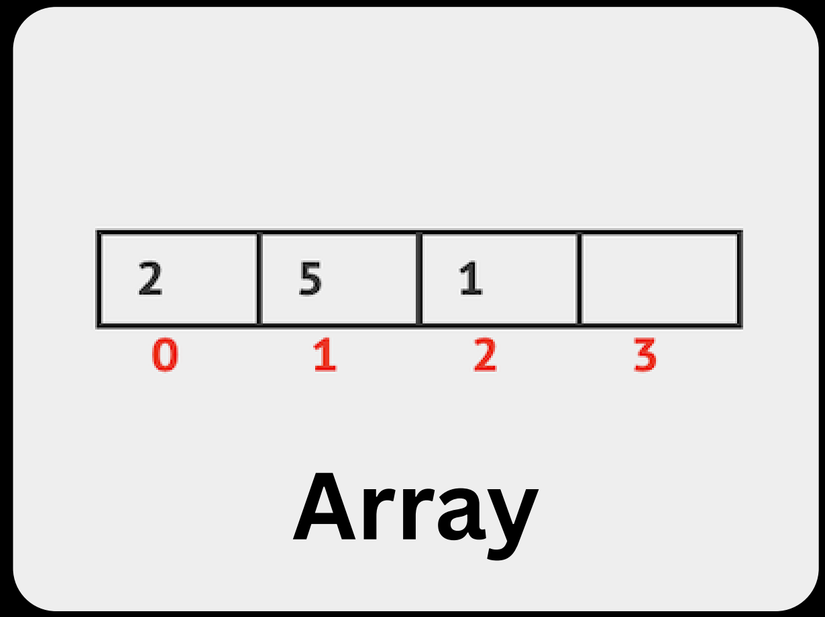
Welcome To
Tech By WebCoder

DSA Series

Topic : 1

Operation In Array

1. **Insert**
2. **Remove**
3. **Select**
4. **Update**



#techbywebcoder

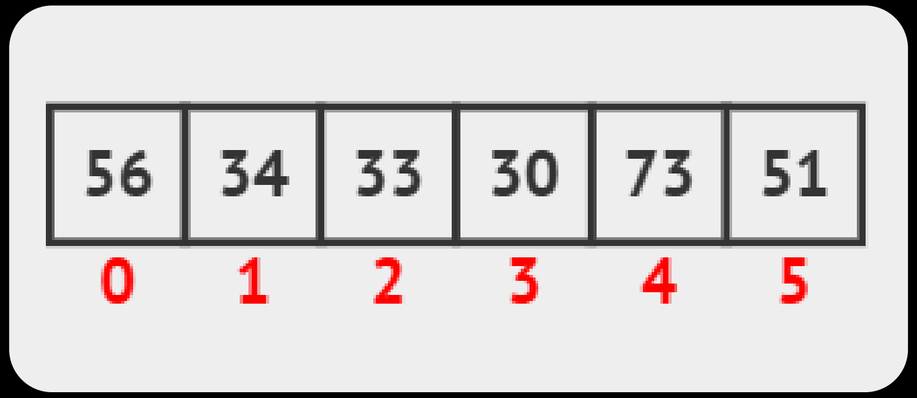
1.

INSERT

Operation In Array

#techbywebcoder

INSERT Operation In Array



Insert 30 to index 3 of array

Algorithm

Step 1: Start

Step 2: Input value to insert

Step 3: Input position where the value should be inserted

Step 4: If position < 0 OR position > size OR array is full

Print "Invalid position" and stop

Step 5: For i = size−1 down to position

Move arr[i] to arr[i + 1]

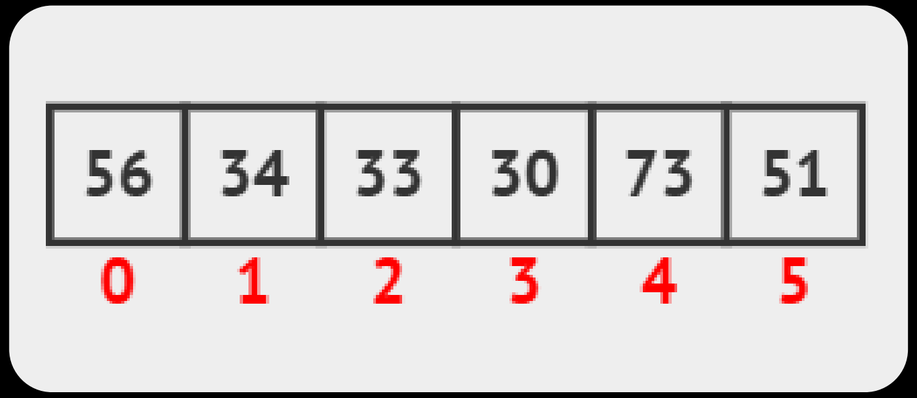
Step 6: Insert value at arr[position]

Step 7: Increase size by 1

Step 8: Stop

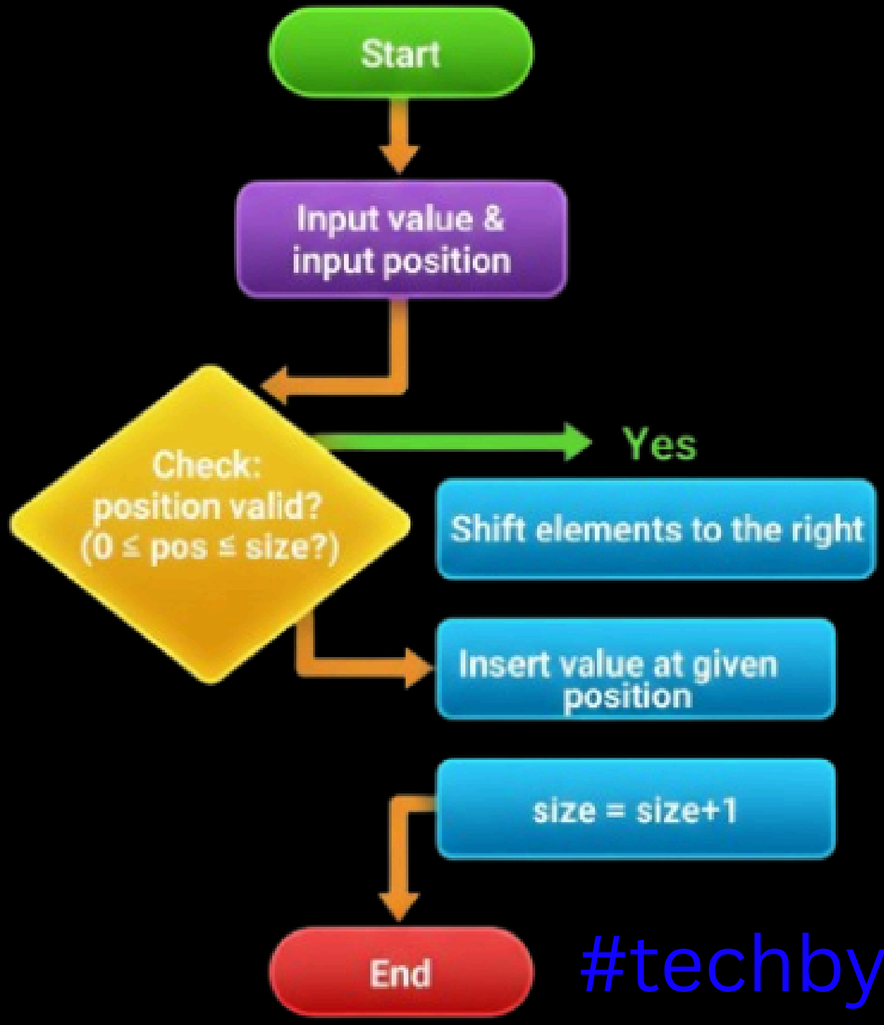
#techbywebcoder

INSERT Operation In Array



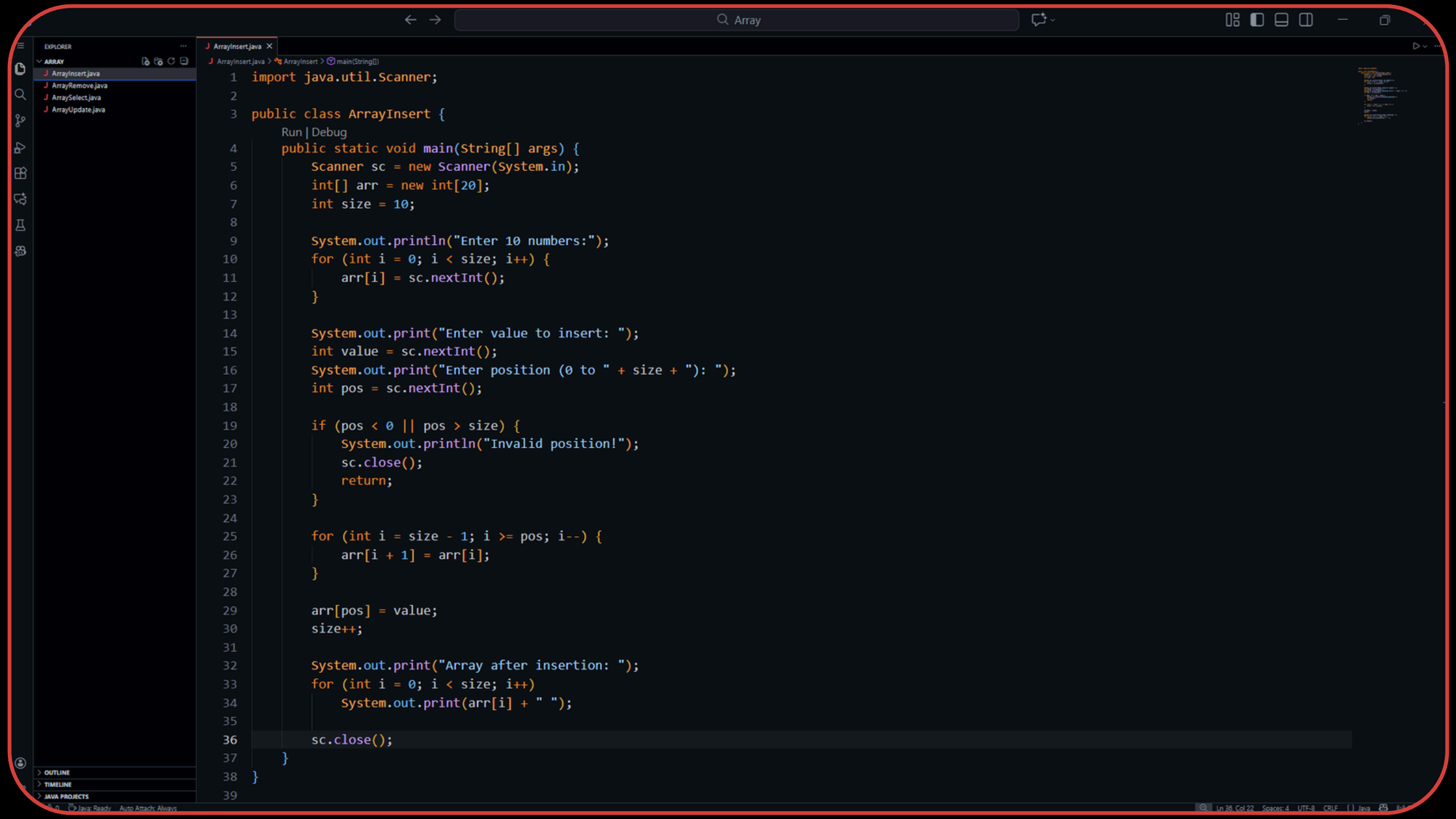
Insert 30 to index 3 of array

Flowchart

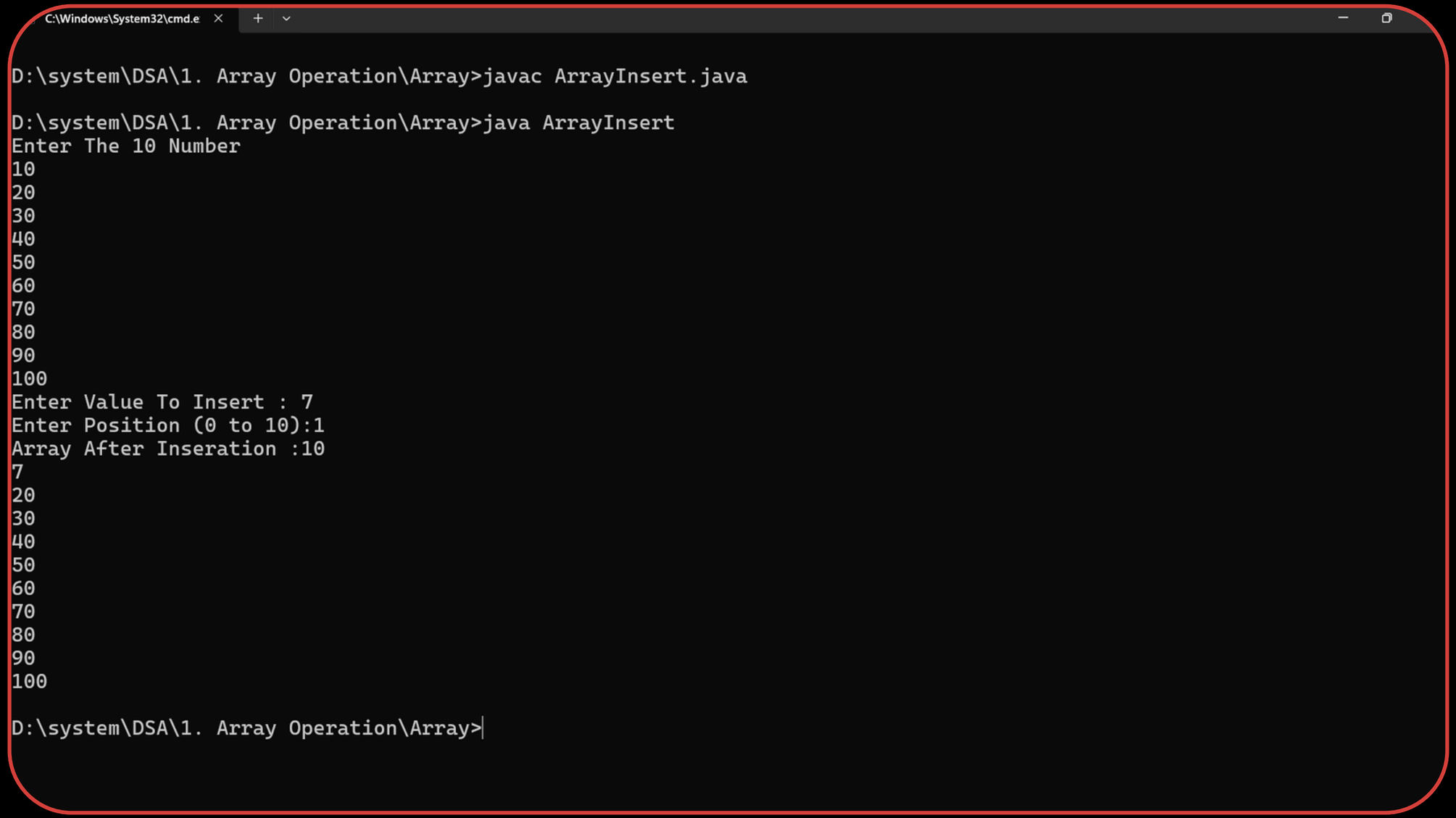


#techbywebcoder

Program


Array
EXPLORER
ARRAY
ArrayInsert.java
ArrayRemove.java
ArraySelect.java
ArrayUpdate.java
OUTLINE
TIMELINE
JAVA PROJECTS
ArrayInsert.java
ArrayInsert.java > ArrayInsert > main(String[])
Run | Debug
import java.util.Scanner;
public class ArrayInsert {
public static void main(String[] args) {
Scanner sc = new Scanner(System.in);
int[] arr = new int[20];
int size = 10;
System.out.println("Enter 10 numbers:");
for (int i = 0; i < size; i++) {
arr[i] = sc.nextInt();
}
System.out.print("Enter value to insert: ");
int value = sc.nextInt();
System.out.print("Enter position (0 to " + size + "): ");
int pos = sc.nextInt();
if (pos < 0 || pos > size) {
System.out.println("Invalid position!");
sc.close();
return;
}
for (int i = size - 1; i >= pos; i--) {
arr[i + 1] = arr[i];
}
arr[pos] = value;
size++;
System.out.print("Array after insertion: ");
for (int i = 0; i < size; i++)
System.out.print(arr[i] + " ");
sc.close();
}
}
Java: Ready
Auto Attach: Always
Ln 36, Col 22
Spaces: 4
UTF-8
CRLF
Java

Output


C:\Windows\System32\cmd.e
D:\system\DSA\1. Array Operation\Array>javac ArrayInsert.java
D:\system\DSA\1. Array Operation\Array>java ArrayInsert
Enter The 10 Number
10
20
30
40
50
60
70
80
90
100
Enter Value To Insert : 7
Enter Position (0 to 10):1
Array After Inseration :10
7
20
30
40
50
60
70
80
90
100
D:\system\DSA\1. Array Operation\Array>

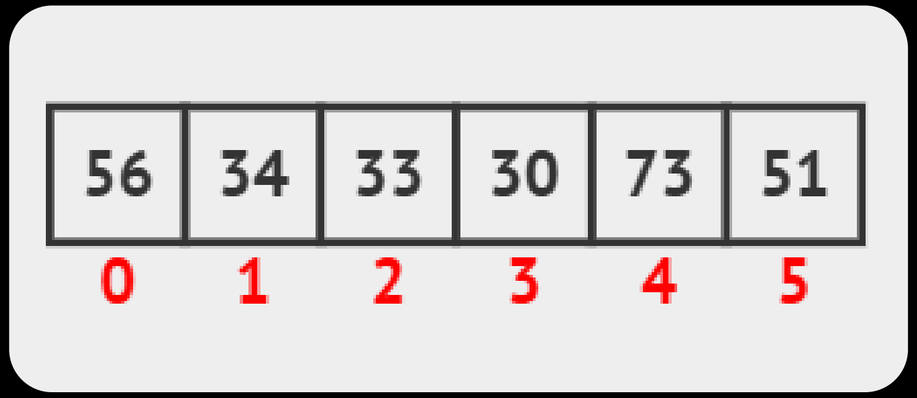
2.

REMOVE

Operation In Array

#techbywebcoder

REMOVE Operation In Array



Remove array[i] from array

Algorithm

Step 1: Start

Step 2: Input position to remove

Step 3: If position < 0 OR position ≥ size

Print "Invalid position" and stop

Step 4: For i = position to size−2

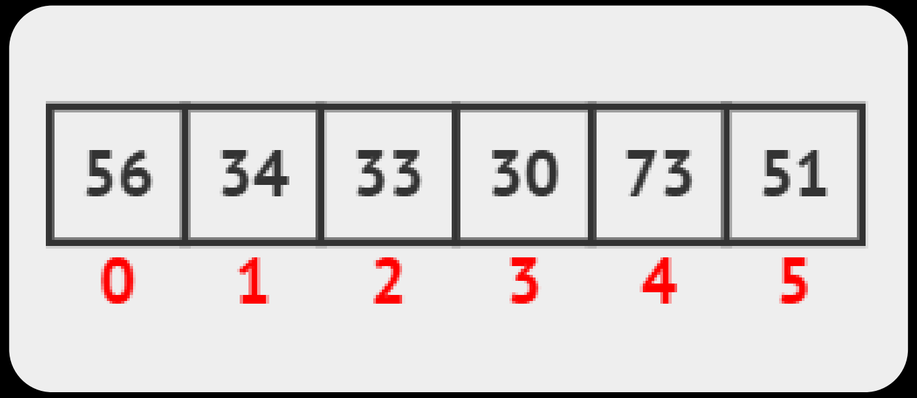
Move arr[i + 1] to arr[i]

Step 5: Decrease size by 1

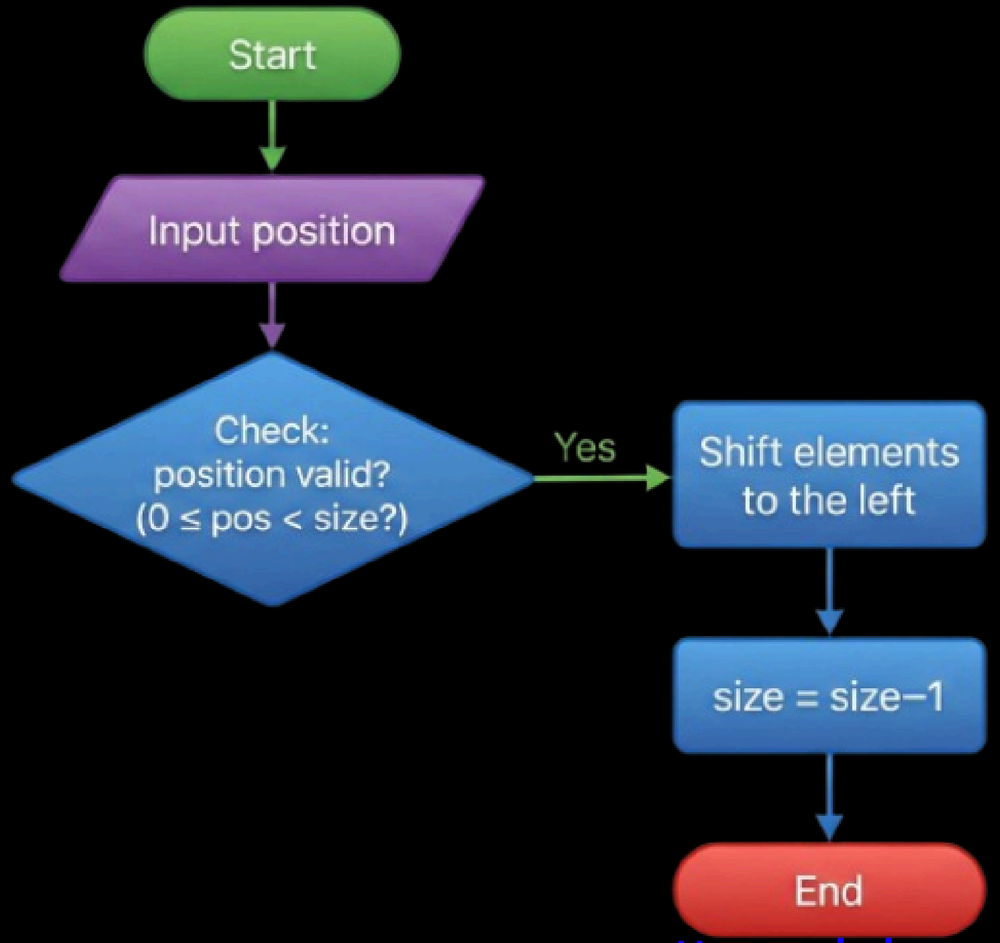
Step 6: Stop

#techbywebcoder

REMOVE Operation In Array



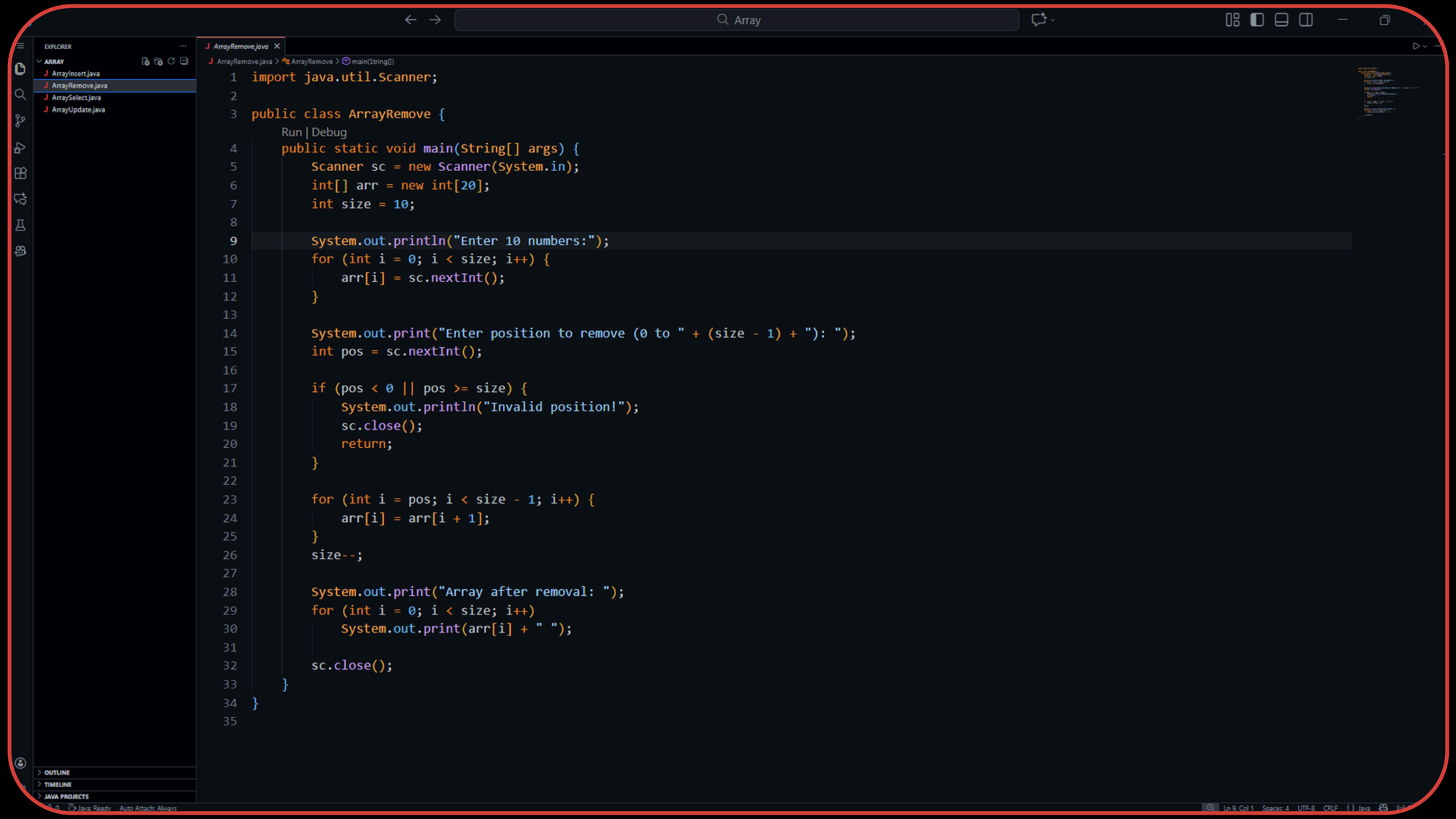
Flowchart



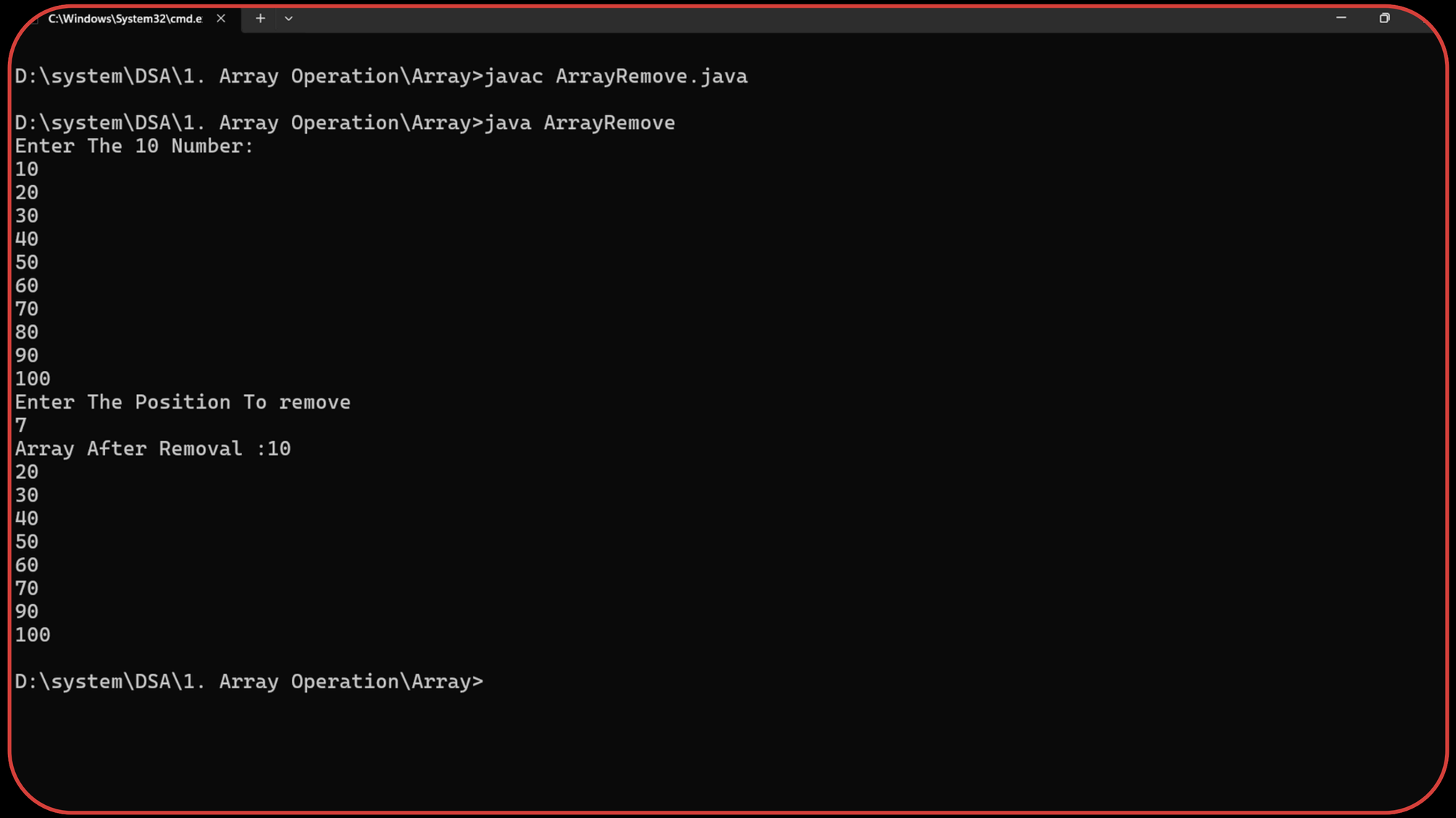
Remove array[i] from array

#techbywebcoder

Program


Array
EXPLORER
ARRAY
ArrayInsert.java
ArrayRemove.java
ArraySelect.java
ArrayUpdate.java
ArrayRemove.java
ArrayRemove.java > ArrayRemove > main(String[])
Run | Debug
import java.util.Scanner;
public class ArrayRemove {
public static void main(String[] args) {
Scanner sc = new Scanner(System.in);
int[] arr = new int[20];
int size = 10;
System.out.println("Enter 10 numbers:");
for (int i = 0; i < size; i++) {
arr[i] = sc.nextInt();
}
System.out.print("Enter position to remove (0 to " + (size - 1) + "): ");
int pos = sc.nextInt();
if (pos < 0 || pos >= size) {
System.out.println("Invalid position!");
sc.close();
return;
}
for (int i = pos; i < size - 1; i++) {
arr[i] = arr[i + 1];
}
size--;
System.out.print("Array after removal: ");
for (int i = 0; i < size; i++)
System.out.print(arr[i] + " ");
sc.close();
}
}
OUTLINE
TIMELINE
JAVA PROJECTS

Output


C:\Windows\System32\cmd.e
D:\system\DSA\1. Array Operation\Array>javac ArrayRemove.java
D:\system\DSA\1. Array Operation\Array>java ArrayRemove
Enter The 10 Number:
10
20
30
40
50
60
70
80
90
100
Enter The Position To remove
7
Array After Removal :10
20
30
40
50
60
70
90
100
D:\system\DSA\1. Array Operation\Array>

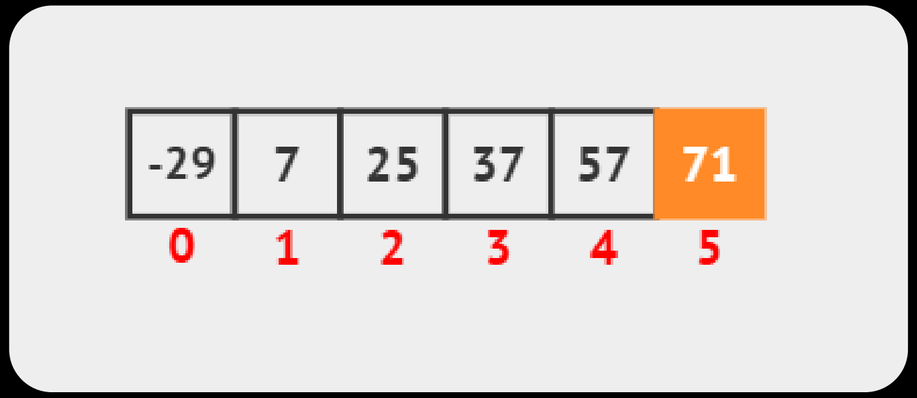
3.

SELECT

Operation In Array

#techbywebcoder

SELECT Operation In Array



Select 5 rank item in sorted array

Algorithm

Step 1: Start

Step 2: Input value to search

Step 3: For i = 0 to size–1

If arr[i] = value

Print "Value found at index i" and stop

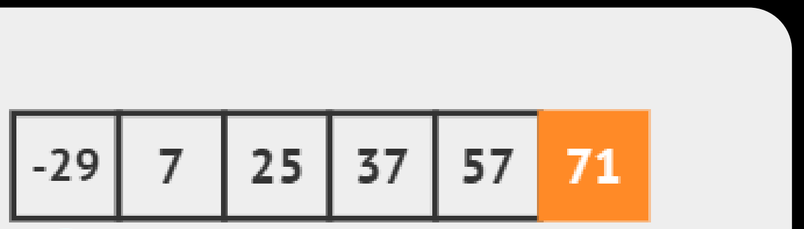
Step 4: If loop ends, print "Value not found"

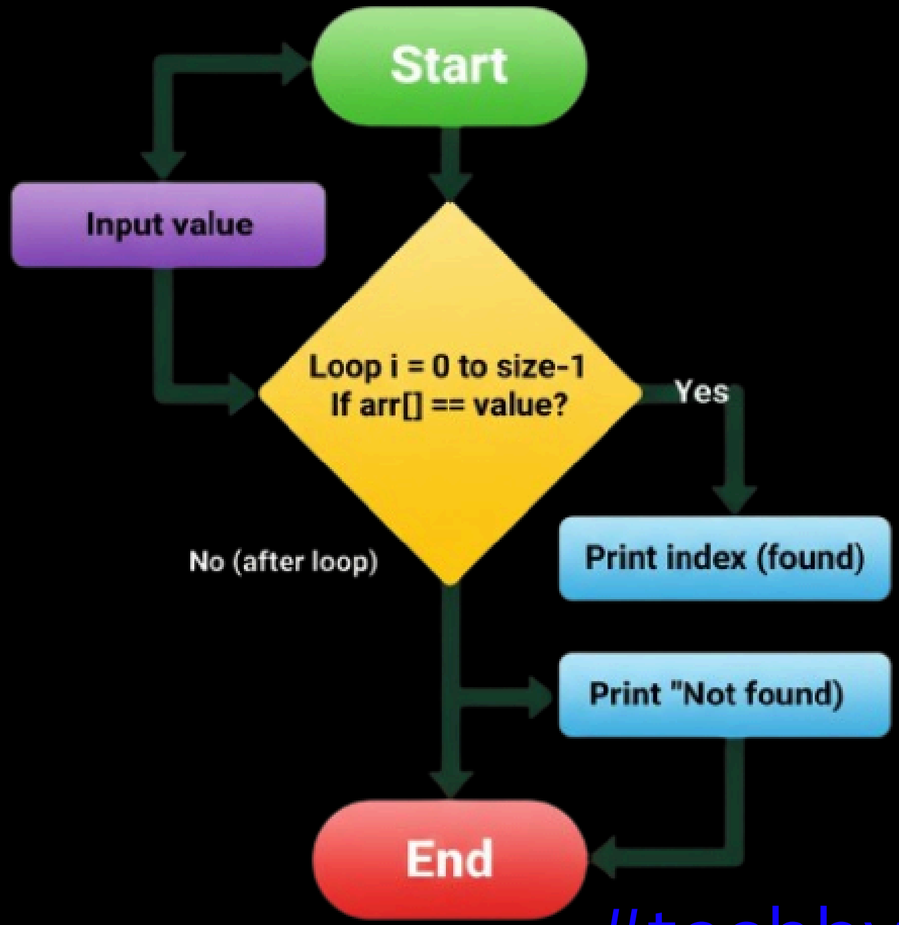
Step 5: Stop

#techbywebcoder

SELECT Operation In Array

Flowchart

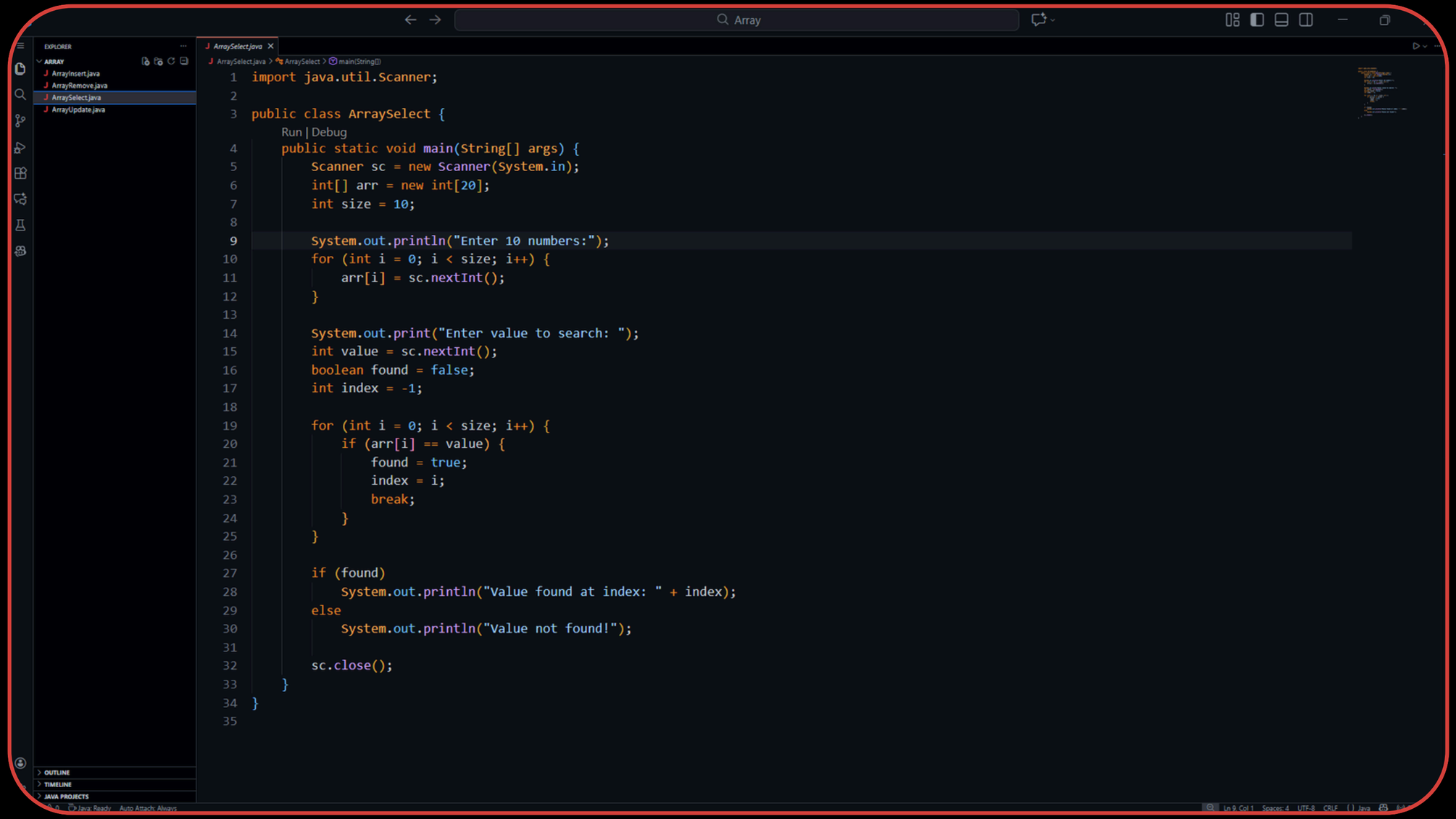




Select 5 rank item in sorted array

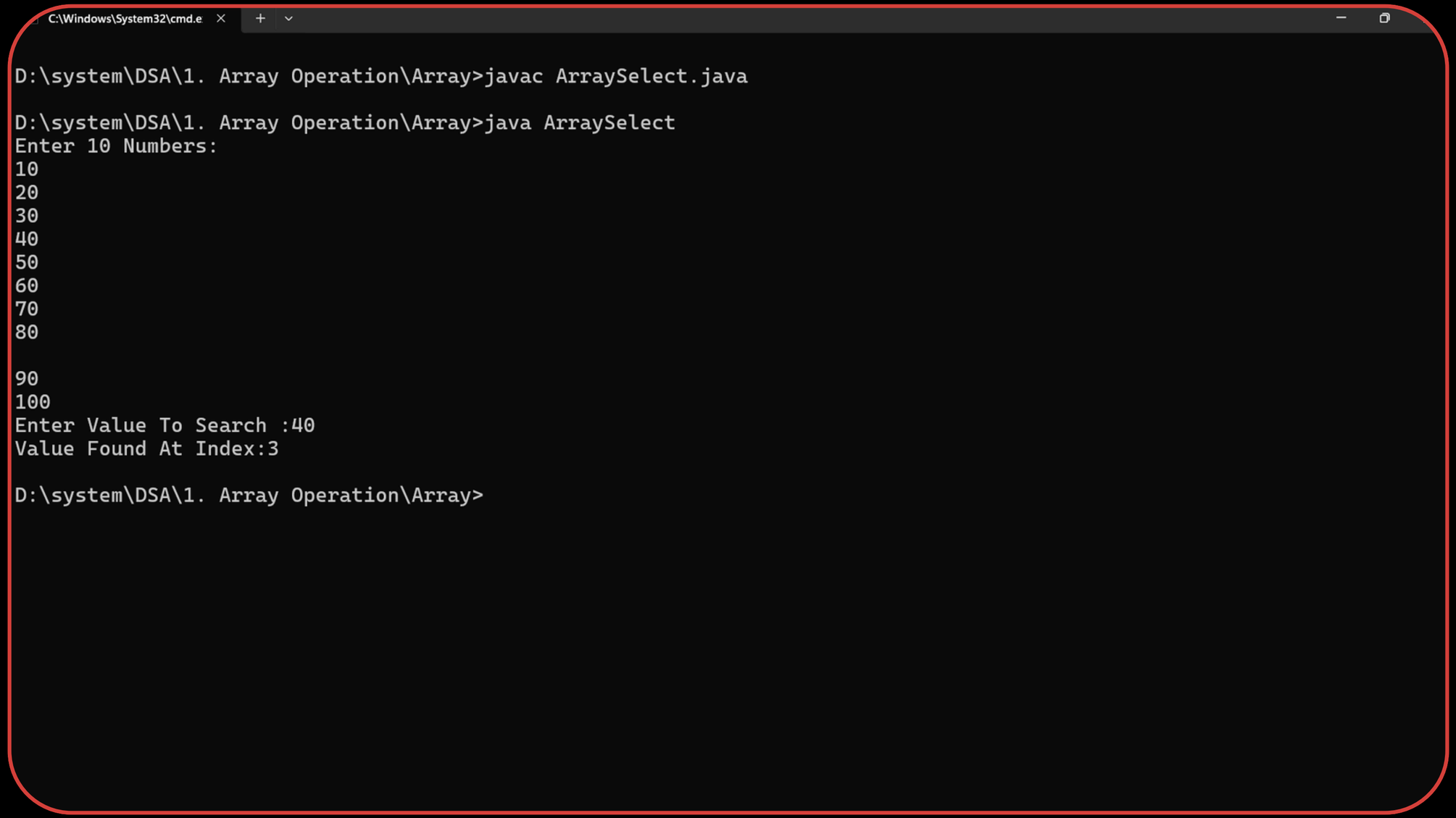
#techbywebcoder

Program




```java
import java.util.Scanner;

public class ArraySelect {
    Run | Debug
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int[] arr = new int[20];
        int size = 10;

        System.out.println("Enter 10 numbers:");
        for (int i = 0; i < size; i++) {
            arr[i] = sc.nextInt();
        }

        System.out.print("Enter value to search: ");
        int value = sc.nextInt();
        boolean found = false;
        int index = -1;

        for (int i = 0; i < size; i++) {
            if (arr[i] == value) {
                found = true;
                index = i;
                break;
            }
        }

        if (found)
            System.out.println("Value found at index: " + index);
        else
            System.out.println("Value not found!");

        sc.close();
    }
}
```

Output


C:\Windows\System32\cmd.e
D:\system\DSA\1. Array Operation\Array>javac ArraySelect.java
D:\system\DSA\1. Array Operation\Array>java ArraySelect
Enter 10 Numbers:
10
20
30
40
50
60
70
80
90
100
Enter Value To Search :40
Value Found At Index:3
D:\system\DSA\1. Array Operation\Array>

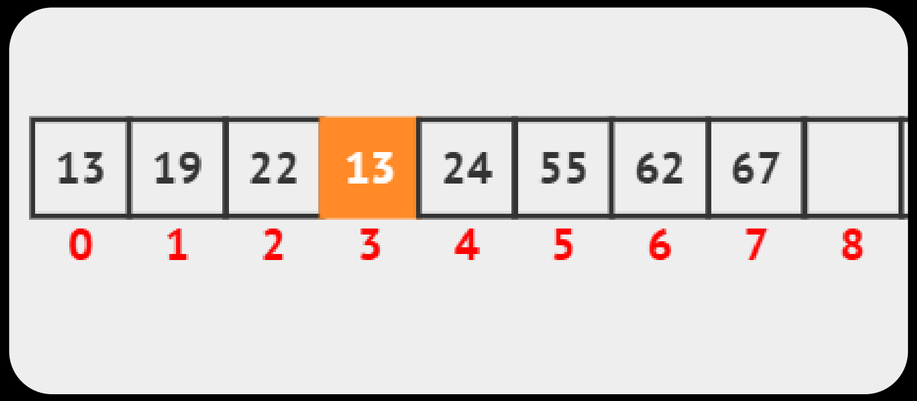
4.

UPDATE

Operation In Array

#techbywebcoder

UPDATE Operation In Array



Update array element at chosen index

Algorithm

Step 1: Start

Step 2: Input position to update

Step 3: If position < 0 OR position ≥ size

Print "Invalid position" and stop

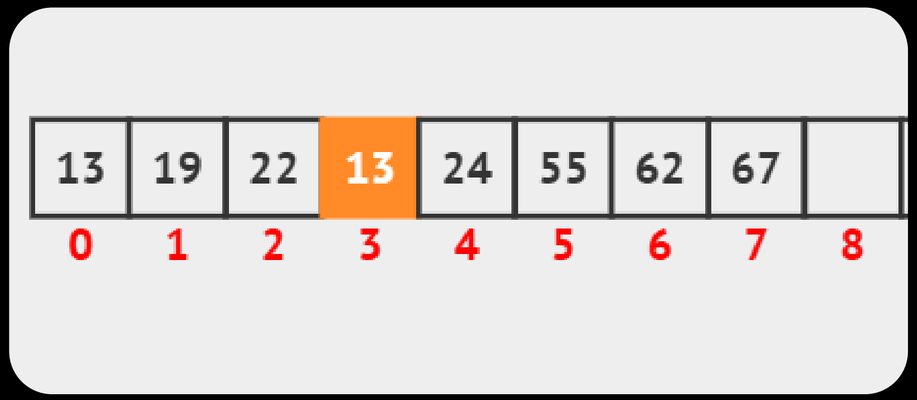
Step 4: Input new value

Step 5: Set arr[position] = new value

Step 6: Stop

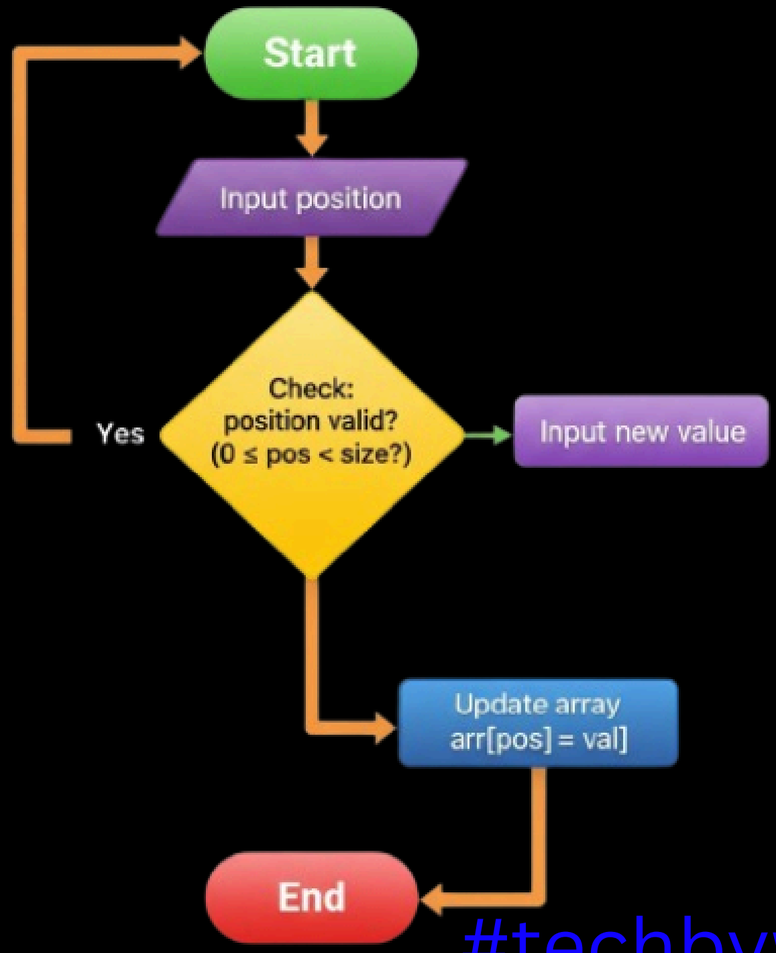
#techbywebcoder

UPDATE Operation In Array



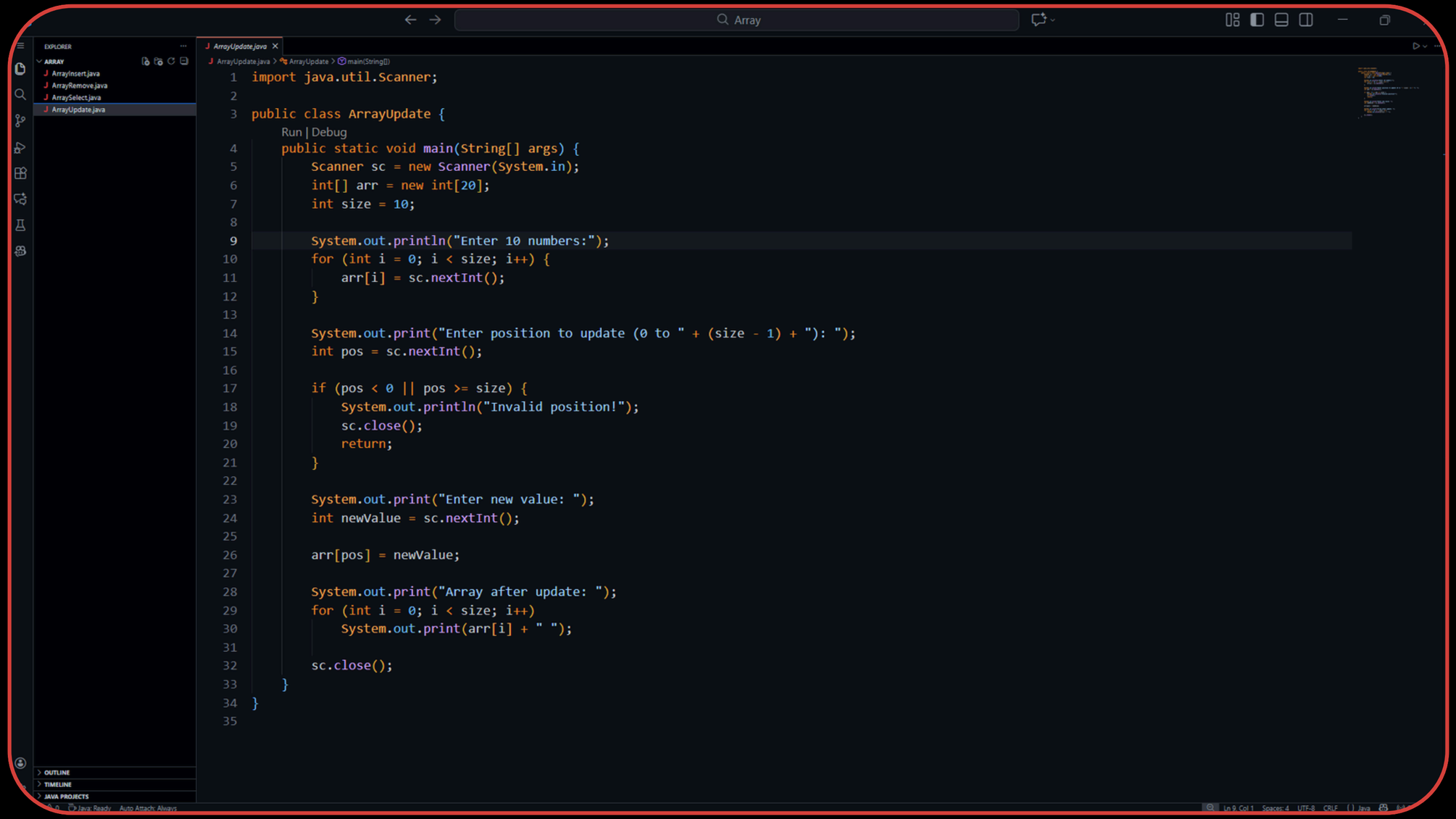
Update array element at chosen index

Flowchart

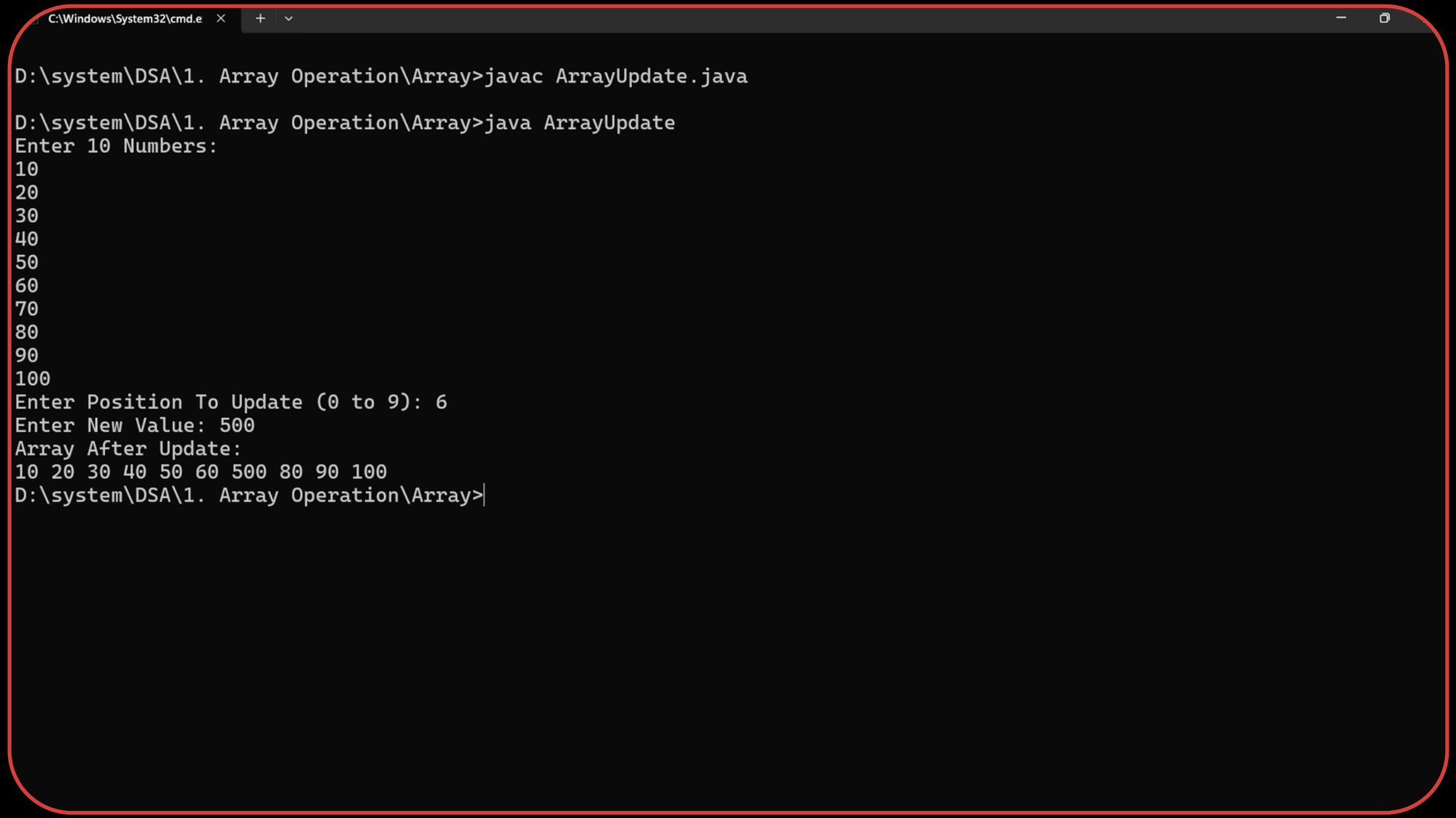


#techbywebcoder

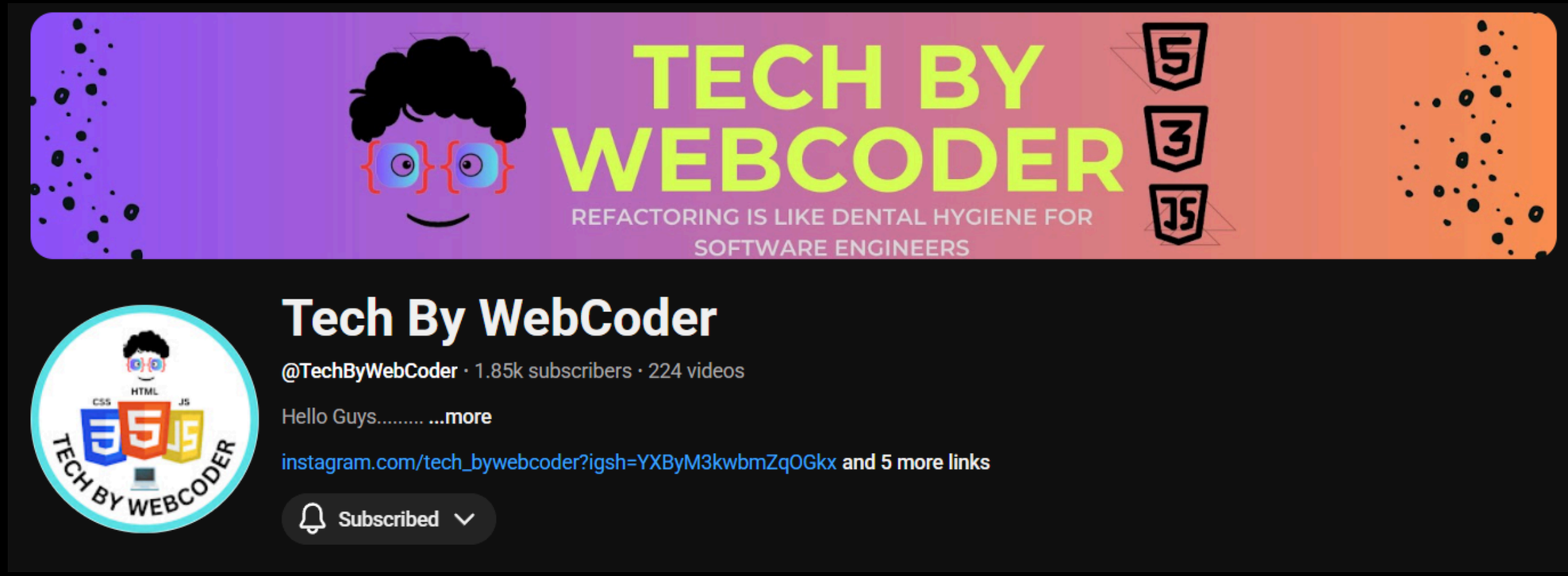
Program


Array
EXPLORER
ARRAY
ArrayInsert.java
ArrayRemove.java
ArraySelect.java
ArrayUpdate.java
ArrayUpdate.java
ArrayUpdate.java > ArrayUpdate > main(String[])
1 import java.util.Scanner;
2
3 public class ArrayUpdate {
Run | Debug
4 public static void main(String[] args) {
5 Scanner sc = new Scanner(System.in);
6 int[] arr = new int[20];
7 int size = 10;
8
9 System.out.println("Enter 10 numbers:");
10 for (int i = 0; i < size; i++) {
11 arr[i] = sc.nextInt();
12 }
13
14 System.out.print("Enter position to update (0 to " + (size - 1) + "): ");
15 int pos = sc.nextInt();
16
17 if (pos < 0 || pos >= size) {
18 System.out.println("Invalid position!");
19 sc.close();
20 return;
21 }
22
23 System.out.print("Enter new value: ");
24 int newValue = sc.nextInt();
25
26 arr[pos] = newValue;
27
28 System.out.print("Array after update: ");
29 for (int i = 0; i < size; i++)
30 System.out.print(arr[i] + " ");
31
32 sc.close();
33 }
34 }
35
OUTLINE
TIMELINE
JAVA PROJECTS
Java: Ready
Auto Attach: Always
Ln 9, Col 1
Spaces: 4
UTF-8
CRLF
Java

Output


C:\Windows\System32\cmd.e
D:\system\DSA\1. Array Operation\Array>javac ArrayUpdate.java
D:\system\DSA\1. Array Operation\Array>java ArrayUpdate
Enter 10 Numbers:
10
20
30
40
50
60
70
80
90
100
Enter Position To Update (0 to 9): 6
Enter New Value: 500
Array After Update:
10 20 30 40 50 60 500 80 90 100
D:\system\DSA\1. Array Operation\Array>

Thank you



Subscribe For More !