Chapter 9 CRT
Hielan Lee-Tremblay

1. The index value of the third element in an array is 2. This is because array's are 0 based so the element 1 would have an index of 0 (hence why there called 0 based). This continues to the third element which will have an index value of 2. (basically just the element place -1 = the index Ex. fifth element - 1 = 4 so the index is 4)

2.

```
int[] quantities = new int[20];
```

int signifies that the array will work with integer values (hold integer elements), the [ ] signify that this is an array (it will be a collection of integer variables array), quantities is the name of the array, = indicates that it is equal to what's to come (right of the equals sign) new tells the program to make space for the array, int[20] indicates that the array will hold 20 integer elements (indexes 0-19)

3.

```
double[] heights = {1.65, 2.15, 4.95};
```

double signifies that the array will work with double values (holds double elements), the [ ] signify that this is an array (it will be a collection of double variables array), heights is the name of the array, = indicates that it is equal to what is of the right of the equal sign, {1.65, 2.15, 4.95} is an array initializer that specifies the values the heights holds (use this { } instead of [ ] because we are given specific decimal values to hold)

4.

```
for (int grade : grades) {
    System.out.println(grade);
}
```

This for-each loop makes int grade equal to each element in the grades array. When grade get's printed it will print EACH of the elements in the grades array, hence why it's called a for-each loop. (there will be and iteration for each element in grades array so that grade prints out each element)

6. When you pass an entire array to a method then the method will be able to adjust values inside of the array, you get access to all of the information (if there are multiple elements), however on the other hand passing a single element to a method makes a copy of the element and doesn't change its actual value (this usually so the method can do calculations with it and return a new calculated value to wherever it's sending information to), sending a single element also only gives access to this element in the array none of the other elements.

7. Offset arrays are usually just what people refer to an "array" in java as. It's offset because it begins at 0 not 1. This is useful in the majority of cases because Java's memory works correlating to 0 based indexes. This means using an offset array with 0 based indexes will match elements, however using a 1 based numeric system will not match the elements to the memory location mapping it to the wrong spot which can lead to errors.

8. This code outputs the letter "i" in the print statement, because since arrays are 0 based the letter E = index 0 so charAt(0) = E, this means that charAt(3) = i not "a".