Object Oriented Programing

Project Documentation

“SUDOKU”

Miguel Marines

INTRODUCTION

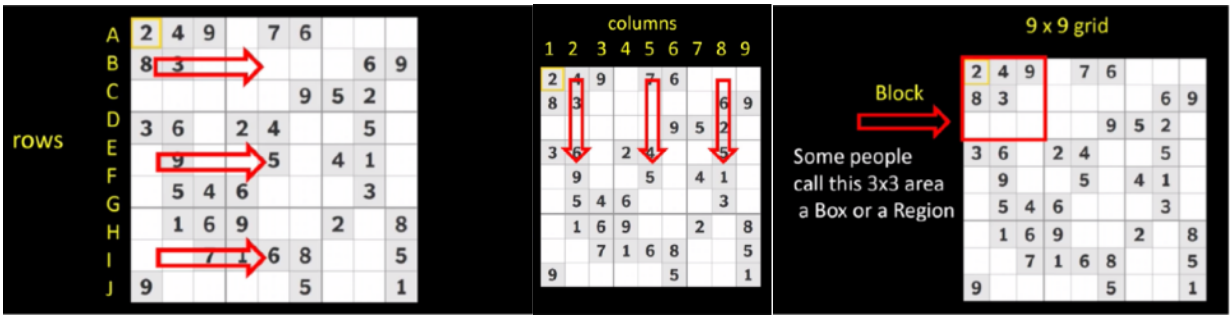
The final project is going to be a logic game called sudoku; sudoku is a placement puzzle that can be played with colors, letters or figures, but for this final project the sudoku is going to be composed of numbers.

The sudoku game exercises the brain extensively, and just like physical exercises, sudoku has benefits like: improving the memory, increasing the concentration, reducing the chances of developing Alzheimer, developing the ability of doing things quickly, reducing stress, helping to develop logical thinking patterns, helping to reduce depression and improving sharpness and strategy while approaching and solving problems.

In addition to the different types of benefits that sudoku has, I will develop this sudoku game in a Java program, because it will allow me to implement the different concepts, knowledge and skills, that I have learned and developed a long the object oriented programming course.

DESCRIPTION OF THE GAME

A Sudoku puzzle consists of 81 cells which are divided into 9 rows, 9 columns and 9 blocks.



The task is to place the numbers from 1 to 9 into the empty cells in such a way that in every row, column and block each number appears only once. An important characteristic is that, Sudoku has at least 17 given numbers but normally there are 22 to 30.

RELEVANT CHARACTERISTICS OF THE GAME

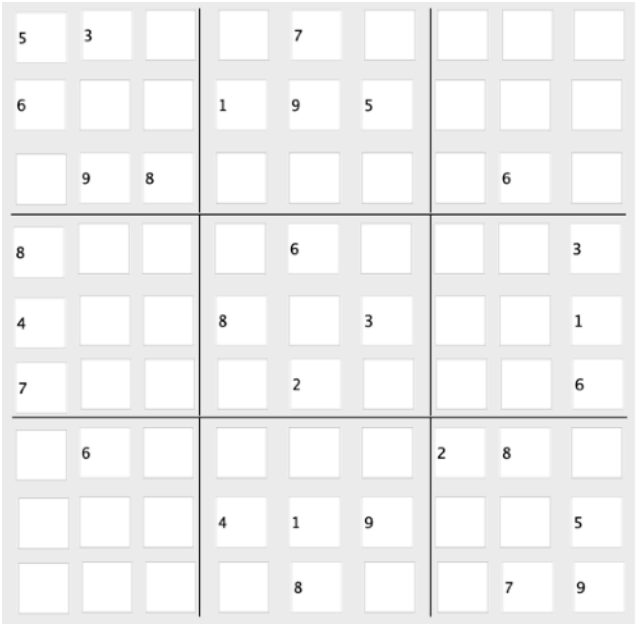
The Sudoku game is going to be composed of at list two packages: TheGraphical Interface and The Game Structure and Logic.

Graphical Interface:

The graphical interface are all the game elements, that the user sees and interacts with.

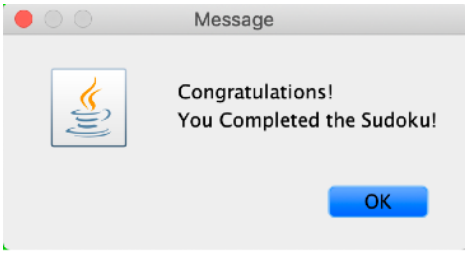
Board: The board has a fixed number of 9 columns, 9 rows, 9 blocks and 81 cells.

Numbers: The board has 30 given numbers, always in a fixed location.



Button: The sudoku has a button that enables the user to check if the numbers entered in the sells are the correct ones.

Message: A message will appear on the screen, when the sudoku is finished correctly.



Cells: The sudoku has 81 cells. When the user presses the button check answers the cells will turn green, (if the cell has the correct number), or red, (if the cell has the wrong number).

The user can click on a cell, with the mouse, and write or delete only numbers from 1 to 9, with the exception of the given numbers that the user cannot modify and are blocked by the program.

The Game Structure and Logic:

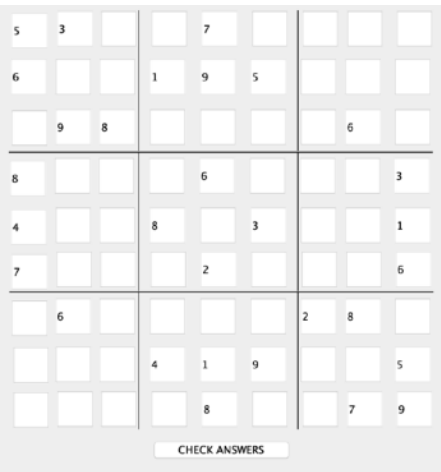
The game structure and logic is the analysis and implementation of the algorithms of the sudoku and its implementation into a java program.

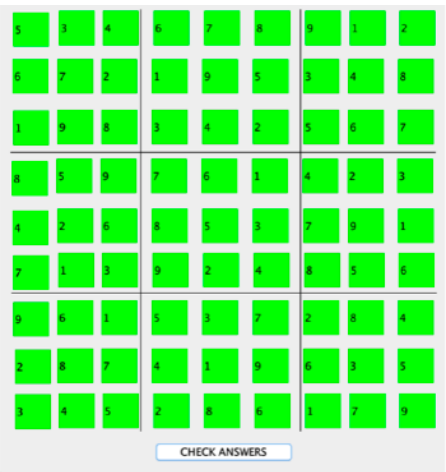
USER INTERACTION WITH THE GAME

The user will interact with the game, by running the program in his or her computer, then the user will fill each of the empty cells with the missing numbers; when the user has finished filling the boxes with the numbers, he or she will give the instruction to the program, by clicking on the check answers button, to check if the sudoku has been finished correctly or wrongly; if the sudoku has been finished correctly a message will apear on the screen (Congratulations! You Completed the Sudoku!), but if the sudoku has been finished wrongly the cells that have the wrong numbers will turn red and the cells that have the right numbers will turn green.

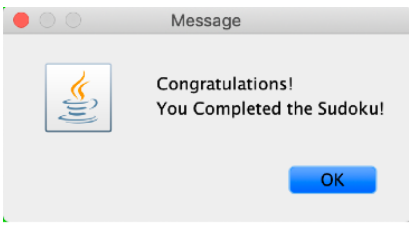
MOCKUPS

Boards:





Message:



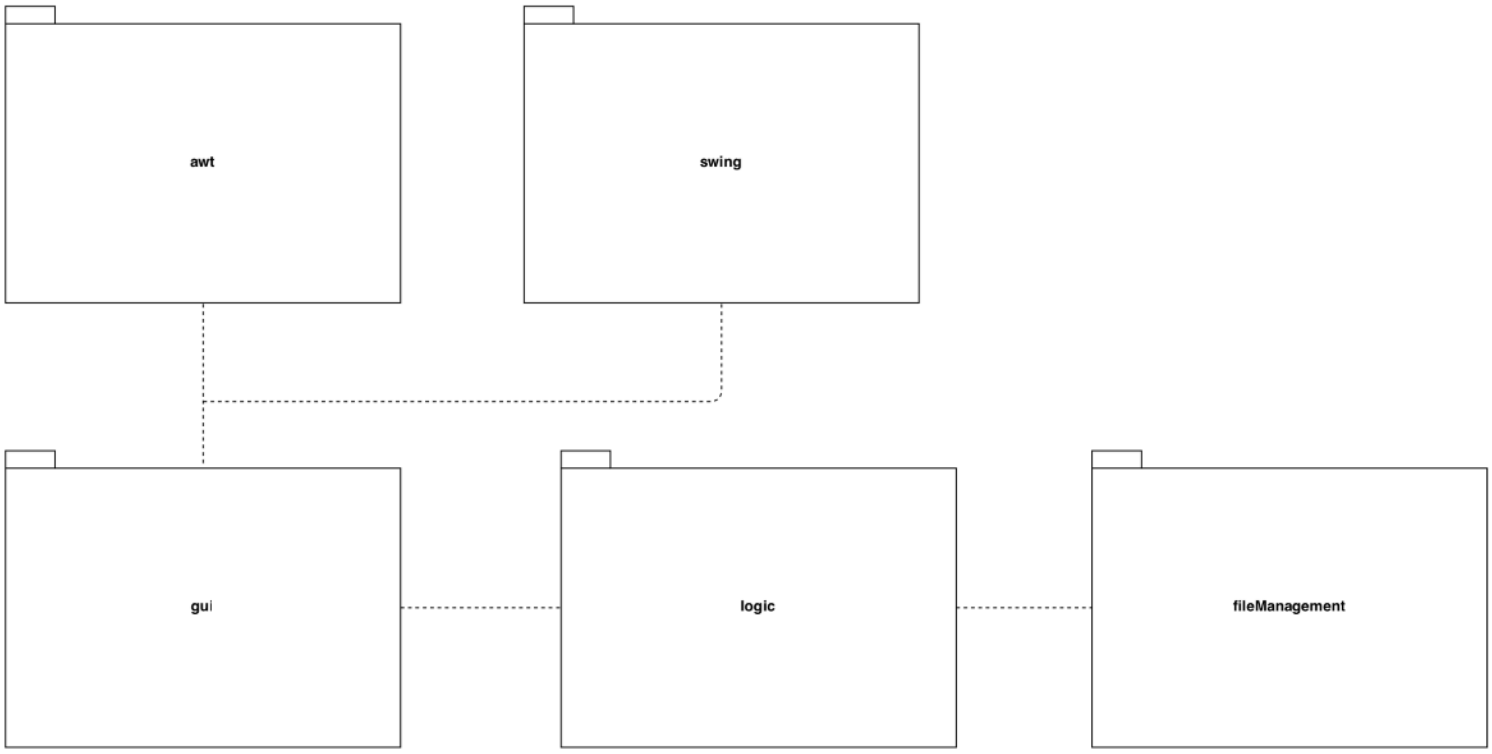
POJECT DEVELOPMENT

Time Log:

Project	Sudoku Final Project				Language	Java
Date	**Start**	**Stop**	**Break time**	**Actual time**	**Phase**	**Comment**
11/10/19	2:00 p.m.	2:30 p.m.	0	30 min	Read the project specifications	Read the project specifications
13/10/19	4:00 p.m.	5:00 p.m.	0	1 hour	Research about the game	Learn how the sudoku is played
15/10/19	2:00 p.m.	3:00 p.m.	0	1 hour	Project Proposal	Identification of features
21/10/19	3:00 p.m.	3:30 p.m.	0	30 min	Analysis of the logic	It was hard to understand the logic
23/10/19	7:00 a.m.	8:00 a.m.		1 hour	Redesign of the logic	It was hard to understand the logic
12/11/19	7:00 a.m.	8:30 a.m.	0	1:30 hours	Instalation of the programs	Eclipse worked well, but the window builder didn't
13/11/19	6:00 a.m.	8:00 a.m.	0	2:00 hours	Code	Start working on the GUI
19/11/19	7:00 a.m.	8:30 a.m.	0	1:30 hours	Code	Finished the GUI
20/11/19	7:00 a.m.	8:30 p.m.	0	1:30 hours	Code	Start working on the Logic
20/11/19	5:00 p.m.	5:30 p.m.	0	30 min	Diagrams	Start thinking on the diagrams
21/11/19	3:00 p.m.	4:00 p.m.	0	1 hour	Code	Keep working on the Logic
22/11/19	7:00 a.m.	8:30 p.m.	0	1:30 hours	Code	Keep working on the Logic
22/11/19	4:00 a.m.	7:00 p.m.	0	3:00 hours	Code	Had to redesigned the Logic to make the program work
22/11/19	9:00 a.m.	10:30 p.m.	0	1:30 hours	DIagrams	Finished the diagrams, had some doubts on the GUI part
30/11/19	4:00 a.m.	6:30 p.m.	0	2:30 hours	Code	Finish the code details
01/12/19	5:00 a.m.	8:30 p.m.	0	3:30 hours	Diagrams and Documentation	Finish all the details
					Total	24 hours

Diagrams:

1. Package Diagram



The awt package is the main package of the AWT, (Abstract Windowing Toolkit). It contains classes for graphics, including the Java 2D graphics capabilities, which helps to define the basic graphical user interface (GUI) framework in Java.

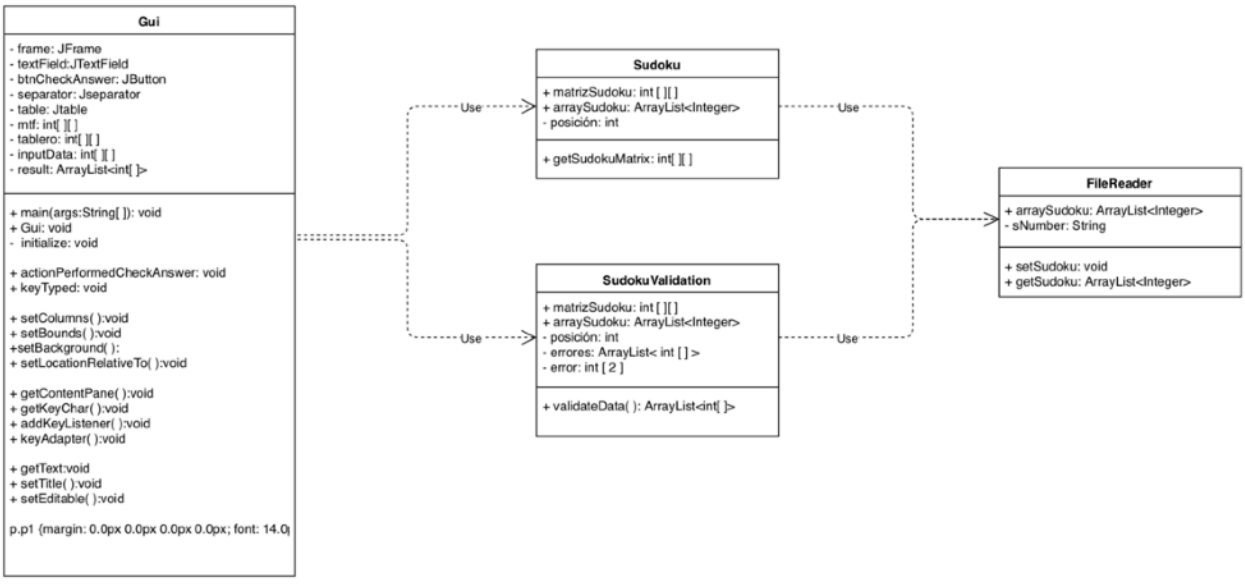
The swing package is used to create window-based applications. It is built on the top of AWT, (Abstract Windowing Toolkit), API and entirely written in java. Unlike AWT, Java Swing provides platform-independent and lightweight components.

The gui package is where all the JButtons, JTextFields, JTextArea, Jframes, etc; that use the awt and swing packages, are called to create the GUI, (Graphical User Interface). In this package we also have the main.

The logic package is the package that contains the classes that make the program work correctly in the areas that are not linked to de gui; In this package is where the game structure is implemented and also where the algorithms of the sudoku are implemented.

The fileManagement package contains the classes that enables the sudoku program to work with files. This package enables the sudoku program to grow and increase its complexity, due that with it, the addition of new sudoku boards would be easier or even to save the game and reopen it.

1. Class Diagrams



There are four classes in this sudoku program. The class Gui is in the gui package, the classes Sudoku and SudokuValidation are in the logic package and the FileReader class is in the fileManagement package.

In the Gui class is where all the buttons, textfields, frames, all the graphical user interfaces, etc are constructed. In this class is where the button check answers verifies if the inputs of the user are correct or not and sets the background color of the cells depending on it. This class is also very important because the main is in this class.

The Sudoku class is in the logic package and it creates an object of the FileReader class in order to read the sudoku from a file. This class is important because it gets the sudoku numbers from a file and stores them in a matrix that is more useful. The created matrix is very useful in the gui class to get and set the 30 determined given numbers of the sudoku and to validate the inputs of the user.

The SudokuValidation class is in the logic package and it creates an object of the FileReader class in order to read the sudoku from a file. This class is very important because it validates the inputs from the user and has a method, “validateData”, that returns the errors that the user has made. This class is very useful because it helps in the gui class to determine when the user has completed successfully the sudoku and to deploy a congratulations message when the sudoku has been completed successfully by the user.

The FileReader class is in the fileManagement package. This class is important because it has two methods that read the sudoku file and store the numbers from the file in an array to further use.

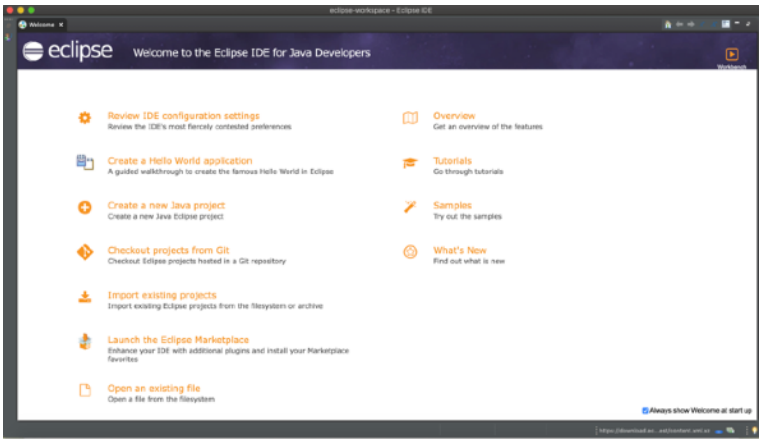
Code:

Attached Files

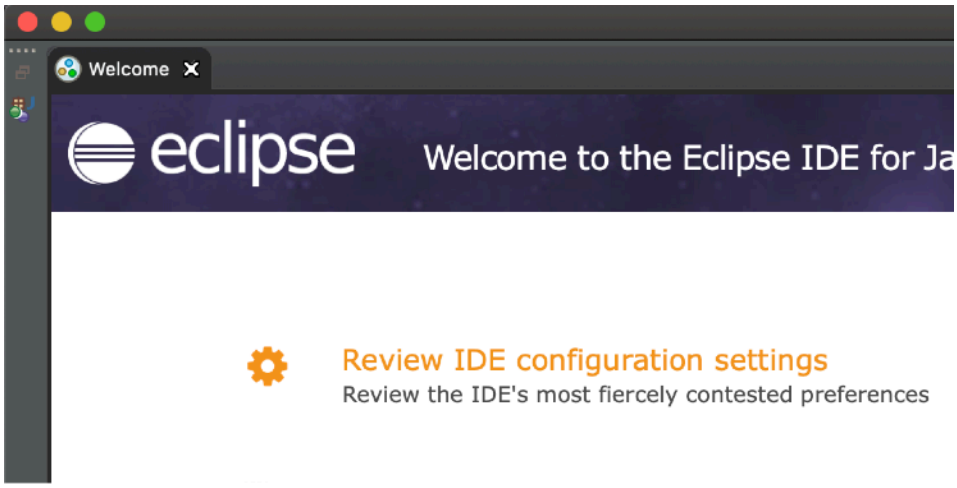
Compilation:

For this particular project, we were aloud to use eclipse, so in order to run the sudoku program, you need to:

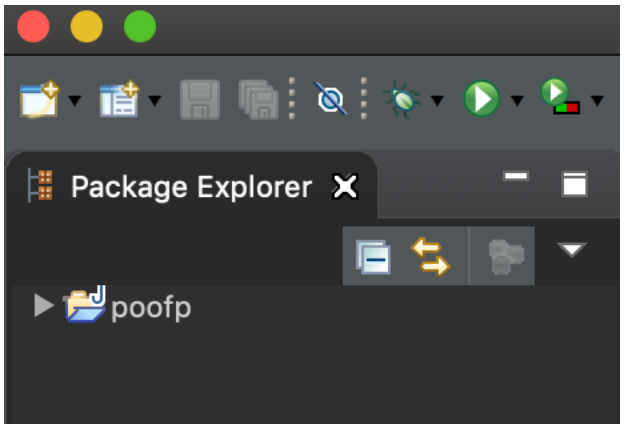
1. Open Eclipse



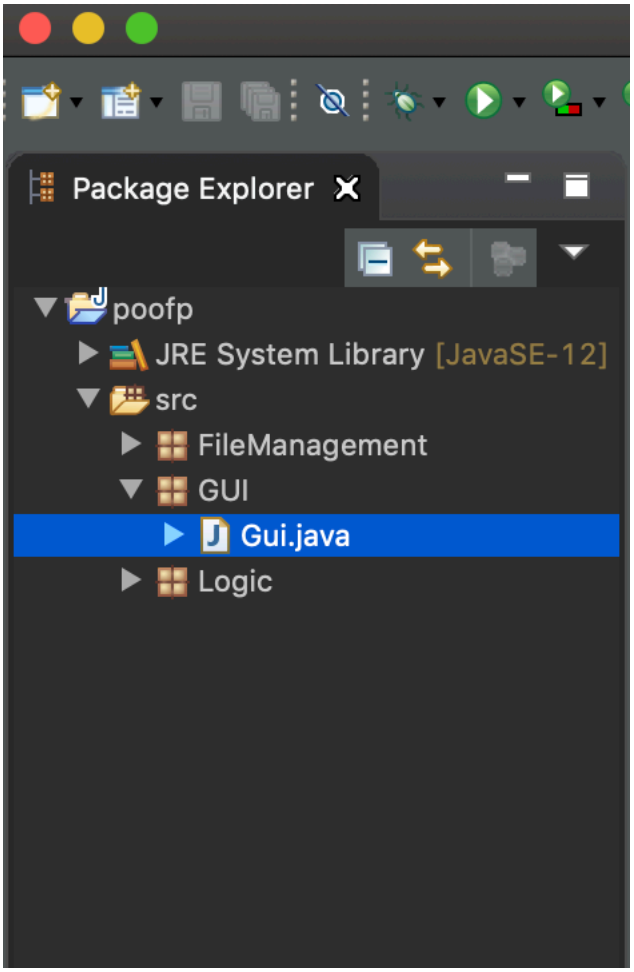
2. Close the welcome page, by clicking on the cross on the left upper corner.



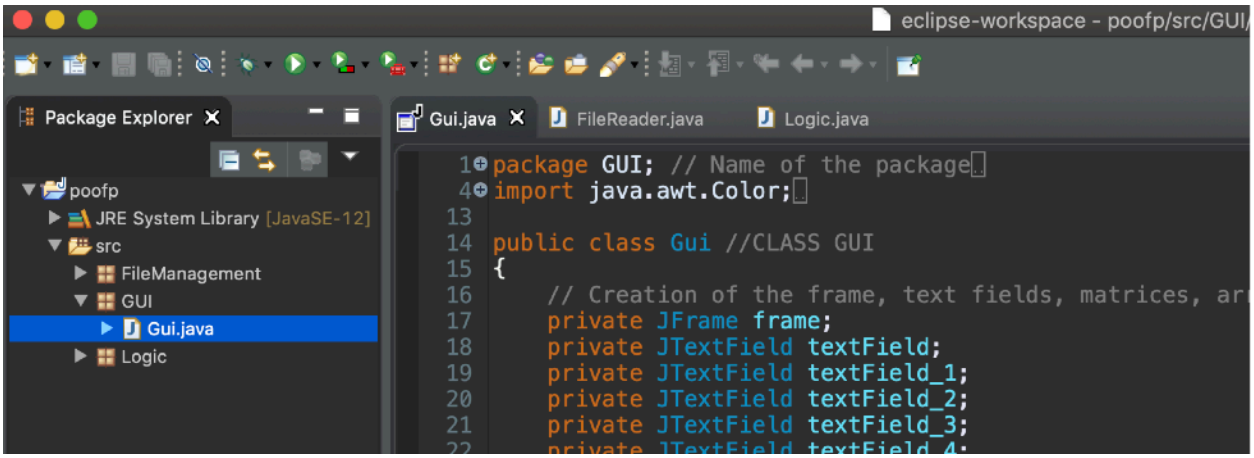
3. Select the project on the left side of the screen, in this case the name is “poofp”



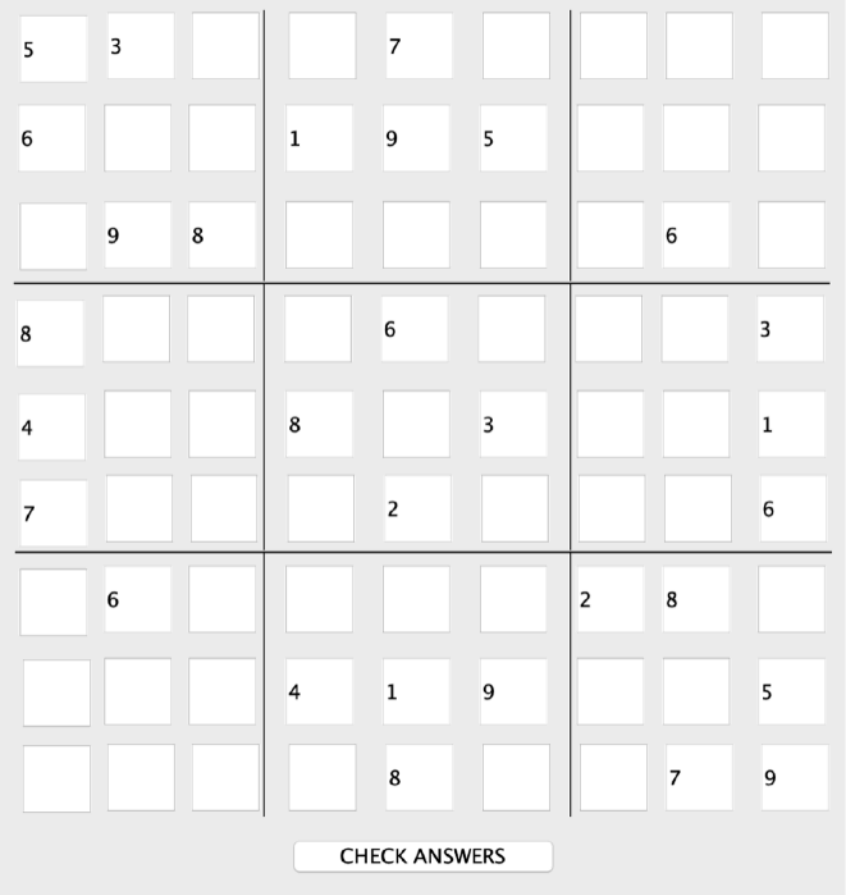
4. Then you need to go to the Gui class, also on the left side of the screen.



5. Then you need to press the biggest green button with a white arrow on it, on the upper part of the screen.



6. The program is open, now is time to play!



Solution of the Sudoku

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