1. Log in
2. Working in Unit 3
   - 1-Dimensional Arrays
   - 2-Dimensional Arrays
   - Array Lists

Oct 22-4:28 PM

Quick Review: The 1-D Array!

```
int[ ] arrayName = new int[n];
```

```
public static void main(String[] args)
{
    int[ ] myArray = new int[5];

    myArray[0] = 7;
    myArray[1] = 13;
    myArray[2] = 40;
    myArray[3] = 47;
    myArray[4] = 99;
}
```

Sep 12-7:31 PM

Remember there's a faster way for arrays too!!!

```
int[ ] betterArray = {7, 13, 40, 47, 99};
```

*** We may work with int, double, String, and boolean Arrays ***

With arrays ...

```
arrayName.length;          //array lengths are an int variable
```

Sep 12-7:31 PM

Two-Dimensional Arrays ...

Can be thought of as a table (rows/columns)

```
int myTable[ ][ ] = new int[3][4] ;        // row x column
                                           // still starts with [0][0]
```

myTable[0][0]	myTable[0][1]	myTable[0][2]	myTable[0][3]
		???	

Answer: myTable [2] [2]

Sep 16-8:19 AM

Create myTable example ...

Remember row x column

C0 C1 C2 C3

```
int[][] myTable = {   { 0 , 1 , 2 , 3 },      <-- row 0
                      { 4 , 5 , 6 , 7 },      <-- row 1
                      { 8 , 9 , 10 , 11 }   };  <-- row 2
```

```
System.out.println("Here's the number seven:  "+myTable[1][3]);

System.out.println(myTable[0][2]+myTable[2][3]);
```

Sep 16-8:11 AM

Learning how to loop through ...

```
int[][] myTable =  {   { 0 , 1 , 2 , 3 },
                       { 4 , 5 , 6 , 7 },
                       { 8 , 9 , 10 , 11 }   };

int sum=0
for(int i=0 ; i<myTable.length ; i++)
    sum+=myTable[i][0];
System.out.print("The sum of the numbers is: " + sum);
```

*** What just happened? ***

Sep 16-8:11 AM

What changes to add the "2nd row"?

```
int[][] myTable = {     { 0 , 1 , 2 , 3 },
                        { 4 , 5 , 6 , 7 },
                        { 8 , 9 , 10 , 11 }    };

int sum=0
for(int i=0 ; i<myTable.length ; i++)
    sum+=myTable[i][1];
System.out.print("The sum of the numbers is: " + sum);
```

*** What just happened? ***

Sep 16-8:11 AM

Remember the for-loop for 1-D Arrays ...

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

int sum=0
for(int i=0 ; i<myArray.length ; i++)
    myArray[i]=i+3;
System.out.print("The Array holds:  ");
for(int i=0 ; i<myArray.length ; i++)
    System.out.print(myArray[i]);
```

The for-loop nicely cycles through every element in the list (but the list is only one row!)

*** What just happened? ***

Sep 16-8:11 AM

Now for 2-D Arrays ...

```
int[][] myArray =  new int[3][4];
int sum=0;

for(int i=0 ; i<myArray.length ; i++)
   for(int j=0 ; j<myArray[i].length ; j++)
      myArray[i][j]=i+j;

System.out.print("The Array holds:  \n");

for(int i=0 ; i<myArray.length ; i++)
{
   for(int j=0 ; j<myArray[i].length ; j++)
      System.out.print(myArray[i][j]+"  ");
  System.out.println();
}
```

We should probably explore this in action ...

Sep 16-8:11 AM

Be aware ...

* for-loops very common with arrays
* 1-dimensional starts with index 0
* 2-dimensional starts with [0][0]

Sep 21-8:00 AM

Things to be working on ...

* Wrap up Unit 3 WS01 - 1D Arrays
* Get going on Unit 3 WS02 - 2D Arrays

Sep 21-8:00 AM