
 |  5  2  7  |  1  6  3  |  9  8  4  |
 |  6  9  8  |  2  4  7  |  3  5  1  |
 |  4  1  3  |  9  5  8  |  2  6  7  |
 +-----------+-----------+-----------+
 |  7  4  6  |  8  3  2  |  5  1  9  |
 |  3  5  2  |  6  1  9  |  7  4  8  |
 |  1  8  9  |  4  7  5  |  6  2  3  |
 +-----------+-----------+-----------+
 |  2  3  4  |  5  9  1  |  8  7  6  |
 |  8  7  1  |  3  2  6  |  4  9  5  |
 |  9  6  5  |  7  8  4  |  1  3  2  |
 +-----------+-----------+-----------+


% java SudokuMaker

Sudoku puzzle

 +-----------+-----------+-----------+
 |  6  8  2  |  7  5  9  |  4  3  1  |
 |  5  3  1  |  6  8  4  |  2  9  7  |
 |  7  4  9  |  1  2  3  |  5  8  6  |
 +-----------+-----------+-----------+
 |  3  1  7  |  5  4  8  |  6  2  9  |
 |  2  9  6  |  3  7  1  |  8  5  4  |
 |  4  5  8  |  9  6  2  |  7  1  3  |
 +-----------+-----------+-----------+
 |  1  6  4  |  2  9  5  |  3  7  8  |
 |  9  7  5  |  8  3  6  |  1  4  2  |
 |  8  2  3  |  4  1  7  |  9  6  5  |
 +-----------+-----------+-----------+


% java SudokuMaker

Sudoku puzzle

 +-----------+-----------+-----------+
 |  5  7  1  |  3  4  9  |  6  8  2  |
 |  8  3  2  |  7  6  5  |  1  9  4  |
 |  4  6  9  |  8  1  2  |  5  7  3  |
 +-----------+-----------+-----------+
 |  3  1  4  |  5  9  8  |  2  6  7  |
 |  9  8  7  |  1  2  6  |  3  4  5  |
 |  2  5  6  |  4  7  3  |  8  1  9  |
 +-----------+-----------+-----------+
  | 6  4  5  | 2  8  7  | 9  3  1  |
  | 7  9  3  | 6  5  1  | 4  2  8  |
  | 1  2  8  | 9  3  4  | 7  5  6  |
  +-----------+-----------+-----------+


% java SudokuMaker

Sudoku puzzle

  +-----------+-----------+-----------+
  | 1  8  5  | 4  7  2  | 3  6  9  |
  | 7  4  6  | 5  3  9  | 1  2  8  |
  | 9  3  2  | 6  1  8  | 4  5  7  |
  +-----------+-----------+-----------+
  | 6  7  4  | 2  8  1  | 9  3  5  |
  | 5  2  8  | 3  9  4  | 6  7  1  |
  | 3  9  1  | 7  6  5  | 8  4  2  |
  +-----------+-----------+-----------+
  | 2  1  7  | 8  4  6  | 5  9  3  |
  | 4  5  9  | 1  2  3  | 7  8  6  |
  | 8  6  3  | 9  5  7  | 2  1  4  |
  +-----------+-----------+-----------+
```