This program is a game of guessing a random number between 1 and 100. It is a looped program using Boolean true and false so the user can continuously guess until they are correct. The main class, GuessingGame, is where the user will start the program. In the main class, Scanner is used to get the users input from the main class to use in its subclass, GuessRules.

GuessRules is where most of the games code is in. In GuessRules, Random generates a random number 1-100 for the user to guess. The number will be random every time the program runs. Boolean is used to loop the program using a while with true and false. Using false, the code will run once the program starts, asking the user for an input number to guess, starting the loop. To keep the guessed number within bounds of 1-100, there is an if-else for numbers over 100 and under 1 to print “Your guess is not within range” giving the user a chance to guess again. If-else and else if is used for if the guess is higher or lower than the random number. Giving the output “Higher” or “Lower” as well as the input for the guess that is on loop. This is also when the number of tries is counted based on how many inputs it takes till the user guesses correctly. If the guess is correct, the true will run, printing the output “You got it! The number was (the random number)”.

There is also a number of tries. The number of tries starts at 0 when the program starts and counts every entry the user inputs until they guess the right number. At the end of the program, along with “You got it! The number was \_\_\_”, There’s a total for the number of guesses it took the user until they guessed the number correctly.

In order to start the program, Run the main in GuessingGame.java. presented with "Guess the number 1-100:" This is where the loop startes using a boolean of truth and false. Based on the users input number, the program will print either "Higher", meaning the user needs to guess a higher number, or "Lower", meaning the user needs to guess a lower number, or (if the number is guessed correctly) the program will print "You got it! The number was (answer). There will also be an output of "It took you (# of tries) tries." printing the total number of tries it took till the user guessed the number correctly. That is the end of the program, to play again, the user just has to restart the program at main.

Graphical user interface, application, Word

Description automatically generated