2. Arrays

An **array** in Java is a container which holds a set number of values of a single type. The length of an array is set at the time that it is declared or instantiated, and remains fixed for the life of an array. Additionally the type contained in an array can be a primitive datatype, or an Object type.

Some useful **array** information and vocabulary is contained here:

- Square Brackets or [] tell Java that we are declaring an array
- *element*: each item stored in an array is called an *element*
- elements within the same Java array must all be the same data type.
- *index*: each element in an array is found or accessed by its numerical *index*.
- arrays are zero-based, so the *index* values start at 0.
- the first element in any array is located at index: 0
- the last element of any array is located at index: *arrayName*.length - 1

Just like any variable in Java, an array must be declared. We declare the array to give it a name, and to allocate space for the size of the array that we declare. Note, that we also must determine the data type of the elements that will be stored in this array, so that Java knows how to store and manage the elements in the array.

Let's look at a Grade Book Example. Imagine that I have a College Programming Course that has 5 coding assignments. For our example, I will set the values of the grades.

> **Coding Challenge**: As a stretch, you could convert these examples to use the Java Scanner() as was described in the User Input Section of ***Week 2: Weekly Curriculum & Videos*.**

One way to represent program a simple **Grade Book** for our five assignment Programming Course could be as follows:

```
String firstName = "Mary";
String lastName = "Brown";
int assignmentOne = 100;
int assignmentTwo = 79;
int assignmentThree = 80;
int assignmentFour = 90;
int assignmentFive = 100;

System.out.println("Student: " + firstName + " " + lastName);
System.out.println("Grades:");
System.out.println("1: " + assignmentOne);
System.out.println("2: " + assignmentTwo);
System.out.println("3: " + assignmentThree);
System.out.println("4: " + assignmentFour);
System.out.println("5: " + assignmentFive);
```

As you can see, this works for small projects, but what if our Programming Course had 20 students each with 5 assignments, or 20 students each with 100 assignments. A simple Grade Book could get out of hand very quickly. This project can become more manageable by using arrays.

How do we declare an array? Arrays can be declared a number of ways, but each declaration has the same components:
(1) a datatype
(2) Square brackets or []
(3) a name.

Notice that the following three examples are very similar, with the third example actually setting the values within the array in the declaration. All three declare an array named *gradesArray*, all three tell Java that we are creating an **array of type int** (e.g. **int []**), and all three tell Java to ? e enough space for 5 elements.

- `int[] gradesArray = new int[5];`
- ```
  int[] gradesArray;
  gradesArray = new int[5];
  ```
- `int[] gradesArray = {100, 79, 80, 90, 100};`

Let's simplify the **Grade Book** Example now by using an array, and a for loop to print the grades:

```
String fullName = "Mary" + " " + "Brown";
int[] gradesArray = {100, 79, 80, 90, 100};

System.out.println("Student: " + fullName);
System.out.println("Grades: ");
for (int i = 0; i < 5; i++) {
    System.out.println((i+1) + ": " + gradesArray[i]);
}
```

Check out this rewrite of our **Grade Book** code, using an enhanced for loop to print the grades:

```
String fullName = "Mary" + " " + "Brown";
int[] gradesArray = {100, 79, 80, 90, 100};
int counter = 1;

System.out.println("Student: " + fullName);
System.out.println("Grades: ");
for (int grade : gradesArray) {
    System.out.println((counter++) + ": " + grade);
}
```

More Arrays

Advanced Topic:
What if you wanted to actually create a more robust Grade Book? One that would take more Student(s), and store their grades inside the Student Object.

Here we are going to explore an example that creates an Array of Objects. In this case, we would like to create an Array to hold our Student data. To accomplish this, we must first define what a Student will look like to be able to declare a Student Class that will be used to create our Objects.

Each Student will have a name, and an Array of grades. It will also need a way to create an Object of type Student (a Constructor), and a method to describe a Student (or print out the data corresponding to the student).

The next step will be to create an Array of type Student. We will also use the Scanner to take data from the console to use to create our arrays, we will create each Student Objects, and we will update that instance of an Object with the data that is input.

To reproduce this example, you will need to create a Java Project in Eclipse, and then create two Java classes within that Java Project.

1. **Student.java** -- a class in Java, but does not have a main method.
2. **GradeBook.java** -- a class in Java with a main method.

Student Class containing an array of grades (**Student.java**):

```
public class Student {
   String fullName = "";
   int[] grades;

   public Student(String fName,int[] grades) {
       this.fullName = fName;
       this.grades = grades;
   }
   public void describe() {
      System.out.println("Student: " + this.fullName);
```
```

```
        System.out.println("Grades:");
        for (int grade : this.grades) {
            System.out.println("\t" + grade + " ");
        }
        System.out.println();
      }
    }
```

Grade Book Example instantiating a Student Object (**GradeBook.java**):

```
import java.util.Scanner;

public class GradeBook {

    public static void main(String[] args) {
        String fullName = "";
        String lineVariable = "---------------------------";
        Scanner sc = new Scanner(System.in);

        System.out.println("Grade Book Example");
        System.out.println("Name of Course:");
        String courseName = sc.nextLine();
        System.out.println("How many students are in this class:");
        int numOfStudents = sc.nextInt();
        Student[] programmingStudents = new Student[numOfStudents];
        System.out.println("How many grades do you have per student?");
        int numOfGrades = sc.nextInt();
        sc.nextLine();

        for (int i = 0; i < programmingStudents.length; i++) {
            int[] grades = new int[numOfGrades];
            System.out.print("Enter Student's Full Name: ");
            fullName = sc.nextLine();
            for (int j = 0; j < grades.length; j++) {
                System.out.print("Enter a grade: ");
                grades[j] = sc.nextInt();
            }
            programmingStudents[i] = new Student(fullName, grades);
            sc.nextLine();
        }

        System.out.println("\n\n"+lineVariable+"\n "
               + courseName + " Grade Book \n"+lineVariable);
        for (Student student : programmingStudents) {
            student.describe();
         }
        System.out.println(lineVariable+"\n");

        sc.close();
    }

}
```

Resource:

- The Java Tutorials -- Arrays