

```
public class Week3And4 {

    public static void main(String[] args) {

        /*Question 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};

        /*Question 1a: Programmatically subtract the values of the first element in the

        * array from the value of the last element in the array (i.e. do not use ages

        * [7] in your code). Print the result to the console.

        */

        System.out.println("\nQuestion # 1a:");

        int lastElementMinusFirstElement = (ages[ages.length - 1] - ages[0]);

        System.out.println(lastElementMinusFirstElement);

        /* Question 1b: Add a new age to your array and repeat the step above to ensure it

        * is dynamic (works for arrays of different lengths)

        */

        System.out.println("\nQuestion# 1b:");
```

```

int[] ages2 = {3,9,23,64,2,8,28,93,95};

int lastElementMinusFirstElement2 = ages2[ages2.length-1]- ages2[0];

System.out.println(lastElementMinusFirstElement2);

/* Question 1c: Use a loop to iterate through the array and calculate the average
* age. Print the result to the console.
*/

System.out.println("\nQuestion# 1c:");

double sum = 0; //not sure if I should've used double or int to show the average of the
ages

for(int x : ages) {

sum += x;

}

System.out.println(sum/ages.length);

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

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

/* Question 2a: Use a loop to iterate through the array and calculate the average

```

```
/* number of letters per name. Print the result to the console

*/

System.out.println("\nQuestion # 2a:");

int sumAvg = 0;

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

    sumAvg += names[i].length();

}

int average = sumAvg / names.length;

System.out.println("The average number of letters in the array is " + average + ".");

/* Question 2b: 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.

*/

System.out.println("\nQuestion # 2b:");

String concatedNames = "";

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

    concatedNames += ( names[i] + " ");

}
```

```

System.out.println(concatatedNames);

/* Question 3: How do you access the last element of any array?

*/

System.out.println("\n Question # 3:");

System.out.println("To access the last element of an array, enter:
'arrayName[arrayName.length - 1]'.");

/*

* Question 4: How do you access the first element of any array?

*/

System.out.println("\n Question # 4:");

System.out.println("To access the first element of an array, enter: 'arrayName[0]'.");

/*

* Question 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

*/

System.out.println("\n Question # 5:");

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

```

```
for (int i = 0; i < names.length; i++) {  
  
    nameLengths[i] = names[i].length();  
  
    //System.out.println( "Value of i: " + i);  
  
}
```

```
for (int x : nameLengths) {  
  
    System.out.println(x);  
  
}
```

```
/*
```

```
* Question 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.
```

```
*/
```

```
System.out.println( "\nQuestion # 6:");
```

```
int sumElementsArray = 0;
```

```
for ( int i = 0; i < nameLengths.length; i ++ ) {  
  
    sumElementsArray += nameLengths[i];  
  
}
```

```
System.out.println(sumElementsArray);
```

```
}
```

```
/*
```

```
* Question 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 concatWords(String word, int n) {
```

```
String answer = "";
```

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

```
    answer += word;
```

```
}
```

```
return answer;
```

```
}
```

```
/*
```

```
* Question 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;
```

```
}
```

/*

* Question 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 validate(int array) {
```

```
int[] numArray = new int[array];
```

```
int sumArray = 0;
```

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

```
sumArray += numArray[i];
```

```
}
```

```
if (sumArray > 100) {
```

```
    return true;
```

```
} else {
```

```
    return false;
```

```
}
```

```
}
```

```
/*
```

```
* Question 10: Write a method that takes an array of double and returns the average of  
all the
```

```
* elements in the array.
```

```
*/
```

```
public static double doubleAvg(int averageNumbers) {
```

```
    double[] avgNumber = new double[averageNumbers];
```

```
    double doubAvg = 0;
```

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

```
        doubAvg += avgNumber[i];
```

```
    }
```

```
    double averageNum = doubAvg / avgNumber.length;
```



```
return averageNum;
```

```
}
```

```
/*
```

```
* Question 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 arrayTest(int one, int two) {
```

```
double[] doubOne = new double[one];
```

```
double[] doubTwo = new double[two];
```

```
double doubAvgOne = 0;
```

```
double dubAvgTwo = 0;
```

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

```
doubAvgOne += doubOne[i];
```

```
}
```

```
double avgNumOne = doubAvgOne / doubOne.length;
```

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

```

dubAvgTwo += doubTwo[i];

}

double avgNumTwo = dubAvgTwo / doubTwo.length;

if ( avgNumOne > avgNumTwo) {

return true;

} else {

return false;

}

}

```

```

/*

```

* Question 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 == true && moneyInPocket > 10.50) {

return true;

```

```
} else {
```

```
return false;
```

```
}
```

```
}
```

```
/*
```

```
* Question 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 getCoffee(double moneyOnGiftcard, boolean noCoffeeAtHome) {
```

```
if (moneyOnGiftcard >= 7.50 && noCoffeeAtHome == true) {
```

```
return true;
```

```
} else {
```

```
return false;
```

```
}
```

```
}
```

```
/*
```

* The method above will determine if you are able to get coffee depending on how much money you have

* on a gift card, and if you have no coffee at home. I created this method because I love getting Starbucks on

* my way to work in the mornings, and I was given gift cards this past Christmas to fund my addiction.

*/

}

```
Problems Javadoc Declaration Console X
<terminated> Week3And4 [Java Application] /Library/Java/JavaVirtualMachines/jdk-17.0.5.jdk/Contents/Home/bin/java (Jan 28, 2023, 7:43:07 PM - 7:43:08 PM) [pid: 2762]

Question # 1a:
90

Question# 1b:
92

Question# 1c:
28.75

Question # 2a:
The average number of letters in the array is 3.

Question # 2b:
Sam Tommy Tim Sally Buck Bob

Question # 3:
To access the last element of an array, enter: 'arrayName[arrayName.length - 1]'.

Question # 4:
To access the first element of an array, enter: 'arrayName[0]'.

Question # 5:
3
5
3
5
4
3

Question # 6:
23
```