Assignments:

1. Write a simple program to implement ternary operator
2. Linear Search - Find the given number is available from array of numbers and it's position
3. Write a program to reverse of Array
4. Write a program to find the slice of array
5. Write a program to update an element
6. Write a program to delete the element
7. Write a program to insert the element

1. Write a simple program to implement ternary operator

Solution:

```
import java.util.Scanner;
public class TernaryOperator {
   public static void main(String[] args) {
      Scanner sc = new Scanner(System.in);
      System.out.println("Enter a no");
      int n = sc.nextInt();
      String result = (n % 2 == 0) ? "Even" : "Odd";
      System.out.println("The number " + n + " is " + result);
      sc.close();
   }
}
```

2. Linear Search - Find the given number is available from array of numbers and it's position

Solution:

```
import java.util.*;
public class LinearSearch {
   public static int linearSearch(int arr[], int key) {
      for (int i = 0; i < arr.length; i++) {
         if (arr[i] == key) {
            return (i + 1);
         }
      }
      return -1;
   }
   public static void main(String[] args) {
      Scanner sc = new Scanner(System.in);
      System.out.println("Enter the length of array");
      int n = sc.nextInt();
      int arr[] = new int[n];
      System.out.println("Enter the Elements of array");
      for (int i = 0; i < arr.length; i++) {
         arr[i] = sc.nextInt();
      }
      System.out.println("Enter the key");
      int key = sc.nextInt();
      int res = linearSearch(arr, key);
      if (res != -1) {
         System.out.println("Key found at position " + res);
      } else {
         System.out.println("Key not found");
      }
      sc.close(); }}
```

3. Write a program to reverse of Array

Solution:

```
import java.util.*;
public class ArrayReverse {
   public static void reverse(int arr[]) {
      int first = 0, last = arr.length - 1;
      while (first < last) {
         int temp = arr[last];
         arr[last] = arr[first];
         arr[first] = temp;
         first++;
         last--;
      }
   }
   public static void main(String[] args) {
      Scanner sc = new Scanner(System.in);
      System.out.println("Enter legth of array");
      int n = sc.nextInt();
      int arr[] = new int[n];
      System.out.println("Enter elements of array");
      for (int i = 0; i < arr.length; i++) {
         arr[i] = sc.nextInt();
      }
      System.out.println("before reverse");
      for (int i = 0; i <= arr.length - 1; i++) {
         System.out.print(arr[i] + " ");
      }
      System.out.println();
      reverse(arr);
      System.out.println("after reverse");
      for (int i = 0; i <= arr.length - 1; i++) {
         System.out.print(arr[i] + " ");
      }
      sc.close();
   }}
```

4. Write a program to find the slice of array

Solution:

```
import java.util.*;
public class ArraySlice {
   public static void main(String[] args) {
      Scanner sc = new Scanner(System.in);
      System.out.println("Enter legth of array");
      int n = sc.nextInt();
      int arr[] = new int[n];
      System.out.println("Enter elements of array");
      for (int i = 0; i < arr.length; i++) {
         arr[i] = sc.nextInt();
      }
      System.out.println("Enter start position from 1 to " + n);
      int start = sc.nextInt();
      System.out.println("Enter end position from 1 to " + n);
      int end = sc.nextInt();
      int slice[] = Arrays.copyOfRange(arr, (start - 1), (end));
      for (int i = 0; i < slice.length; i++) {
         System.out.print(slice[i] + " ");
      }
      sc.close();
   }
}
```

5. Write a program to update an element

Solution:

```
import java.util.*;
public class UpdateArray {
   public static void main(String[] args) {
      Scanner sc = new Scanner(System.in);
      System.out.println("Enter legth of array");
      int n = sc.nextInt();
      int arr[] = new int[n];
      System.out.println("Enter elements of array");
      for (int i = 0; i < arr.length; i++) {
         arr[i] = sc.nextInt();
      }
      System.out.println("Original array: " + Arrays.toString(arr));
      System.out.print("Enter the position to update (1 to " + arr.length + "):
");
      int position = sc.nextInt();
      System.out.print("Enter the new value: ");
      int newValue = sc.nextInt();
      if (position > 0 && position <= arr.length) {
         arr[position - 1] = newValue;
         System.out.println("Updated array: " + Arrays.toString(arr));
      } else {
         System.out.println("Invalid position!");
      }
      sc.close();
   }
}
```

6. Write a program to delete the element

Solution:

```
import java.util.*;

public class DeleteElement {
   public static void main(String[] args) {
      Scanner sc = new Scanner(System.in);
      System.out.println("Enter legth of array");
      int n = sc.nextInt();
      int arr[] = new int[n];
      System.out.println("Enter elements of array");
      for (int i = 0; i < arr.length; i++) {
         arr[i] = sc.nextInt();
      }
      System.out.println("Original array: " + Arrays.toString(arr));
      System.out.print("Enter the element to delete : ");
      int ele = sc.nextInt();
      boolean found = false;
      for (int i = 0; i < arr.length; i++) {
         if (ele == arr[i]) {
            for (int j = i; j < arr.length - 1; j++) {
               arr[j] = arr[j + 1];
            }
            arr[arr.length - 1] = 0;
            n--;
            found = true;
         }
      }
      if (found) {
         System.out.println("Array after deleting value " + ele);
```

```
            for (int i = 0; i < n; i++) {
                System.out.print(arr[i] + " ");
            }
            System.out.println();
        } else {
            System.out.println("Value not found in array");
        }

        sc.close();
    }
}
```

7. Write a program to insert the element

Solution:

```
import java.util.*;
public class InsertElement {
   public static void main(String[] args) {
      Scanner sc = new Scanner(System.in);
      System.out.println("Enter legth of array");
      int n = sc.nextInt();
      int arr[] = new int[n];
      System.out.println("Enter elements of array");
      for (int i = 0; i < arr.length; i++) {
         arr[i] = sc.nextInt();
      }
      System.out.println("Enter the position you want to add new
element");
      int position = sc.nextInt();
      System.out.println("Enter the value");
      int value = sc.nextInt();
      if (position < 1 || position > n) {
         System.out.println("Invali position");
      } else {
         int arr2[] = new int[n + 1];
         for (int i = 0; i < position - 1; i++) {
            arr2[i] = arr[i];
         }
         arr2[position - 1] = value;
         for (int i = position - 1; i < arr.length; i++) {
            arr2[i + 1] = arr[i];
         }
         System.out.println("Array after insertion: " + Arrays.toString(arr2));
      }}}
```