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. Lastly, in the Learning Management System, click the “Add Submission” button and paste the URL to your GitHub repository.

**Coding Steps:**

* Create an array of int called ages that contains the following values: 3, 9, 23, 64, 2, 8, 28, 93.
* 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.
* Add a new age to your array and repeat the step above to ensure it is dynamic (works for arrays of different lengths).
* Use a loop to iterate through the array and calculate the average age. Print the result to the console.
* Create an array of String called names that contains the following values: “Sam”, “Tommy”, “Tim”, “Sally”, “Buck”, “Bob”.
* Use a loop to iterate through the array and calculate the average number of letters per name. Print the result to the console.
* 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.
* How do you access the last element of any array?
* How do you access the first element of any array?
* 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.
* 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.
* 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”).
* 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).
* 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.
* Write a method that takes an array of double and returns the average of all the elements in the array.
* 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.
* 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.
* Create a method of your own that solves a problem. In comments, write what the method does and why you created it.

**Screenshots of Code:**

**Arrays:**

**package** week3;

**public** **class** arrays {

**public** **static** **void** main(String[] args) {

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

\* 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.

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

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

\*/

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

**int** first = ages[0];

**int** last = ages[ages.length-1];

**int** dif = first - last;

System.***out***.println("Subtract the value of the 1st element from the last element is " + dif);

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

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

**int**[] ages2 = { 3, 9, 23, 64, 2, 8, 28, 93, 3 };

**int** first2 = ages2[0];

**int** last2 = ages2[ages2.length-1];

**int** dif2 = first2 - last2;

System.***out***.println("Subtract the value of the 1st element from the last element is " + dif2);

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

**double** sum = 0;

**double** avg = 0;

**for**(**int** x = 0;x <= ages2.length-1;x++) {

sum = sum + ages2[x];

**if** (x > 0) {

avg = sum / x;

}

System.***out***.println("Average is " + avg);

}

/\*

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

How do you access the last element of any array?

array[array.length-1]

How do you access the first element of any array?

array[0]

\*/

String[] names = {"Sam","Tommy","Tim","Sally","Buck","Bob"};

//Use a loop to iterate through the array and calculate the average number of letters per name.

**double** sum2 = 0;

**double** avg2 = 0;

**for**(**int** y = 0;y <= names.length-1;y++) {

sum2 = sum2 + names[y].length();

**if** (y > 0) {

avg2 = sum2 / y;

}

System.***out***.println("Average is " + avg2);

}

//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.

**for**(**int** z= 0;z <= names.length-1;z++) {

System.***out***.print(names[z] + " ");

}

/\*

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.

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

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

String name = names[i];

nameLengths[i] = name.length();

}

**int** sum3 = 0;

**for**(**int** h = 0; h <= nameLengths.length-1; h++) {

sum3 = sum3 + nameLengths[h];

System.***out***.println("Sum is: " + sum3);

}

}

}

**Methods:**

**package** week3;

**public** **class** methods {

**static** String catNumTimes(String word, **int** n) {

String output = word;

**if**(n > 1)

**for**(**int** x = 2; x <= n; x++) {

output += word;

}

**return** output;

}

**static** **void** catFirstLast(String first, String last) {

System.***out***.println(first + " " + last);

}

**static** **void** testOver100(**int**[] input) {

**int** sum = 0;

**for**(**int** x = 0; x < input.length; x++) {

sum = sum + input[x];

}

System.***out***.println(sum > 100);

}

**static** **double** returnAverage(**double**[] input) {

**double** tot = 0;

**for**(**int** x = 0; x < input.length; x++) {

tot = tot + input[x];

}

**return**(tot/input.length);

}

**static** **void** averageTest(**double**[] array1, **double**[] array2) {

System.***out***.println(*returnAverage*(array1) > *returnAverage*(array2));

}

**static** **void** willBuyDrink(**boolean** hotYesNo, **double** money) {

System.***out***.println((hotYesNo && money > 10.50));

}

**static** **void** commentsPercent(**double** code, **double** comments) {

System.***out***.println("You have spent " + comments/code\*100 + "% lines on commenting your code");

}

**public** **static** **void** main(String[] args) {

//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”).

System.***out***.println(*catNumTimes*("Hello" , 3));

//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).

*catFirstLast*("Michael","Franzen");

//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.

**int**[] myArray = { 50, 10, 41 };

*testOver100*(myArray);

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

//in the array.

**double**[] myDoubleArray = { 21.1, 21.1, 21.2 };

*returnAverage*(myDoubleArray);

//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.

**double**[] myDoubleArray2 = {25.2, 22.2, 22.2};

**double**[] myDoubleArray3 = {22.3, 22.3, 22.3};

*averageTest*(myDoubleArray2, myDoubleArray3);

//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.

**boolean** isHotOutside = **true**;

**double** moneyInPocket = 10.51;

*willBuyDrink*(isHotOutside, moneyInPocket);

//Create a method of your own that solves a problem. In comments, write what the method

//does and why you created it.

//commentsPercent will return the percent of this class that was comments

//So I know how painless or painful it was to document my code

**int** linesOfCode = 81;

**int** linesOfComments = 16;

*commentsPercent*(linesOfCode,linesOfComments);

}

}

**Screenshots of Running Application:**

**Arrays:**

Subtract the value of the 1st element from the last element is -90

Subtract the value of the 1st element from the last element is 0

Average is 0.0

Average is 12.0

Average is 17.5

Average is 33.0

Average is 25.25

Average is 21.8

Average is 22.833333333333332

Average is 32.857142857142854

Average is 29.125

Average is 0.0

Average is 8.0

Average is 5.5

Average is 5.333333333333333

Average is 5.0

Average is 4.6

Sam Tommy Tim Sally Buck Bob Sum is: 3

Sum is: 8

Sum is: 11

Sum is: 16

Sum is: 20

Sum is: 23

**Methods:**

HelloHelloHello

Michael Franzen

true

true

true

You have spent 19.753086419753085% lines on commenting your code

**URL to GitHub Repository:** [**https://github.com/mikefranzen/week3.git**](https://github.com/mikefranzen/week3.git)