CP213 Lesson 6

* All elements of an array are the same type
* Array elements are in adjacent memory locations
* When declaring arrays, you can put the brackets on either side of the variable name
* Arrays have one instance variable – length (it cannot be changed)
* To add multiple values when declaring an array use brace brackets and comma separated values
* Since arrays are of class type, comparisons should use .equals()
* When an array of objects is created each index is initialized to null
* The use of = only copies the memory address
* To make a copy of an array you can use a for loop and add values to the new array individually
* If a and b are arrays and a==b returns true that means that a and b reference the **same** memory location
* If we don’t know how many arguments we will have to pass in the method, **varargs** needs to be followed to avoid maintenance problems by writing less code.

Ex.

accessModifier methodName(datatype… arg) {

// method body

}

* To define a varargs, three dots are used
* This is a cleaner way to do method overloading
* Rules
  + There can only be one variable argument in the method
  + The variable argument must be the last argument passed

The following would generate an error:

void methodName (String... a, int... b) {}

So would:

void methodName (String... a, int b) {}

* An “enumerated type” or just “enum” is like an array of constants

Ex.

enum WorkDay {MONDAY, TUESDAY, WEDNESDAY, THURSDAY, FRIDAY};

WorkDay meetingDay, availableDay;

meetingDay = WorkDay.THURSDAY;

availableDay = null;

* Two enums can be compared using ==
* Every enum has a .values() method that returns an array whose elements are the values of the enumerated type
* Use the .valueOf(String) method to convert an inputted string into an enum
* You can create a multi-dimensional array by adding multiple [] to the array definition
* Technically…you could create an array of n dimension

Ex.

char[][] page = new char[30][100];

* Using .length is equivalent to the number of “rows” in the array, thus page.length =30
* To get the number of columns in the array take the length of the first index page[0].length
* An array that has a different number of elements per row is called a **ragged array**

Keywords

* Args
* An array as a return type
* Array index
* Array Length
* ArrayList
  + Object in java that acts as an array except it has no fixed sized. Also has special methods like .contains() which can be helpful for high level searches.
* Array script
* Dynamic array
* Enumeration
  + Enums sort of act as an array of constants. These constants appear like strings however they are not and you can therefore compare them with == and !=.
* Homogenous
* Index out of boundary exception
* Multidimensional array
  + An array of arrays, 2D arrays can be very useful when creating tables or multiple rows of data. Arrays in java can technically have n dimensions.
* Partially filled Array
* Pass an array to a method
* Ragged array
  + A multi dimensional array that is not “square” or does not have the same number of elements in each row.
* values
* valueOf
  + Use the .valueOf(String) method to convert an inputted string into an enum