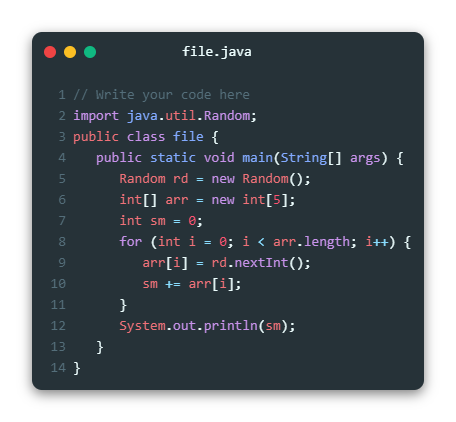
**Assignment 2**

**Use arrays to structure the raw data and to perform data comparison & operations**

Write a program which creates an integer array and displays sum of its elements.



# Use arrays to structure the raw data and to perform data comparison & operations

Write a program which performs addition of elements which are stored in two arrays of type double.

Arrays lengths may be variable in size. The resultant values must be stored in an integer array. Display the resultant integer array in a formatted way.

**Example:**

**Input:**

dInputArray1[]

10.0

20.0

30.0

dInputArray2[]

20.0

50.0

30.0

70.0

80.0

**Output:**

iSumArray[]

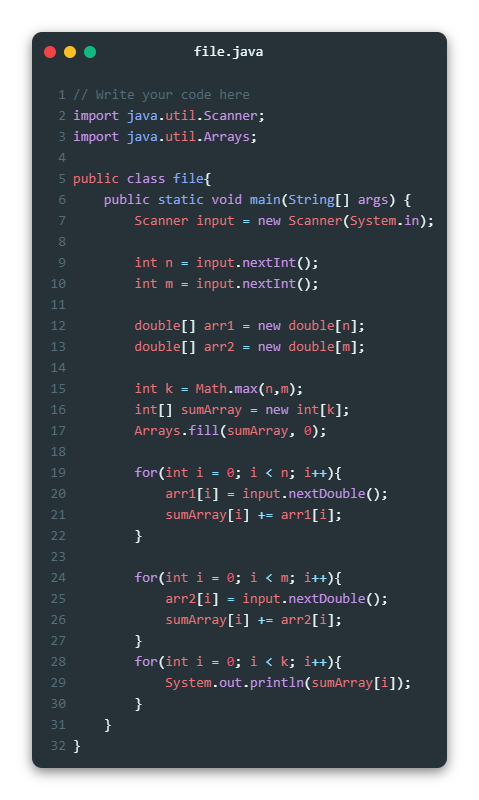
10

70

60

70

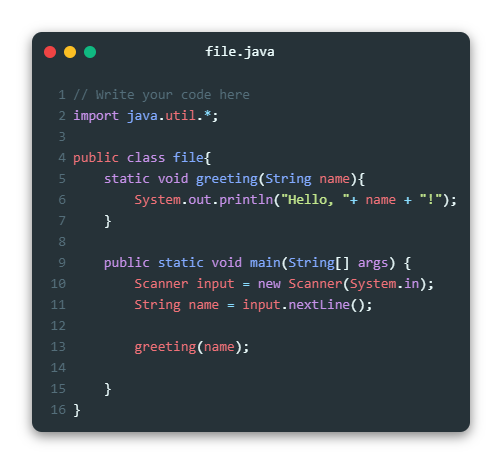
80



# Use arrays to structure the raw data and to perform data comparison & operations

Write a **method** that receives a name as **parameter** and prints on the console. “Hello, <name>!” Example

|  |  |
| --- | --- |
| **Input** | **Output** |
| Peter | Hello, Peter! |

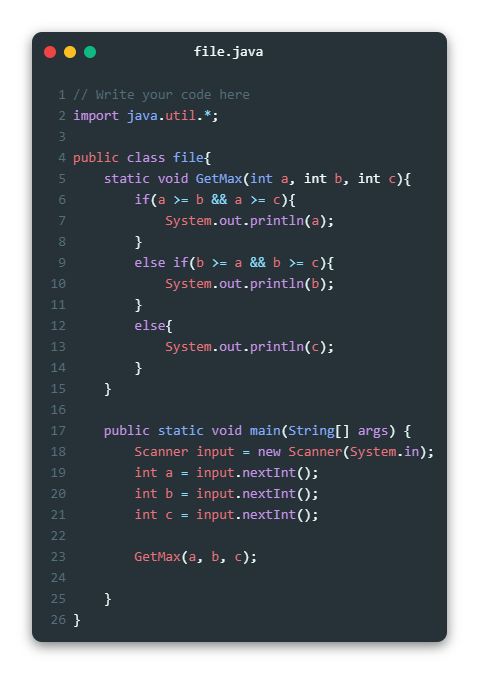


# Use arrays to structure the raw data and to perform data comparison & operations

Create a method **GetMax(int a, int b, int c)**, that returns maximal of three numbers. Write a program that reads three numbers from the console and prints the biggest of them.

**Examples**

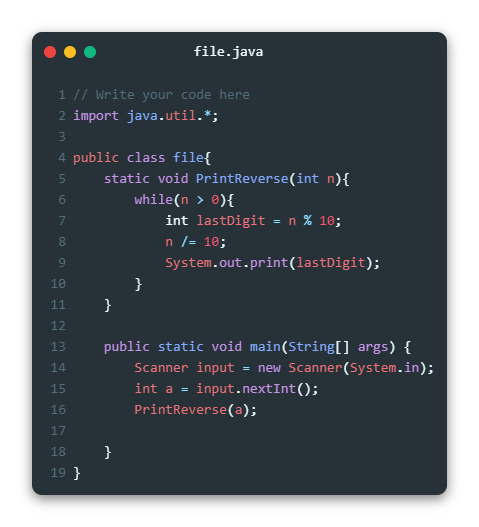
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Input** | **Output** |  | **Input** | **Output** |
| 1  2  3 | 3 | -100  -101  -102 | -100 |



# Use arrays to structure the raw data and to perform data comparison & operations

Write a method that **prints the digits** of a given decimal number in a **reversed order**. **Examples**

|  |  |  |
| --- | --- | --- |
| **Input** | **Output** |  |
| 256 | 652 |

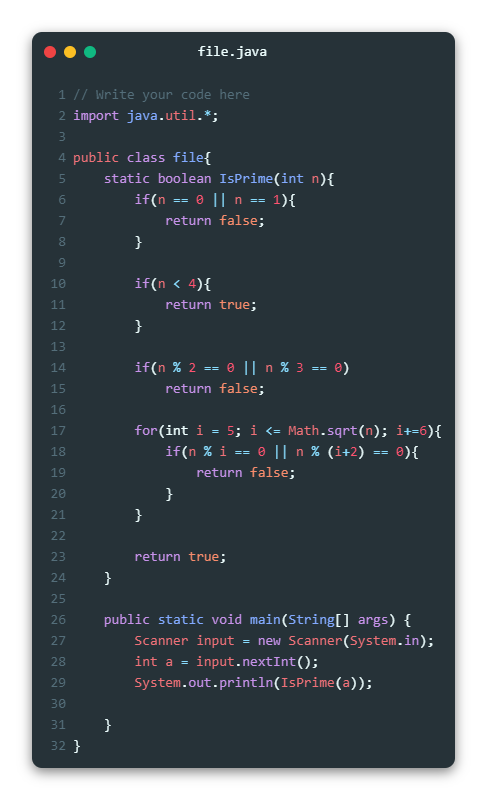


# Use arrays to structure the raw data and to perform data comparison & operations

Write a Boolean method **IsPrime(n)** that check whether a given integer number **n** is prime.

**Examples**:

|  |  |
| --- | --- |
| **n** | **IsPrime(n)** |
| 0 | false |
| 1 | false |
| 2 | true |
| 3 | true |
| 4 | false |
| 5 | true |
| 323 | false |
| 337 | true |
| 6737626471 | true |
| 117342557809 | false |



# Use arrays to structure the raw data and to perform data comparison & operations

Write a method that calculates **all prime numbers in given range** and returns them as list of integers

Write a method to **print a list of integers**. Write a program that takes two integer numbers (each at a separate line) and prints all primes in their range, separated by a comma.

**Examples**

|  |  |
| --- | --- |
| **Start and End**  **Number** | **Output** |
| 0  10 | 2, 3, 5, 7 |
| 5  11 | 5, 7, 11 |
| 100  200 | 101, 103, 107, 109, 113, 127, 131, 137, 139, 149, 151, 157, 163, 167,  173, 179, 181, 191, 193, 197, 199 |
| 250  950 | 251, 257, 263, 269, 271, 277, 281, 283, 293, 307, 311, 313, 317, 331, 337, 347, 349, 353, 359, 367, 373, 379, 383, 389, 397, 401, 409, 419,  421, 431, 433, 439, 443, 449, 457, 461, 463, 467, 479, 487, 491, 499,  503, 509, 521, 523, 541, 547, 557, 563, 569, 571, 577, 587, 593, 599,  601, 607, 613, 617, 619, 631, 641, 643, 647, 653, 659, 661, 673, 677,  683, 691, 701, 709, 719, 727, 733, 739, 743, 751, 757, 761, 769, 773,  787, 797, 809, 811, 821, 823, 827, 829, 839, 853, 857, 859, 863, 877,  881, 883, 887, 907, 911, 919, 929, 937, 941, 947 |
| 100  50 | *(empty list)* |



# Use arrays to structure the raw data and to perform data comparison & operations

Write a program that can **calculate the area** of **four different geometry figures** - triangle, square, rectangle and circle.

**On the first line** you will get the **figure type**. Next you will get parameters for the chosen figure, **each on a different line**:

* Triangle - side and height
* Square - side
* Rectangle - width and height
* Circle - radius

The output should be rounded to the second digit after the decimal point: **Examples**

|  |  |
| --- | --- |
| **Input** | **Output** |
| triangle  3  6 | 9.00 |
| rectangle  4  5 | 20.00 |



**Use arrays to structure the raw data and to perform data comparison & operations**

Write a method which accepts two integer arrays and returns an array of unique elements.

Example:

Array 1 = { 10, 5, 20, 15, 25, 30}

Array 2 = {50, 12, 5, 30, 15, 70}

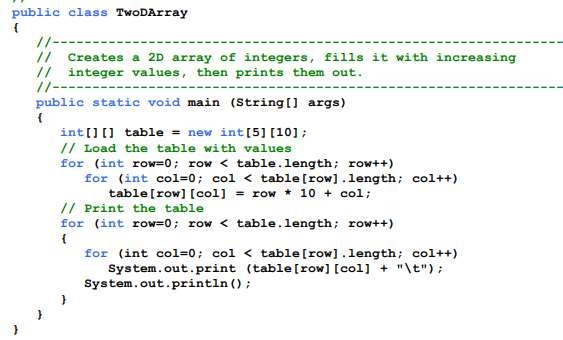
Result\_Array = {10, 20, 25, 50, 12, 70}

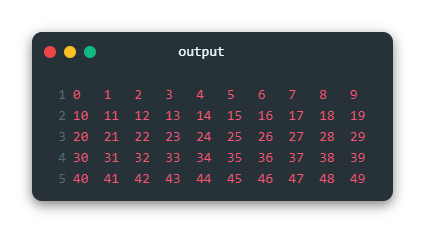
Int [] uniqElements(int array1[], int array2[]);



**Use arrays to structure the raw data and to perform data comparison & operations**

Analyze below given code and predict the output.





**Use arrays to structure the raw data and to perform data comparison & operations**

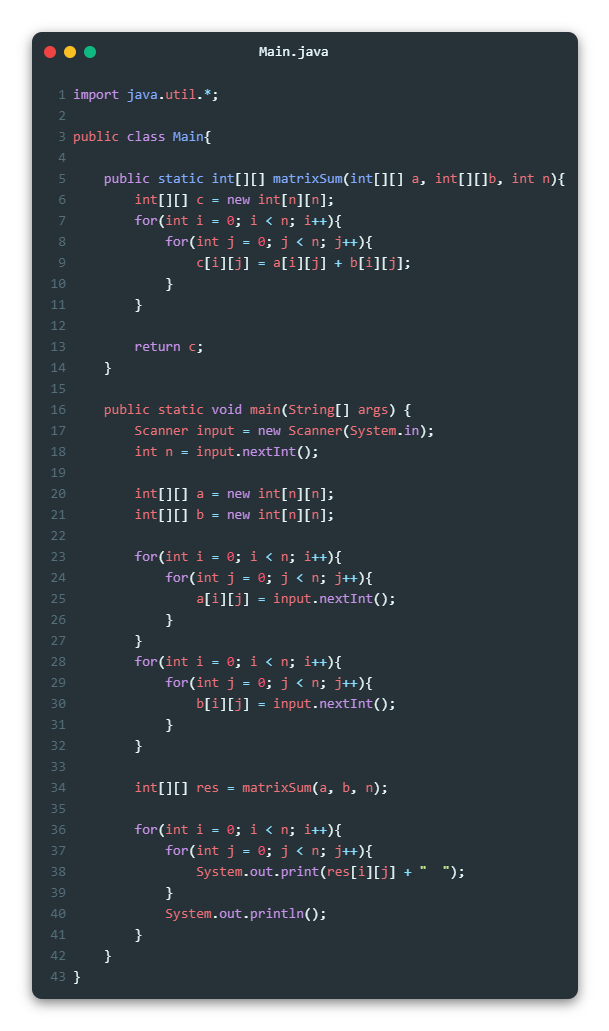
Write a method which accepts two matrices of Size N X N and returns summation of resultant Matrix.

Example:

Matrix A: [1,2,3] [4,5,6]

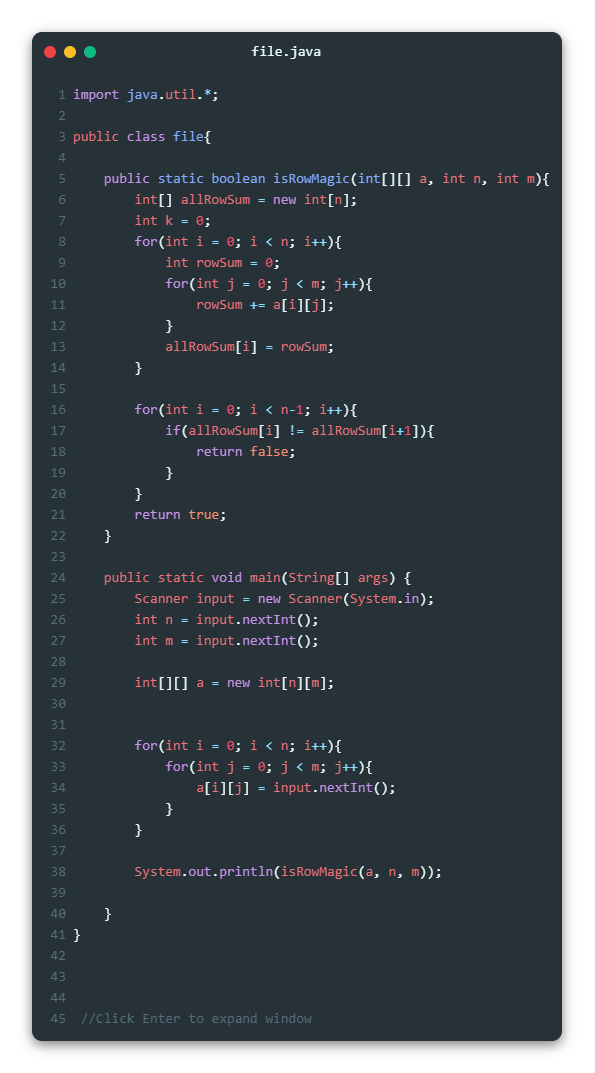
Matrix B: [4,5,6] [7,8,9]

Matrix C = A + B = [5,7,9] [11,13,15]

****

**Use arrays to structure the raw data and to perform data comparison & operations**

Write a method public static boolean isRowMagic(int[][] a) that checks if the array is row-magic (this means that every row has the same row sum).



**Use arrays to structure the raw data and to perform data comparison & operations**

Write a method public static boolean isMagic(int[][] a)

that checks if the array is a magic square. This means that it must be square, and that all row sums, all column sums, and the two diagonal-sums must all be equal.

