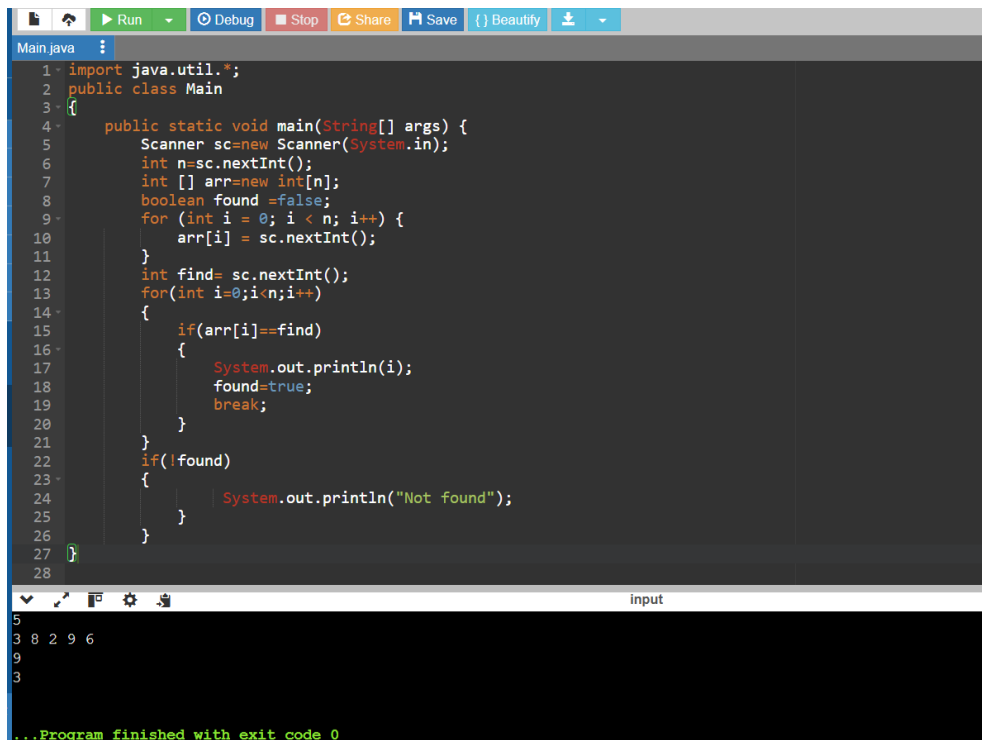


1. Search for an Element



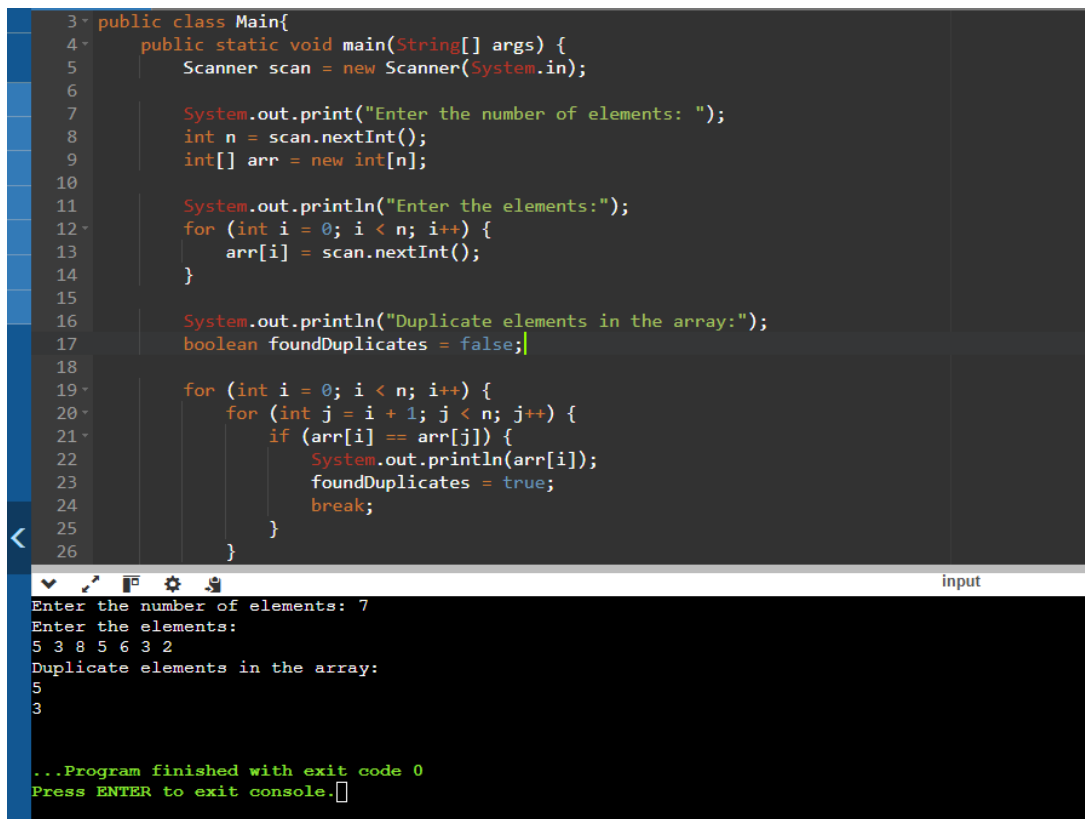
The screenshot shows a Java IDE with a file named 'Main.java'. The code implements a linear search algorithm. It imports `java.util.*`, defines a `Main` class, and has a `main` method. In the `main` method, it creates a `Scanner` object `sc` to read input from `System.in`. It prompts the user to enter the number of elements `n` and then reads `n` integers into an array `arr`. Next, it prompts for the element to be found (`find`) and iterates through the array to check if `find` is present. If found, it prints the index `i` and sets `found` to `true`. If the loop ends without finding the element, it prints "Not found". The console output shows the input sequence: 5, 8, 2, 9, 6, followed by the search value 3, and the output "Not found". The program finishes with exit code 0.

```
1 import java.util.*;
2 public class Main
3 {
4     public static void main(String[] args) {
5         Scanner sc=new Scanner(System.in);
6         int n=sc.nextInt();
7         int [] arr=new int[n];
8         boolean found =false;
9         for (int i = 0; i < n; i++) {
10             arr[i] = sc.nextInt();
11         }
12         int find= sc.nextInt();
13         for(int i=0;i<n;i++)
14         {
15             if(arr[i]==find)
16             {
17                 System.out.println(i);
18                 found=true;
19                 break;
20             }
21         }
22         if(!found)
23         {
24             System.out.println("Not found");
25         }
26     }
27 }
28
```

input

5
3 8 2 9 6
3
...Program finished with exit code 0

2.Duplicates



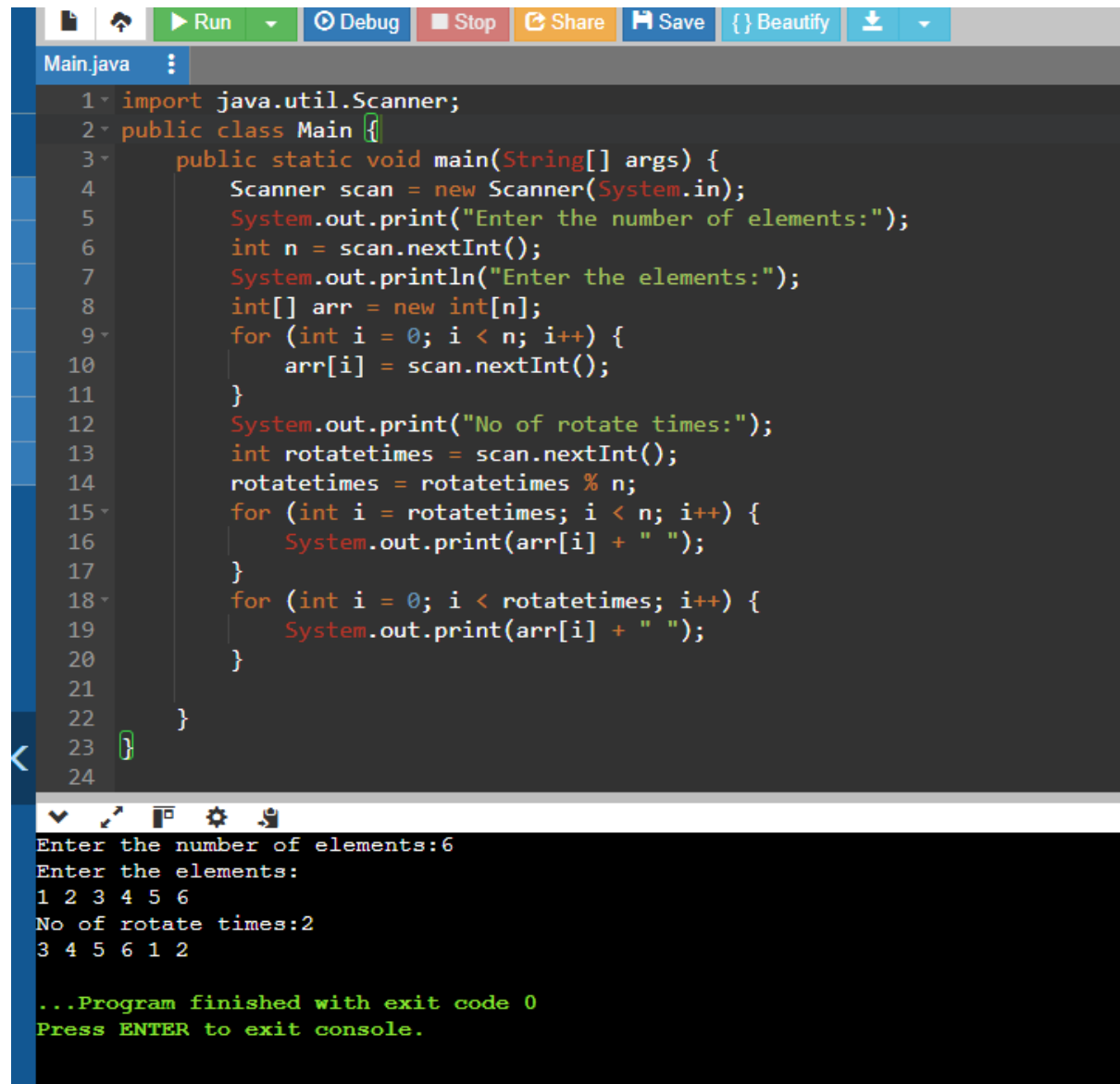
The screenshot shows a Java IDE with a file named 'Main.java'. The code implements an algorithm to find duplicate elements in an array. It uses a `Scanner` object `scan` to read input. It prompts the user to enter the number of elements `n` and then reads `n` integers into an array `arr`. It then iterates through the array using two nested loops: the outer loop iterates over each element `arr[i]`, and the inner loop iterates over all subsequent elements `arr[j]` (where `j > i`). If a duplicate is found (`arr[i] == arr[j]`), it prints the duplicate value and sets `foundDuplicates` to `true`. The console output shows the input sequence: 5, 3, 8, 5, 6, 3, 2, followed by the output "Duplicate elements in the array: 5" and "3". The program finishes with exit code 0.

```
3 public class Main{
4     public static void main(String[] args) {
5         Scanner scan = new Scanner(System.in);
6
7         System.out.print("Enter the number of elements: ");
8         int n = scan.nextInt();
9         int[] arr = new int[n];
10
11         System.out.println("Enter the elements:");
12         for (int i = 0; i < n; i++) {
13             arr[i] = scan.nextInt();
14         }
15
16         System.out.println("Duplicate elements in the array:");
17         boolean foundDuplicates = false;
18
19         for (int i = 0; i < n; i++) {
20             for (int j = i + 1; j < n; j++) {
21                 if (arr[i] == arr[j]) {
22                     System.out.println(arr[i]);
23                     foundDuplicates = true;
24                     break;
25                 }
26             }
27         }
28     }
29 }
```

input

Enter the number of elements: 7
Enter the elements:
5 3 8 5 6 3 2
Duplicate elements in the array:
5
3
...Program finished with exit code 0
Press ENTER to exit console.

Left rotate



The image shows a Java IDE with a code editor and a console window. The code editor displays a Java program for left rotating an array. The console window shows the program's execution, including user input and the resulting array after rotation.

```
Main.java
1 import java.util.Scanner;
2 public class Main {
3     public static void main(String[] args) {
4         Scanner scan = new Scanner(System.in);
5         System.out.print("Enter the number of elements:");
6         int n = scan.nextInt();
7         System.out.println("Enter the elements:");
8         int[] arr = new int[n];
9         for (int i = 0; i < n; i++) {
10             arr[i] = scan.nextInt();
11         }
12         System.out.print("No of rotate times:");
13         int rotatetimes = scan.nextInt();
14         rotatetimes = rotatetimes % n;
15         for (int i = rotatetimes; i < n; i++) {
16             System.out.print(arr[i] + " ");
17         }
18         for (int i = 0; i < rotatetimes; i++) {
19             System.out.print(arr[i] + " ");
20         }
21     }
22 }
23
24
```

Enter the number of elements:6
Enter the elements:
1 2 3 4 5 6
No of rotate times:2
3 4 5 6 1 2
...Program finished with exit code 0
Press ENTER to exit console.