Arrays

array - list of values of the same type

ex: array of ints:

```
10 24 19 31 43 51 12
```

Some terminology:

- length = 7
- indices: → 0 1 2 3 4 5 6
- indices go from 0 to length - 1 (6)
- element: 24, 19, etc.

Creating an array:

ex: array of size 10 of ints

```
int [10] heights = new int[10];
```

can access length via heights.length ⇒ 10

This creates an empty array of size 10

```
0 1 2 ... 9
```
To fill an array:

- Fill one index:
  \[ \text{heights}[2] = 72 \]

- Fill with variable for index:
  \[ \text{int x = 5;} \]
  \[ \text{heights[x] = 61;} \]

Can access elements of array:

\[ \text{ex: int sum = heights[0] + heights[1]} \]

Q: How to print the element in the middle?

System...println("middle element is: "+heights[heights.length/2]),

Typically, we fill an array with a loop.

Q: What kind of loop? Usually a for loop since we know number of iterations.

(ex: int[i] nums = new int[10]

(i) Fill nums with numbers from 0-9:

for (int i = 0; i < nums.length; i++)
	nums[i] = i)
(2) Print the values:

```java
for (int i = 0; i < nums.length; i++)
    System.out.println(nums[i]);
```

(3) Program to input 10 ints from user and print in reverse

```java
Scanner scan = new Scanner(System.in);

int arr[] = new int[10];
// get input
for (int i = 0; i < 10; i++)
    System.out.println("Enter number ");
    arr[i] = scan.nextInt();

// print in reverse
for (int i = length - 1; i >= 0; i--)
    System.out.println(arr[i]);
```

*Notice: ok to have "int i" in both for loops since they each have scope local to their for loop.*
(2) How to modify previous program to get number of inputs from user?

//Before for loop:

System.out.println("How many inputs?");
int num = scan.nextInt();
int[] arr = new int[num];

(3) Get max of numbers in arr?

int max = arr[0];
for(int i = 0; i < arr.length; i++)
if(arr[i] > max)
max = arr[i]

Initializer List: Set initial elements of array.

int[] scores = {98, 98, 65, 93, 100};
char[] vowels = {'a', 'e', 'i', 'o', 'u'};

Arrays, like Strings, are objects and can cause exceptions (errors caused by objects).

<Sample code>
Static Methods in Driver

Recall Lab #1

- Read user phrase, character
- Replace character with x
- Now: count the number of x's in the new phrase

New code:

```java
int count = 0;
for (int i = 0; i < newStr.length(); i++)
    if (newStr.charAt(i) == 'x')
        count++;
```

Now, main() method contains all the code. Better to place some code in other methods.

Method - function: set of code that performs an operation

Method header - defines properties of the method

```
public static return-type method-name (type1 var1,
    in same file as calling method
    can be void)
```

```
method is what method returns to
```

```
parameter list
```

<<SAMPLE CODE>>

Method replaceAndCount:

```java
```
```
public class ModifyString {

    public static void main (String[] args) {}
    // will fill in later

    public static int replaceAndCount (String s, String c) {
        String newstr = s.replace (c, "x");  // notice String 'x'
        int count = 0;
        for (int i = 0; i < newstr.length; i++)
            if (newstr.charAt(i) == 'x')  // char 'x'
                count++;

        return count;  // return must be last executed
                      // statement
    }

    // Convention: top-most method
    public static void main (String[] args) {
        Scanner scan = 
        System.out.println("Enter string: ");
        String str = scan.nextLine();
        System.out.println("Enter char to replace:");
        String chr = scan.nextLine();

        int numXs = replaceAndCount (str, chr);
    }
Sys.printl("num Xs:" + numXs);

// or replace last 2 lines with:
Sys.printl("num Xs:" + replaceAndCount(str, chr));

Back to scope:

 scan } have . S } have scope
cstr } Scope local . c } local to
chr } to main() newstr } replaceAndCount()
numXs } count

If a variable is going to be used by many methods,
make that variable accessible to all methods

public class ModifyString {

    public static String str; // global scope
    // accessible by all methods

} (Sample Code)