# Part 1:

First the user is asked if they would like through the prompt of input being between -1 to 5. 1 being the ability to add to the knowledge base; 2 to clear the current knowledge base; 3 to display the current knowledge base; 4 to enter a query; 5 to run the query that has been entered on the knowledge base and finally, -1 will exit out of part one and return to the main menu.

If 1 is chosen, the user will be prompted to enter the data into the knowledge base. This data is to be structured in [Literal1,Literal2,!Literal3] with the number of literals being able to be any number greater than one. The ‘¬’ symbol was replaced with the ‘!’ due to issues in Java in processing the symbol. The program gets the user input via starting an infinite while loop, that can be exited by inputting -1. If the input is not -1 then the program will create a new instance of the Clause object with the input that the user entered being the only parameter. The Clause object first takes the data and strips the surrounding square brackets from the input. It then splits up the string using the regex of ‘,’ it then loops through these new strings creating new Literal objects. It checks to see if the string contains the ‘!’ symbol if it does then the symbol is removed. When the Literal is made it takes two parameters, the string of the name of the literal and a boolean. The boolean marks if the literal is false. It then adds these new literals to a list called “listOfLiterals” which can be retrieved from the object later. This will continue until the user enters -1 to go back to the menu to choose another action.

If 2 is entered, then the knowledge base stored in the local space is set to null and the clearKnowledgeBase method in the InputManager class is called. This sets the knowledge base that is also stored in this class to null then creates a new empty ArrayList.

If 3 is inputted, then first it checks if the knowledge base is not null. If it is not, then it will start a for loop of the length of the knowledge base and output each element in the knowledge base. If the knowledge base is null, then it will do nothing.

4 prompts the user to input the query in the negative form and stores it in the Input Manager.

5 first makes sure the knowledge base is not null, the calls the method to run the query in the Query class. It then checks if the boolean returned by this function is true or false, if its true it will output ‘SOLVED’ otherwise it will output ‘NOT SOLVED’