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

Oct 22-4:28 PM

Random Side Note to Add to the Day ...

Different types of programs and their names ...

Applications - Run on a computer ... typically what we do here

Applets - Run on web pages

Servlets - Run on servers

Apps - Run on mobile devices

Sep 3-2:50 PM

I would like to use a list of my favorite numbers in my program ...

```
public static void main(String[] args)
{
    int a = 7;
    int b = 13;
    int c = 40;
    int d = 47;
    int e = 99;
}
```

Remember, there's a faster way ...

Sep 12-7:31 PM

Faster way ...

```
int a = 7, b = 13, c = 40, d = 47, e = 99;
```

BUT... Now there's another new way to store a list of items ...

The Array!

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

Sep 12-7:31 PM

3 Ways to create an empty array ("table of data") ...

```
double [ ] arrayName = new double[10];

double arrayName[ ] = new double[10];

double[ ] arrayName;
arrayName = new double[10];
```

*** Each of the above examples creates an array of length 10 called arrayName ***
*** The array currently holds no data though ***
*** Once initialized, the array length is permanent and CAN'T be changed ***

Sep 12-7:31 PM

```
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;
}
```

Required! Tells computer you are making an array

How many things are in your list

What type of data will be in the array ... all items must be same data type

Name of your array (like variable name)

Index Number (starts with 0)

Sep 12-7:31 PM

Here's the code without the distractions ...

```
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;
}
```

new
1) new Scanner ...
2) new DecimalFormat ...
3) new "array" ...

These are all objects!
We will learn more later
in this course about objects!

Sep 12-7:31 PM

Hey ... there's a faster way for arrays too!!!

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

What would the following lines output??? (remember index #'s)

```
System.out.println(betterArray[3]);

System.out.println("My 2nd number is: "+betterArray[1]);
```

Sep 12-7:31 PM

Types of arrays that are possble ...

```
int[ ] betterArray = {7, 13, 40, 47, 99};
double[ ] doubleArray = {2.2, 3.3, 3.4};
String[ ] stringArray = {"me", "you", "them"};
char[ ] charArray = {'A', 'B', 'C', 'D', 'F'};
boolean[ ] boolArray = {true, false, true, true};
```

Special Note: Notice proper notation for character and boolean types!

Sep 12-7:31 PM

What do arrays hold if you don't fill them???

```
int[ ] betterArray = new int[5];
```
... reserves 5 slots for integers, they are all 0's to begin with.

```
double[ ] doubleArray = new double[7];
```
... reserves 7 slots for doubles, they are all 0's to begin with.

```
String[ ] stringArray = new String[3];
```
... reserves 3 slots for Strings, they are all *null* to begin with.

```
boolean[ ] boolArray = new boolean[4];
```
... reserves 4 slots for booleans, they are all false to begin with.

Sep 12-7:31 PM

For Loops are Common with Array "filling" ...

```
int[ ] evenArray = new int[5];
    for(int i=0 ; i<5 ; i++)
        evenArray[i]=2*i;
    for(int i=0 ; i<5 ; i++)
        System.out.println(evenArray[i]);
```

*** This very basic example fills an array and then prints the array elements ***

go to oddNumbers class in netbeans ...

Sep 15-11:15 AM

Special Note About Array Lengths ...

```
int[ ] thisArray = new int[3];      //array with 3 elements

thisArray[0]                        //index 0 is element 1
thisArray[1]                        //index 1 is element 2
thisArray[2]                        //index 2 is element 3
thisArray[3]                        //does not exist!!!
```

* If an array has 'n' elements, index #'s go 0 through 'n-1'

Oct 22-8:05 PM

If you aren't "in the array", you will get an error!

int[ ] arrayParty = new int[4];

All of the following throw an error!

arrayParty[7]
arrayParty[-3]
arrayParty[4]

Error: ArrayIndexOutOfBoundsException

Oct 22-8:05 PM

Finding length of an array ...

Remember With String ...

variableName.length(); //function that counts characters

With arrays ...

arrayName.length; //array lengths are a stored //integer variable

Sep 15-11:50 AM

What if we want to store the first 100 even numbers?

```
public class manyEvenNumbers {

    public static void main(String[] args)
    {
        int[ ] evensGalore = new int[100];

        for(int i=0 ; i<evensGalore.length ; i++)
        {
            evensGalore[i]=2*(i+1);
            System.out.println(evensGalore[i]);
        }
    }
}
```

System.out.println("The 34th even number is " + evensGalore[33];

Sep 15-12:56 PM

Additional comments about arrays ...

1. Arrays are technically objects, therefore object rules apply to arrays.
2. superCoolArray[21]
   ... the 21 is the "subscript' or index number
3. It is very common for a program to "loop through" an array to ...

   - Search for something
   - Sum up all the numbers
   - Count how many times something occurs
   - Find the first occurrence of something

Sep 3-9:31 PM

Arrays ... are they full???

```
double[ ] woohoo = new double[7];
woohoo[0] = 10;
woohoo[1] = 20;
woohoo[2] = 30;
```

Physical Size - Is 7 for this array (can hold 7)

Logical Size - Is 3 for this array (is holding 3)

System.out.println(woohoo[2]+" "+woohoo[3]);

Output: 30.0 0.0 //why?

Sep 3-9:31 PM

Parallel Arrays ...

This basically just means that you have two (or more) arrays that are connected in meaning:

```
String[ ] studentName = new String[3];
int[ ] testScore = new int[3];
int[ ] period = new int[3];

studentName[0] = "Betty";
studentName[1] = "Charlie";
studentName[2] = "Kathy";
testScore[0] = 93;
testScore[1] = 85;
testScore[2] = 68;
period[0] = 5;
period[1] = 3;
period[2] = 1;
```

The basic idea is that all items with the same index number "belong together";

Sep 16-8:18 AM

Create "ParallelArrayUserInput in Netbeans ...

Sep 24-11:49 AM

User input to make the parallel array "useful" ...

```
public class parallelArrayUserInput{
public static void main(String[ ] args){

int numStudents = 5;                                    //can use a variable to
                                                        //determine length of our array
String name[ ] = new String[numStudents];
int grade[ ] = new int[numStudents];

for ( int j = 0;  j < numStudents  ;  j++)
{
      Scanner kbReader1 = new Scanner(System.in);
      System.out.print("Enter the student name: ");
      name[j] = kbReader1.nextLine( ); //input from keyboard

      Scanner kbReader2 = new Scanner(System.in);
      System.out.print("Enter the grade: ");
      grade[j] = kbReader2.nextInt( );
}
}}
```

Sep 16-8:19 AM

Things to do ...

1. Complete Unit 3 WS 01 - 1D Arrays

Oct 22-7:37 PM