Lesson 15 - Collections II

Recall,

> C# has many more complex data types built-in that help us to store and manipulate data. One such example is the `Array` type. An `Array` stores collections of data of the **same type**. An example, let us suppose we want to store 5 number, without using arrays, you would something like this,

Multidimensional Arrays

`Arrays` can have multiple dimensions. a 2 dimensional array is declared as,

```
int[,] myArray = new int[3,4];
```

The resultant array can be represented in the form of a table as follows,

	Column 1	Column 2	Column 3	Column 4
Row 1	`[ 0 ] [ 0 ]`	`[ 0 ] [ 1 ]`	`[ 0 ] [ 2 ]`	`[ 0 ] [ 3 ]`
Row 2	`[ 1 ] [ 0 ]`	`[ 1 ] [ 1 ]`	`[ 1 ] [ 2 ]`	`[ 1 ] [ 3 ]`
Row 3	`[ 2 ] [ 0 ]`	`[ 2 ] [ 1 ]`	`[ 2 ] [ 2 ]`	`[ 2 ] [ 3 ]`

Think of a multidimensional array as a **matrix**.

```
int[,] multidimensionalArray = new int[3,3];

int[,] multidimensionalArray =
{
        { 1, 2, 3 },
        { 4, 5, 6 },
        { 7, 8, 9 }
}
```

Jagged Arrays

These are arrays of arrays. So an array not made of integers, or boolean values or strings, but an array made up of arrays.

```
int[][] jaggedArray = new int[3][];

int[][] jaggedArray =
{
        new int[] { 1, 2, 3, 7},
        new int[] { 4, 5 },
        new int[] { 7, 8 , 9 }
};
```

Think of these like separate rows that can have different number of items.

Nested Loops

Nested loops means loops inside a loop. They can be used to index multidimensional and jagged arrays.

Example: Multidimensional Arrays

```
int numStudents = 2;
int numCourses = 3;

int[,] classGrades = new int[numStudents, numCourses];

Console.WriteLine("\nTaking Input\n");

for (int i = 0; i < numStudents; i++)
{
  Console.WriteLine("\nGrades for student " + (i+1));

  for (int j = 0; j < numCourses; j++)
  {
        Console.Write($"Grade in course {j+1}: ");
        classGrades[i,j] = Convert.ToInt32(Console.ReadLine());
  }
}
```

Example: Jagged Arrays

```
int numStudents = 2;
int numCourses = 3;
```

```
int[][] classGrades = new int[numStudents][];

Console.WriteLine("\nTaking Input\n");

for (int i = 0; i < numStudents; i++)
{
    Console.WriteLine("\nGrades for student " + (i + 1));
    classGrades[i] = new int[numCourses];

    for (int j = 0; j < numCourses; j++)
    {
        Console.Write($"Grade in course {j + 1}: ");
        classGrades[i][j] = Convert.ToInt32(Console.ReadLine());
    }
}
```