

```

import java.io.*;

public class AddNumArgs {

    //Global Variables for ARGS
    static long[] numLong = new long[2];
    static int[] numInt = new int[2];
    static double[] numDouble = new double[2];
    static float[] numFloat = new float[2];
    static String[] input = new String[2];
    static Boolean[] inputBoolean = new Boolean[2];
    static Boolean keyDouble = false;
    static Boolean keyLong = false;

    public long addNum (long numA, long numB) {
        return numA + numB;
    }

    public double divideNum (double numA, double numB) {
        return numA / numB;
    }

    public static void main(String args[])
    throws ArrayIndexOutOfBoundsException
    { //removed IOException so program would work
        try{ //ARGS[] required arguments
            int i=0;
            while (args[i] == "") {}
            //
        }
        catch (ArrayIndexOutOfBoundsException e1) {
            System.out.println ("\nYou must enter arguments, two numbers, to start the program.\n" +
                                "Please restart the program with arguments. Thank-you\n");
            System.exit(0);
        }
        try{ //Program will only accept two arguments
            int i=2;

```

AddNumArgs.java

```
while (args[i] == "" ) {}
System.out.println("\nYou have entered a third number." +
    "Please enter only two numbers as arguments. Thank-you.\n");
System.exit(0);
}
catch (ArrayIndexOutOfBoundsException e2) { //empty catch
}
try {
    input[0] = args[0];
    input[1] = args[1];
    System.out.println("\nCongrats ... I am using your ARGS.");
}
catch (ArrayIndexOutOfBoundsException e3) {
    System.out.println("\nA serious error has occurred in the algorithm.");
    System.exit(0);
}
try {
    System.out.println("Let's see if you typed whole numbers ...");
    numLong[0] = Long.parseLong(input[0]);
    numLong[1] = Long.parseLong(input[1]);
    keyLong = true;
    //numInt[0] = Integer.parseInt(input[0]);
}
catch (NumberFormatException e4) {
    try {
        System.out.println("Let's see if you typed decimals ...");
        numDouble[0] = Double.parseDouble(input[0]);
        numDouble[1] = Double.parseDouble(input[1]);
        keyDouble = true;
        //Float.parseFloat(input[0]);
    }
    catch (NumberFormatException e5) {
        System.out.println("Looks like you didn't type a number." +
            "You might have typed a Boolean ... let's see.");
        try {
            //anything typed will equal a false Boolean
            inputBoolean[0] = Boolean.parseBoolean(input[0]);
```

```

                                AddNumArgs.java
        inputBoolean[1] = Boolean.parseBoolean(input[1]);
        System.out.println("Looks like you typed two Booleans, " +
            "But we cannot do anything with them yet.");
        System.out.println("\nYour Booleans are: " + inputBoolean[0] + " & " + inputBoolean[1]);
    }
    catch (NumberFormatException e6) {
        System.out.println("Actually, it looks like you just typed Strings");
    }

}
} //Long-Int Try

AddNumArgs obj = new AddNumArgs();
//
//Need to check if args[] is empty
//Need to check if args[2] is empty, this is good
//Need to parse from strings to ... this becomes an algorithm
//
if (keyLong == true) {
    long ansSum = obj.addNum(numLong[0], numLong[1]);
    System.out.println("\nYour whole numbers are: " + numLong[0] + " & " + numLong[1]);
    System.out.println("\nSum of two numbers is: " + ansSum);
}
if (keyDouble == true) {
    double ansDiv = obj.divideNum(numDouble[0], numDouble[1]);
    System.out.println("\nYour decimal numbers are: " + numDouble[0] + " & " + numDouble[1]);
    System.out.println("\nDivision of two numbers is: " + ansDiv);
    System.out.printf("%.2f", ansDiv);
}
System.out.println("\n");
/*
try { //forcing division by zero, java.io has smrt response
    ansDiv = obj.divideNum(args[0], args[1]);
    //Execution will not continue if error in above line
    System.out.println("Excellent, you have not divided by zero");
} catch (Exception e) {
    System.out.println("Please do not divide by zero.\n Answer formatted to 108");
}

```

AddNumArgs.java

```
    ansDiv = 108.0; //infinity
    System.out.printf("%.2f", ansDiv);
}
*/
}
}
```