1.Youngest-oldest 1

```java
Import java.io.*;
Import java.util.*;
Public class Solution {
    Public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        Int n = sc.nextInt();
        If (n < 0) {
            System.out.println("Invalid Input");
            Return;
        }
        Int minAge = Integer.MAX_VALUE;
        Int maxAge = Integer.MIN_VALUE;
        Boolean invalidAgeFound = false;
        For (int I = 0; I < n; i++) {
            Int age = sc.nextInt();
            If (age < 0) {
                invalidAgeFound = true;
                break;
            }
            If (age < minAge) {
```

```java
        minAge = age;

      }

      If (age > maxAge) {

        maxAge = age;

      }

    }

    If (invalidAgeFound) {

      System.out.println("Invalid Input");

    } else {

      System.out.println("Youngest=" + minAge);

      System.out.println("Oldest=" + maxAge);

    }

    Sc.close();

  }

}
```

2.Array 176

```java
Import java.util.*;

Public class Main {

  Public static void main(String[] args) {

    Scanner sc = new Scanner(System.in);

    Int n = sc.nextInt();
```

```
    Int[] doors = new int[n];

    For (int I = 0; I < n; i++) {

        Doors[i] = sc.nextInt();

    }

    Int target = sc.nextInt();

    Int index = -1;

    For (int I = 0; I < n; i++) {

        If (doors[i] == target) {

            Index = I;

            Break;

        }

    }

    If (index == -1) {

        System.out.println("-1");

    } else {

        System.out.printf("Door Number is %03d-DN%n", index);

    }

  }

}
```

3.Difference of the character 1

Import java.util.*;

```java
Public class Main {

    Public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);

        String s = sc.next();

        Int countStar = 0, countHash = 0;

        For (char c : s.toCharArray()) {

            If (c == '*') countStar++;

            Else if (c == '#') countHash++;

        }

        Int diff = countStar – countHash;

        If (diff == 0) {

            System.out.println(0);

            Return;

        }

        Int absDiff = Math.abs(diff);

        Int width = (absDiff % 2 == 0) ? 2 : 3;

        String padded = String.format("%0" + width + "d", absDiff);

        If (diff < 0) padded = "-" + padded;

        System.out.println("The Difference of the character in the given string: " + padded);

    }
```

```
}
```

4.Count the Positive and Negative Integer Number 1

```
Import java.util.*;

Public class Main {

    Public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);

        Int n = sc.nextInt();

        Int positive = 0, negative = 0;

        For (int I = 0; I < n; i++) {

            Int val = sc.nextInt();

            If (val > 0) {

                Positive++;

            } else if (val < 0) {

                Negative++;

            }

        }

        System.out.printf("Count of Positive Integer is %.2f%n", (double) positive);

        System.out.printf("Count of Negative Integer is %.2f%n", (double) negative);

    }

}
```

5.ARRAY MEAN 4

```java
Import java.util.*;
Public class Main {
    Public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        Int n = sc.nextInt();
        Double sum = 0;
        For (int I = 0; I < n; i++) {
            Sum += sc.nextInt();
        }
        Double mean = sum / n;
        System.out.printf("Array Mean Value is %.2f%n", mean);
    }
}
```

6.Count distinct elements 8

```java
Import java.util.*;
Public class Main {
    Public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        Int n = sc.nextInt();
        Set<Integer> distinctSet = new HashSet<>();
```

```
For (int I = 0; I < n; i++) {

    distinctSet.add(sc.nextInt());

}

Int distinctCount = distinctSet.size();

If (distinctCount == 1) {

    System.out.println("There are 1 distinct element in the array.");

} else {

    System.out.println("There are " + distinctCount + " distinct
elements in the array.");

}

  }

}
```