PINNACLE COLLEGE

INTRODUCTION TO JAVA PROGRAMMING

TEACHER: KPIZINGUI EPHREM MAXIMILEN

EMAIL: kpizmax@gmail.com

TEL: 0245809867

ARRAY

- An array is a collection (sequence) of a fixed number of variables called elements or components, wherein all the elements are of the same data type.
- A one-dimensional array is an array in which the elements are arranged in a list form.

- The general form to declare a one-dimensional array is:

```
dataType[] arrayName;
```

where dataType is the element type and arrayName is a reference variable.

- The general syntax to instantiate an array object is:

```
arrayName = new dataType[intExp];
```

You can combine the statements in Lines 1 and 2 into one statement as follows:

```
dataType[] arrayName = new dataType[intExp];
```

- where intExp is any expression that evaluates to a positive integer. Also, the value of intExp specifies the number of elements in the array.

Example:

The statement:

```
int[] num = new int[5];
```

declares and creates the array num consisting of 5 elements. Each element is of type int.

The elements are accessed as num[0], num[1], num[2], num[3], and num[4].

Accessing Array Elements

- The general form (syntax) used to access an array element is:

  arrayName[indexExp]

- where indexExp, called the index, is an expression whose value is a nonnegative integer less than the size of the array. The index value specifies the position of the element in the array. In Java, the array index starts at 0.

SPECIFYING ARRAY SIZE DURING PROGRAM EXECUTION

- During program execution, you can first prompt the user to specify the size of the array and then instantiate the object.
- The following statements illustrate this concept (suppose that console is a Scanner

object initialized to the standard input device defined as static before the main void method):

```
int arraySize;
 System.out.print("Enter the size of the array: ");
 arraySize = console.nextInt();
System.out.println();
int[] list = new int[arraySize];
```

- For example, if the value of arraySize is 15, list is an array of size 15.

Array Initialization during Declaration

An array can also be initialized with specific values when it is declared. For example, the following Java statement declares an array, sales, of five elements and initializes those elements to specific values:

```
double[] sales = {12.25, 32.50, 16.90, 23, 45.68};
```

and separated by commas.

Here, sales[0] = 12.25, sales[1] = 32.50, sales[2] =16.90, sales[3] = 23.00, and sales[4]= 45.68.

NB: Note the following about declaring and initializing arrays:

When declaring and initializing arrays, the size of the array is determined

by the number of initial values in the initializer list within the braces.

If an array is declared and initialized simultaneously, we do not use the

operator new to instantiate the array object.

INSTANCE VARIABLE LENGTH

The variable length contains the size of the array. Because length is a public member, it can be directly accessed in a program using the array name and the dot operator.

Consider the following declaration:

```
int[] list = {10, 20, 30, 40, 50, 60};
```

This statement creates the array list of six elements and initializes the elements using the values given. Here, list.length is 6.

Consider the following statement:

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

This statement creates the array numList of 10 elements and initializes each element to 0.

Because the number of elements of numList is 10, the value of numList.length is 10.

Processing One-Dimensional Arrays

- Some basic operations performed on a one-dimensional array are initializing the array, reading data into the array, storing output data in the array, and finding the largest and/or smallest element in the array. If the data type of an array element is numeric, some common operations are to find the sum and average of the elements of the array.
- Each of these operations requires the ability to step through the elements of the array, which is easily accomplished by using a for loop.

Suppose that we have the following statements:

```
int[] list = new int[100]; //list is an array of size 100
```

The following for loop steps through each element of the array list, starting at the first element of list:

```
  for (int i = 0; i < list.length; i++) //Line 1
//process list[i], the (i + 1)th element of list //Line 2
```

- If processing list requires inputting data into list, the statement in Line 2 takes the form of an input statement, such as in the following code. The following statements read 100 numbers from the keyboard and store the numbers into list:

```
for (int i = 0; i < list.length; i++) //Line 1
    list[i] = console.nextInt(); // Line 2
```

- Similarly, if processing list requires outputting data, then the statement in Line 2 takes the form of an output statement. The following for loop outputs the elements of list:

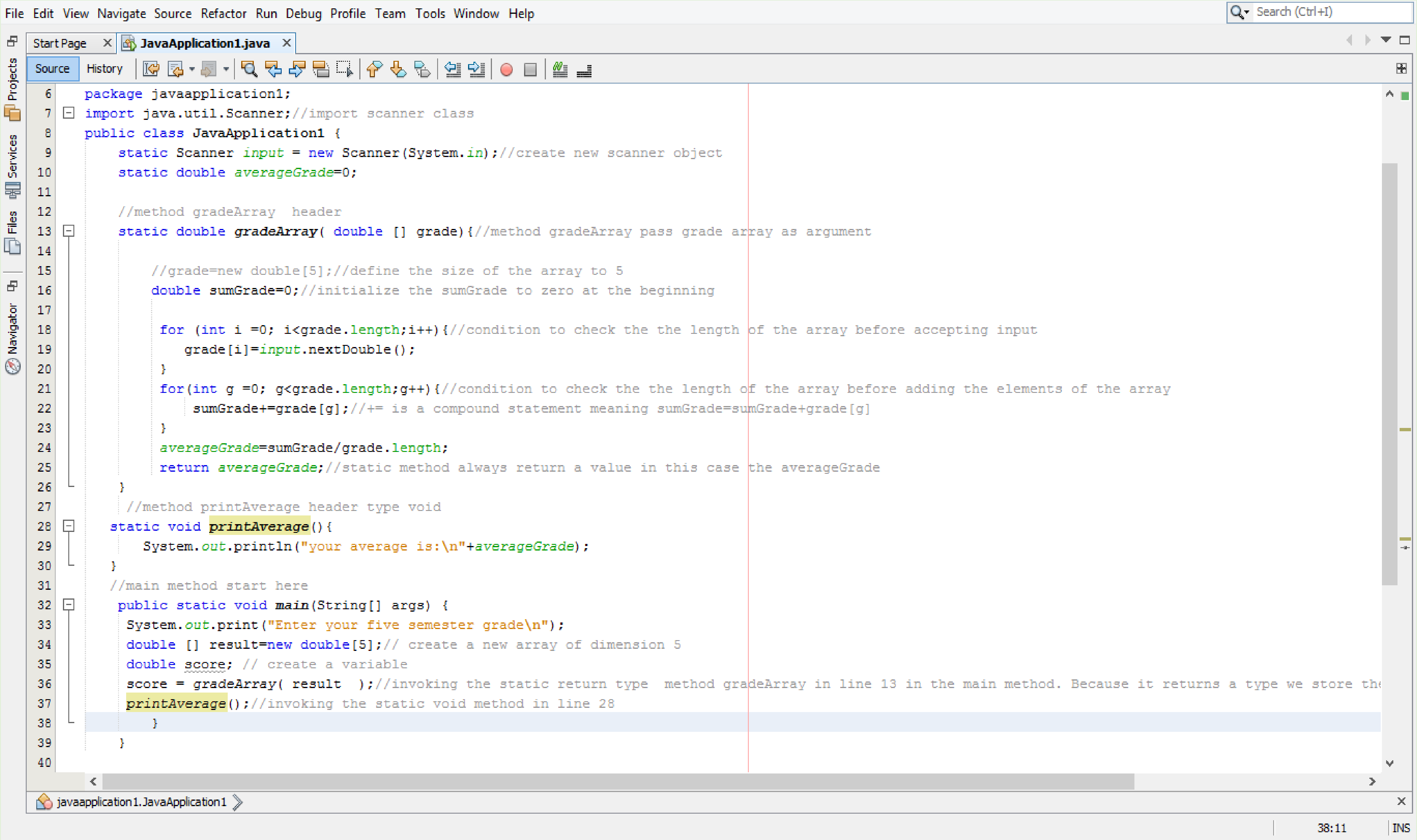
```
for (int i = 0; i < list.length; i++) //Line 1
    System.out.print(list[i] + " "); //Line 2
```

Case study

Write a java program that prompts a student to enter his grade in an array of size 5 and calculate his average grade.

Your program should have:

1. A method to enter your five grade in the array and return your average score.
2. A method to display your average score.



```
 6  package javaapplication1;
 7  import java.util.Scanner;//import scanner class
 8  public class JavaApplication1 {
 9      static Scanner input = new Scanner(System.in);//create new scanner object
10      static double averageGrade=0;
11
12      //method gradeArray  header
13      static double gradeArray( double [] grade){//method gradeArray pass grade array as argument
14
15          //grade=new double[5];//define the size of the array to 5
16          double sumGrade=0;//initialize the sumGrade to zero at the beginning
17
18           for (int i =0; i<grade.length;i++){//condition to check the the length of the array before accepting input
19              grade[i]=input.nextDouble();
20           }
21           for(int g =0; g<grade.length;g++){//condition to check the the length of the array before adding the elements of the array
22               sumGrade+=grade[g];//+= is a compound statement meaning sumGrade=sumGrade+grade[g]
23           }
24           averageGrade=sumGrade/grade.length;
25           return averageGrade;//static method always return a value in this case the averageGrade
26      }
27       //method printAverage header type void
28     static void printAverage(){
29         System.out.println("your average is:\n"+averageGrade);
30     }
31     //main method start here
32      public static void main(String[] args) {
33       System.out.print("Enter your five semester grade\n");
34       double [] result=new double[5];// create a new array of dimension 5
35       double score; // create a variable
36       score = gradeArray( result  );//invoking the static return type  method gradeArray in line 13 in the main method. Because it returns a type we store the
37       printAverage();//invoking the static void method in line 28
38         }
39      }
40
```

Tutorial

Write a java program that prompts a student to enter his grades in an array and calculate his average grade.

Your program should have:

1. A method to enter your grade in an array and return your average score. The size of the array should be prompt by the user this is equal to the number of quizzes taken by the student in that course.
2. A method to display your average score including your surname, first name and index number and program of study
3. A method to display your grade.

85< A+ <100 ; 80<A- < 84; 70<B+<79; 60<B-<69; 50<C<59 and 0<F<49. (25 marks)