



## Intro to Java Weeks 3-4 Coding Assignment

**URL to GitHub Repository:** <https://github.com/kymzaidi/Week-3-4-Coding-Assignment-Submission>

**URL to Public Link of your Video:** <https://youtu.be/F-39q2boB4Q>

---

### Instructions:

1. Follow the **Coding Steps** below to complete this assignment.

- 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.
- Create a new repository on GitHub for this week's assignment and push your completed code to this dedicated repo.
- Create a video showcasing your work:
  - In this video: record and present your project verbally while showing the results of the working project.
  - Easy way to Create a video: Start a meeting in Zoom, share your screen, open Eclipse with the code and your Console window, start recording & record yourself describing and running the program showing the results.
  - Your video should be a maximum of 5 minutes.
  - Upload your video with a public link.
  - Easy way to Create a Public Video Link: Upload your video recording to YouTube with a public link.

2. In addition, please include the following in your Coding Assignment Document:

- The URL for this week's GitHub repository.
- The URL of the public link of your video.

3. Save the Coding Assignment Document as a .pdf and do the following:

- Push the .pdf to the GitHub repo for this week.
  - Upload the .pdf to the LMS in your Coding Assignment Submission.
-



## Intro to Java Weeks 3-4 Coding Assignment

### Coding Steps — Arrays and Methods

```
package Week04;

public class CodingAssignment {

    public static void main(String[] args) {

        // TODO Auto-generated method stub

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

        int[] ages = {3, 9, 23, 64, 2, 8, 28, 93};

        //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).

        int[] ages1 = {3, 9, 23, 64, 2, 8, 28, 93, 10};

        System.out.println(ages1[ages1.length-1] - ages[0]);

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

        double sum = 0;

        for (int num : ages) {

            sum += num;

        }

        double average = 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".
```



## Intro to Java Weeks 3-4 Coding Assignment

```
String[] names = {"Sam", "Tommy", "Tim", "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.

int sumOfAllLettersPerName = 0;

for (String name : names) {

    sumOfAllLettersPerName += name.length();

}

double averageLettersPerName = sumOfAllLettersPerName / names.length;

System.out.println(averageLettersPerName);

// 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 concatenateAllTheNames = " ";

for (String name : names) {

    concatenateAllTheNames += name + " ";

}

System.out.println(concatenateAllTheNames);

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

System.out.println(names[names.length - 1]);

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

System.out.println(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.

int[] nameLengths = new int[names.length];

for (int i = 0; i < names.length; i++) {

    nameLengths[i] = names[i].length();

}
```



## Intro to Java Weeks 3-4 Coding Assignment

```
}

// 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 sum1 = 0;

for (int name : nameLengths) {

    sum1 += nameLengths[name];

}

System.out.println(sum1);

}

// 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 expect the method to return "HelloHelloHello").

public static String concatenatedWord(String word, int n) {

    String result = "";

    for (int i = 0; i < n; i++){

        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).

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

    String fullName = firstName + " " + lastName;

    return fullName;

}
```



## Intro to Java Weeks 3-4 Coding Assignment

// 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.

```
public static boolean sum(int[] array) {  
  
    int sum = 0;  
  
    for (int i = 0; i < array.length; i++) {  
  
        sum += array[i];  
  
    }  
  
    return sum > 100;  
  
}
```

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

```
public static double returnAverage(double[] array) {  
  
    double sum = 0;  
  
    for (double number : array) {  
  
        sum += number;  
  
    }  
  
    return sum / array.length;  
  
}
```

// 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 returnTrueAverage(double[] array1, double[] array2) {  
  
    double sum1 = 0;  
  
    for (int i = 0; i < array1.length; i++) {  
  
        sum1 += array1[i];  
  

```



## Intro to Java Weeks 3-4 Coding Assignment

```
}

double average1 = sum1 / array1.length;

double sum2 = 0;

for (int i = 0; i < array2.length; i++) {

    sum2 += array2[i];

}

double average2 = sum1 / array2.length;

return average1 > average2;

}

// 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.

public static boolean WillMyHusbandReturnOnTimeSoICanFinishThisHomework(double
time) {

    if (time < 11.00) {

        return true;

    }

}
```



## Intro to Java Weeks 3-4 Coding Assignment

```
} else {
```

```
return false;
```

```
}
```

```
//waiting on my husband to finish this homework. if he can make it home by 11pm, i  
can submit this homework by midnight. If not, i fail. This is a boolean expression.
```

```
}
```

```
}
```