```java
import java.util.Arrays;
import java.util.ArrayList;
import java.util.List;
import java.lang.Math;
//link to gitgub https://github.com/MiloGonz1990/ArraysAndMethods
//link to youtube https://youtu.be/1u5DcrE9bmc
public class ArrayAndMethods {

        public static void main(String[] args) {
                //Create an array of int called ages

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

                // 1.a subtract the value of the first element in the array from the
value in the last element of the array
                int firstElement = ages[0];
                int lastElement = ages[ages.length - 1];

                System.out.println(lastElement - firstElement);

                //1.b Add a new age to your array and repeat the step above to
ensure it is dynamic
                //to make array dynamic i had to convert array into ArrayList using
a for loop
                List<Integer> newAgelist = new ArrayList<>();

        for (int i : ages) {
            newAgelist.add(i);
        }

        newAgelist.add(100);
        System.out.println("new arrayList with new number: "+ newAgelist);

        System.out.println(newAgelist.get(newAgelist.size() - 1)-
newAgelist.get(0));
        //arrayList makes it easy to add and remove an index with the build in
methods java has
        newAgelist.remove(newAgelist.size()-1);
        System.out.println("removed new age: "+ newAgelist);
        //1.c Use a loop to iterate through the array and calculate the average age
        double ageSum = 0;
        for(int i : ages) {
                ageSum = ageSum+i;
        }
        double avgAge = ageSum/ages.length;
        System.out.println("The average age is: "+ avgAge + "\n");

        //2.Create an array of String called names
                String[] names = {"Sam", "Tommy", "Tim", "Sally", "Buck", "Bob"};
                System.out.println("Array of names: "+ Arrays.toString(names));
```

```java
                //Use a loop to iterate through the array and calculate the average
number of letters per name
                double sumLetters = 0;
                for(String name : names) {
                        sumLetters = sumLetters+name.length();
                }
                double avgLetters = sumLetters/names.length;

                System.out.println("average number of letters: "+ avgLetters);

                //3.How do you access the last element of any array? To get the last
element, you can access it using the index array.length - 1

                System.out.println("last element of array: "+ ages[ages.length -
1]);

                //4.How do you access the first element of any array? To get the
first element, you can access it using index 0.

                System.out.println("first element of array: "+ ages[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();
                }
                for(int numLetter : nameLengths ) {
                        System.out.println(numLetter);
                }
                //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
            // Creating variable to store the sum
            int sumArray = 0;
            for(int i = 0;i < nameLengths.length; i++ ) {
                sumArray += nameLengths[i];
            }
            System.out.println("sum of all elements in array: "+sumArray);



                //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.

            System.out.println(Xword("Java", 5));


                //8.Write a method that takes two Strings, firstName and lastName,
and returns a full name
```

```java
        System.out.println(FullName("Maelo", "Gonzalez"));


            //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.

        System.out.println(greaterThan100(ages));

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

        double[] dubArr = {45.4,53.54,46.6,99.9};
        System.out.println(doubleArry(dubArr));
            //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.

        double[] secondDubArr = {55.5,32.5,99.9,88.8};
        System.out.println(firstOrSecond(dubArr,secondDubArr));


            //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.
        System.out.println(willBuyDrink(true,20.20));
            //13.Create a method of your own that solves a problem. In comments,
write what the method does and why you created it.
        //used method to return the square root of a number
        System.out.println(squareRoot(100));

        }
        // java return methods
    //7.
        public static String Xword(String word, int n) {
            String Xword = "";
            for (int i = 0; i < n; i++) {
                Xword += word;
            }
            return "\n" + Xword;

        }
    //8.
        public static String FullName(String firstName, String lastName) {
            return  firstName + " " + lastName;
        }
    //9.
        public static Boolean greaterThan100(int[] arrayInt) {
            int sum = 0;
            for(int i = 0; i < arrayInt.length; i++) {
                    sum += arrayInt[i];
```

```java
            }
            if (sum>100) {
                    return true;
            }
            return false;
        }
    //10.
        public static double doubleArry(double[] arr) {
            //Average Using the Java Stream
            return Arrays.stream(arr).average().orElse(Double.NaN);
        }
    //11.
        public static boolean firstOrSecond(double[] firstArr, double[]
secondArr) {
            return doubleArry(firstArr) > doubleArry(secondArr);
        }
        //12.
        public static boolean willBuyDrink(boolean isHotOutside, double
moneyInPocket) {
            if(isHotOutside==true&&moneyInPocket>10.50) {
                    return true;
            }else {
                    return false;
            }
        }
        //13.
        public static double squareRoot(double a) {

            return Math.sqrt(a);
        }

}
```