Duplicate values

```
package abst;
abstract class Duplicate1 {
	int[] numbers = { 1, 9, 3, 8, 5, 4, 6, 7, 4, 8, 2, 10, 10 };
	abstract public void duplicate();
}

class Duplicate2 extends Duplicate1{
	public void duplicate() {
		int l =numbers.length;
		 for (int i = 0; i < l-1; i++) {
			for (int j = i + 1; j < l; j++) {

			  if (numbers[i] == numbers[j]) {
			 System.out.println("Duplicate found: " +
numbers[i]);
				}
			}
		}

	}
}
public class Duplicate {
  public static void main(String[] args) {
	Duplicate2 d =new Duplicate2 ();
	d.duplicate();

    }

}
```

OUTPUT:

```
Duplicate found: 8
Duplicate found: 4
Duplicate found: 10
```

2.Add Two Matrix

```
public class Two {

    public static void main(String[] args) {


        int[][] matrix1 = {

            {1, 2, 3},

            {4, 5, 6},

            {7, 8, 9}

        };


        int[][] matrix2 = {

            {9, 8, 7},

            {6, 5, 4},

            {3, 2, 1}

        };

        int rows1 = matrix1.length;

        int columns1 = matrix1[0].length; // Get the number of columns in the first matrix

        int rows2 = matrix2.length;

        int columns2 = matrix2[0].length; // Get the number of columns in the second matrix
```

```
        if (rows1 != rows2 || columns1 != columns2) {
            System.out.println("Matrix dimensions do not match. Cannot perform addition.");
            return;
        }

        int[][] resultMatrix = new int[rows1][columns1]; // Use rows1 and columns1 to create the result matrix

        for (int i = 0; i < rows1; i++) {
            for (int j = 0; j < columns1; j++) {
                resultMatrix[i][j] = matrix1[i][j] + matrix2[i][j];
            }
        }

        System.out.println("Result Matrix (Matrix Addition):");
        for (int i = 0; i < rows1; i++) {
            for (int j = 0; j < columns1; j++) {
                System.out.print(resultMatrix[i][j] + " ");
            }
            System.out.println();
        }
```

```
    }

}
```

Output:

```
Matrix Addition:
10 16 16
13 10 17
28 14 12
```

3.Remove duplicate values

```
package abst;
public class Remove {
    public static void main(String[] args) {
        int[] originalArray = {1, 2, 2, 3, 4, 4, 5};
        int[] uniqueArray = removeDuplicates(originalArray);

        System.out.print("Original Array: ");
        printArray(originalArray);

        System.out.print("Array with Duplicates Removed: ");
        printArray(uniqueArray);
    }

    public static int[] removeDuplicates(int[] arr) {
        int n = arr.length;
        if (n == 0 || n == 1) {
            return arr;
        }

        int[] uniqueArray = new int[n];
        int uniqueCount = 0;

        for (int i = 0; i < n; i++) {
            boolean isDuplicate = false;
            for (int j = 0; j < i; j++) {
                if (arr[i] == arr[j]) {
                    isDuplicate = true;
                    break;
```

```
                }
            }
            if (!isDuplicate) {
                uniqueArray[uniqueCount] = arr[i];
                uniqueCount++;
            }
        }

        int[] result = new int[uniqueCount];
        System.arraycopy(uniqueArray, 0, result, 0, uniqueCount);
        return result;
    }

    public static void printArray(int[] arr) {
        for (int i : arr) {
            System.out.print(i + " ");
        }
        System.out.println();
    }
}
```

OUTPUT:

Original Array: 1 2 2 3 4 4 5
Array with Duplicates Removed: 1 2 3 4 5