

Intro to Java Week 3 Coding Assignment

Points possible: 70

| Category | Criteria | % of Grade |
|---------------|---|------------|
| Functionality | Does the code work? | 25 |
| Organization | Is the code clean and organized? Proper use of white space, syntax, and consistency are utilized. Names and comments are concise and clear. | 25 |
| Creativity | Student solved the problems presented in the assignment using creativity and out of the box thinking. | 25 |
| Completeness | All requirements of the assignment are complete. | 25 |

Instructions: In Eclipse, or an IDE of your choice, write the code that accomplishes the objectives listed below. Ensure that the code compiles and runs as directed. Take screenshots of the code and of the running program (make sure to get screenshots of all required functionality) and paste them in this document where instructed below. Create a new repository on GitHub for this week's assignments and push this document, with your Java project code, to the repository. Add the URL for this week's repository to this document where instructed and submit this document to your instructor when complete.

Coding Steps:

1. Create an array of int called ages that contains the following values: 3, 9, 23, 64, 2, 8, 28, 93.

```
1
2 public class AssignmentCode {
3
4     static final int sum2 = 0;
5
6     public static void main(String[] args) {
7         //1.Create an array of int called ages that contains the following values: 3, 9, 23, 64, 2, 8, 28, 93.
8         int [] ages = {3, 9, 23, 64, 2, 8, 28, 93};
9
10    }
```

- a. Programmatically subtract the value of the first element in the array from the value in the last element of the array (i.e. do not use `ages[7]` in your code). Print the result to the console.

```
/* a. Programmatically subtract the value of the first element in the array from
 * the value in the last element of the array (i.e. do not use ages[7] in your
 * code). Print the result to the console.
 */
System.out.println(ages[ages.length-1]-ages[0]);|
```

- b. Add a new age to your array and repeat the step above to ensure it is dynamic (works for arrays of different lengths).

```
//b. Add a new age to your array and repeat the step above to ensure it is dynamic (works for arrays of different lengths).
int [] ages2 = {3, 9, 23, 64, 2, 8, 28, 93,80,85};
System.out.println(ages2[ages2.length-1]-ages2[0]);
```

- c. Use a loop to iterate through the array and calculate the average age. Print the result to the console.

```
//c. Use a loop to iterate through the array and calculate the average age. Print the result to the console.
int sum =0;
for( int i =0; i<=ages.length-1;i++) {
    //int sum =0;
    sum = sum + ages[i];
    //sum++;
}
double average = (double)sum/(ages.length);
System.out.println(average);
```

2. Create an array of String called names that contains the following values: "Sam", "Tommy", "Tim", "Sally", "Buck", "Bob".

```
//2. Create an array of String called names that contains the following values:  
//"Sam", "Tommy", "Tim", "Sally", "Buck", "Bob".  
  
String [] names = {"Sam", "Tommy", "Sally", "Buck", "Bob"};
```

- a. Use a loop to iterate through the array and calculate the average number of letters per name. Print the result to the console.

```
//a. Use a loop to iterate through the array and calculate the average number of letters per name.  
//Print the result to the console.  
  
int sumOfLetters= 0;  
for (int j=0; j<=names.length-1; j++) {  
    sumOfLetters += names[j].length();  
}  
//System.out.print(names[j] + " ");  
double average2 = (double)sumOfLetters/names.length;  
System.out.println(average2);
```

- b. Use a loop to iterate through the array again and concatenate all the names together, separated by spaces, and print the result to the console.

```
// b. Use a loop to iterate through the array again and concatenate all the names together, separated by spaces, and print the result to the console.  
String allNames = " ";  
for (int n =0; n<names.length; n++) {  
    allNames = allNames + " ";  
}  
System.out.print(allNames);
```

3. How do you access the last element of any array?

To access the last element of any array is = `(name of the array variable[name of the array variable.length-1])`.

Example: In this assignment the name of the array variable is names. Therefore to access the last element of the array = `names[names.length-1]`

4. How do you access the first element of any array?

To access the first element of any array : names of array variable at index 0. We can write it in this way: `name of the array variable[0]`.

Example: In this assignment the name of the array variable is names. Therefore to access the first element of the array = `names[0]`

5. Create a new array of int called nameLengths. Write a loop to iterate over the previously created names array and add the length of each name to the nameLengths array.

```
/*  
 * 5. Create a new array of int called nameLengths. Write a loop to iterate over  
 * the previously created names array and add the length of each name to the  
 * nameLengths array.  
 */  
int[] nameLengths = new int[5];  
for(int m =0; m<names.length-1; m++) {  
    nameLengths[m] = names[m].length();  
}
```

6. Write a loop to iterate over the nameLengths array and calculate the sum of all the elements in the array. Print the result to the console.

```
/*
 * 6. Write a loop to iterate over the nameLengths array and calculate the sum
 * of all the elements in the array. Print the result to the console.
 */
int sum4 = 0;
for(int h=0; h<nameLengths.length-1; h++) {
    sum4 =sum4 + nameLengths[h];
}
System.out.println(sum4);
}
```

7. Write a method that takes a String, word, and an int, n, as arguments and returns the word concatenated to itself n number of times. (i.e. if I pass in “Hello” and 3, I would expect the method to return “HelloHelloHello”).

```
/*
 * 7. Write a method that takes a String word, and an int n, as arguments and
 * returns the word concatenated to itself n number of times. (i.e. if I pass in
 * “Hello” and 3, I would expect the method to return “HelloHelloHello”).
 */
public static String concatenatedWords(String word, int n){

    String result= "";

    if(n==0) return result;

    for (int i =1; i<=n; i++) {
        result = result + word;
    }
    return result;
}
```

8. Write a method that takes two Strings, firstName and lastName, and returns a full name (the full name should be the first and the last name as a String separated by a space).

```
/*
 * 8. Write a method that takes two Strings, firstName and lastName, and returns
 * a full name (the full name should be the first and the last name as a String
 * separated by a space).
 *
 *
 */

public static String writeFullName(String firstName, String lastName){

    String fullName = firstName + " " + lastName;
    return fullName;

}
```

9. Write a method that takes an array of int and returns true if the sum of all the ints in the array is greater than 100.

```
//9. Write a method that takes an array of int and returns true if the sum of all the ints in the array is greater than 100.

    static int [] num= {40,20,30,50};
    private static int sum1;

    static boolean findSum(int[] add){

        int sum5=0;

        if(sum5 >100) return true;

        for( int i =1; i<add.length; i++) {
            sum5 = sum5+add[i];
        }

        return true;

    }
```


10. Write a method that takes an array of double and returns the average of all the elements in the array.

```
//10. Write a method that takes an array of double and returns the average of all the elements in the array.  
  
public static double calculateAverageOfAllElements(double [] arr) {  
  
    double [] decimal = {40.45, 60.50, 80.40, 100.30,20.0};  
  
    double sum6 =0;  
  
    double average =0;  
  
    for( int i =0; i<decimal.length-1; i++) {  
  
        sum6 = sum6 + decimal[i];  
  
        double average1 = (double) sum6/decimal.length;  
    }  
  
    return average;  
}
```

11. Write a method that takes two arrays of double and returns true if the average of the elements in the first array is greater than the average of the elements in the second array.

```
//11. Write a method that takes two arrays of double and returns true if
//the average of the elements in the first array is greater than the average of the elements in the second array.

    public static boolean isSum1GreaterThanSum2(double [] array1, double [] array2) {

        if (sum1>sum2) return true;

        double[] array11 = {30.30, 20.50, 50.50};

        double sum1 = 0;

        for( int i =0; i<array11.length-1; i++) {

            sum1 = sum1 + array11[i];

        }

        double [] array22 = {20.50, 10.50, 30.40};

        double sum2 =0;

        for( int j =0; j<array22.length-1; j++) {

            sum2 = sum2 + array22[j];

        }

        return true;

    }
```


12. Write a method called willBuyDrink that takes a boolean isHotOutside, and a double moneyInPocket, and returns true if it is hot outside and if moneyInPocket is greater than 10.50.

```
//12. Write a method called willBuyDrink that takes a boolean isHotOutside, and a double moneyInPocket,
//and returns true if it is hot outside and if moneyInPocket is greater than 10.50.

public static boolean willBuyDrink(boolean isHotOutside, double moneyInPocket) {

    if (isHotOutside && moneyInPocket>10.50) {
        return true;
    }else {

        return false;
    }
}
}
```

13. Create a method of your own that solves a problem. In comments, write what the method does and why you created it.

```
//13. Write a method called willBecomeBackendSoftware that takes a boolean doYouStudy5HoursADay, and a boolean doYouPracticeMoreCodes
//and returns true if it is you study 5 hours a day and if you practice more codes.

public static boolean willBecomeBackendSoftware(int study5HoursADay, boolean doYouPracticeMoreCodes) {

    //The method check if a Prominoe student study 5 hours a day and practice more codes.
    //If the student studies 5 hours a day and practice more then he will become a backend software programmer

    if(study5HoursADay ==5 && doYouPracticeMoreCodes) {
        //return true;
        System.out.println("I am going to be a Backend Software.");
    }
    return true;
}
}
```

Screenshots of Code:

Screenshots of Running Application:

URL to GitHub Repository: