**Chapter 7 – Multidimensional Arrays**

Two dimensional arrays are used to store a matrix or table of information. We can also use them to represent items on a grid using (X, Y) coordinates. A two-dimensional array is a one-dimensional array in which each element is another one-dimensional array.

Let’s say we want to create an array to hold a Tic-Tac-Toe board. Obviously a Tic-Tac-Toe board has nine spots which would be handled with a one-dimensional array, but a two-dimensional array makes the organization of the data much easier to manage by utilizing an (X, Y) index to represent the different spots on the Tic-Tac-Toe board.

**One-Dimensional Array** **Two-Dimensional Array**

0 1 2 (0, 0) (1, 0) (2, 0)

3 4 5 (0, 1) (1, 1) (2, 1)

6 7 8 (0, 2) (1, 2) (2, 2)

**Declaring a One-Dimensional Array**

*char[ ] TicTacToe = new char[9];*

**Declaring a Two-Dimensional Array**

*char [ ] [ ] TicTacToe = new char [3] [3];*

**Initializing a Two-Dimensional Array**

We use nested for loops to initialize, process, or output two-dimensional arrays. Imagine in this situation, I want all spaces in the board to be set to spaces. This code will accomplish it:

*for (int y = 0; y <3; y++){*

*for(int x = 0; x < 3; x++){*

*TicTacToe [x] [y] = ‘ ‘;*

*}*

*}*

**Setting a value to an element in a Two-Dimensional Array**

Let’s say I want to place an X in the middle of my two-dimensional TicTacToe board: *TicTacToe [1] [1] = ‘X’;*

**Two-Dimensional Arrays and Methods**

The same rules apply for sending a Two-Dimensional Array to a method. If the array is changed inside the method, it is also changed outside the array. Suppose we wanted to send our 2D array to a method: *outputBoard(TicTacToe);*

*public static void outputBoard(char [] [] ttt){*

*for (int y = 0; y < 3; y++)*

*{*

*for (int x = 0; x < 3; x++)*

*{*

*System.out.print(ttt[x][y]);*

*}*

*System.out.println();*

*}}*