Object-Oriented Python Tic Tac Toe

Implement the game of Tic Tac Toe using object-oriented Python (OOP). The program should be written as two Python files. One will be the the class implementation of the game of Tic Tac Toe. The other will be used to instantiate an instance of the class and call the class methods to play the game. Both programs should be written according to PEP8 standards. There should be no warnings when using pycodestyle on the Python files created. The programs should use version 3.8 of Python.

Features that should be included with game play in the class:

1. The class should contain all components required to play a full game of Tic Tac Toe
2. The code should be as DRY (don't repeat yourself) as possible
3. Use docstrings to document the purpose of the class and any modules created
4. The __init__ constructor should provide default values to set the dynamic values for game play (i.e. player names, who plays first, who plays what piece, number of players, etc.)
5. Game play allows for 1 or 2 players
6. If game play is set to 1 player, player 2 will be the "Computer"
7. Implement code to play as the computer
8. Store game records into a records file when game play is complete
9. Read the records file when the game play values have been set
10. Create a function that can be called to change all the game values
11. Display game records before the game board each time it is displayed
12. Display the game board using the numbers 1-9 as the position values to play
13. Prompt each player for the number of the position that they want to play
14. Check if the game is over after a player plays
15. When the game is over as a result of a tie, increment the draw value of each player
16. When the game is over as a result of a win, increment the win value of the player that won and the loss value of the player that lost
17. After a game is over, prompt the players to see if they want to play another game, if they do, the player's turn should switch to the next player before game play begins
18. Perform error checking on all values entered by the players, if a value entered is invalid, continue prompting the players until a valid input is received. If one cannot be obtained, a SystemExit exception should be raised

Features that should be included with the calling program:

1. Create a function that handles the signal for CTRL+C and exits with a message informing the user
2. Be able to handle a CTRL+D and exit with a message informing the user
3. Check to see if the program is being called as the main program
4. Catch the SystemExit error, if thrown
5. Import the class from the Python file created
6. Create an instantiation of the class created
7. Execute the class functions to create an instance of the game and execute the functions required to play