Tic Tac Toe

- There are two players for the game.
- The goal of the game is to get three noughts or crosses aligned within a 3x3 grid.
- This can occur horizontally, vertically or diagonally.
- If there no noughts or crosses aligned, the players have drawed and the game starts again.

Coding Structure

The Data structure we will use will be a 2D array, Lists
The constants are the 0 X as the are the only symbols within the game
The variables are the coordinates of the 0 and X
User can input a coordinate within the 3x3 grid to place their symbol

Variables	Datatype	Description
Grid Coordinates	integers	This variable allows the user to allocate their symbol in the 3x3 grid
Score	Integer	This will store the amount of rounds the user has won
Game Checker	Boolean/While loop	This will check who is the winner and loser of the game and if it is a draw
To continue playing or to stop (Yes/No)	String/boolean?	This will allow the user to decide whether they want to continue playing or to stop
Grid	Function	Whenever this function is called it will show a blank grid

Data Structure

Data Structure	Purpose
2D Arrays	This used to display the grid to the user in a neat format

User Interface

User interacting with the program	Messages received
Finishing the game	Win, lose or draw
Typing number to place character	X or 0 will appear on the place they clicked
Multiplayer or Singleplayer(Ai)	User plays game against chosen player

Test Plan

Version 1

Test Num	Description of Test	Test data	Expected outcome	Actual Outcome	Comments and fixes
1	The grid for noughts and crosses appear when called	Normal data Function called	When the player want to play or restart a game a blank grid should appear	The grid would be blank appear	None
2	The user will be able to type down the coordinates to place down the symbol	Normal Data (coordinates) (0,1) Abnormal Data (2,1) Extreme (2,2)	The symbol should appear in that area	The characters will not appear in the grid.	To make the user inputs an integer and split the coordinates so it can be interpreted as columns and rows
3	The user will have the option to continue playing or stop	Normal data Yes or No	If yes is chosen a blank grid should be displayed, if no the game would come to an end	The game would	To use boolean to allow the player to stop or continue

4	The counter will count the amount of rounds the players have won	Normal Data Counter increases by 1 Extreme Counter stop when game ends	The counter to increase by 1 every a round has been won by one of the players	The counter would increase	To create a counter variable for the noughts and crosses
5	Checking if there is a winner	Normal Data	The game should announce which player has won	Player has been announced as winner	None
6	Checking if there is a draw	Normal Data	The game should recognise when the board is full an announce that it is a draw	Noticed when the grid was full	None

Version 2

Test Num	Description of test	Test Data	Expected Outcome	Actual Outcome	Comments and fixes
1	The grid for noughts and crosses appear when called	Normal data Function called	When the player starts a game a blank grid should appear	The grid would be blank appear	None
2	The user will be able to type numbers to place down the character	Normal Data 2 Abnormal Data 5 Extreme 9	The symbol should appear in that area	The character did appear in the grid	None
4	Checking if there is a winner	Normal Data	The game should announce which player	Player has been announced as winner	None

			has won		
5	Checking if there is a draw	Normal Data	The game should recognise when the board is full an announce that it is a draw	The game had a runtime error when the grid was full	Try to say the space has been taken by a character
6	Game finished	Normal Data	The game should stop playing when a winner is announced or a tie	The game would continue asking for positions	Use a while loop to end the game

Version 3

Test Num	Description of Test	Test data	Expected Outcome	Actual Outcome	Comments and fixes
1	AI	Normal Data Import Random	The AI should select a number between 1,9 and a character should be place in that area	The character appeared in the grid	None
2	Checking if there is a winner	Normal Data	The game should announce which player has won	Player has been announced as winner	None
3	Checking if there is a draw	Normal Data	The game should recognise when the board is full an announce that it is a draw	The game would continues asking for positions when board is full	Try to say the space has been taken by a character

Evaluation

For my final version of the I had use functions to call my variables and make it easier for reading and interacting with the game. I had uses dictionaries to format my grid and allow users to select numbers to place their characters. The selections were used to identify winners and ties.
However, I had a runtime error with the tie as it was not producing the results I wanted when the board was full.
The program is easy to use as the format for placing the characters are numbers from 1 to nine. It provides a AI which selects random numbers from 1,9 for the player to play with. The user can crash the game if they were to place incorrect values for the code. To fix this problem I will need to use the try and accept.
The code is easy to read and self explanatory when read. The code should be easy to modify as it in a organised format and composed of functions which make it easier to read for a programmer.
The code is not robust as the user can crash the game by selecting incorrect values. I had used functions and a global variable to use my memory efficiently.
The code can be improved by fixing the runtime error with the draw. To make a smarter AI so it selects spots that are not random. To check if the space is taken by a character. The performance of my program was basic and simple