**Task 1 – list explorer**

**Algorithm**1. Create an integer array to store temperatures.

2. Initialize it with a set of sample values.

3. Print all values in the array.

4. Calculate the sum of all values.

5. Compute the average (sum divided by number of elements).

6. Find the highest value in the array.

**Psudocode**

DECLARE array of integers

INITIALIZE array with temperature values

FOR each element in array

PRINT value

ADD to sum

IF value > max, UPDATE max

CALCULATE average = sum / number of elements

PRINT sum, average, and highest temperature

**Code(java)  
public class App {**

**public static void main(String[] args) {**

**int [] temperatures = {30, 32, 31, 29, 28, 27, 26, 25, 24, 23};**

**int highest = Integer.MIN\_VALUE;**

**int sum = 0;**

**for (int i=0; i<temperatures.length; i++) {**

**if (highest < temperatures[i]) {**

**highest = temperatures[i];**

**}**

**sum += temperatures[i];**

**System.out.println("Temperature at index " + i + ": " + temperatures[i]);**

**}**

**double average = (double) sum / temperatures.length;**

**System.out.println("sum: " + sum);**

**System.out.println("Average temperature: " + average);**

**System.out.println("Highest temperature: " + highest);**

**}**

**}**

**Output (test case 1)**

mUp\Bridge-course\day-7\task - 1\List explorer'; & 'C:\Program Files\Java\jdk-21\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\SRINIVASA\Documents\StemUp\Bridge-course\day-7\task - 1\List explorer\bin' 'App'

NIVASA\x5cDocuments\x5cStemUp\x5cBridge-course\x5cday-7\x5ctask - 1\x5cList explorer\x5cbin' 'App' ;0fbe4a85-635d-46ca-82e8-92f240080922Temperature at index 0: 30

Temperature at index 1: 32

Temperature at index 2: 31

Temperature at index 3: 29

Temperature at index 4: 28

Temperature at index 5: 27

Temperature at index 6: 26

Temperature at index 7: 25

Temperature at index 8: 24

Temperature at index 9: 23

sum: 275

Average temperature: 27.5

Highest temperature: 32

**Observations**

- Demonstrates use of arrays and enhanced for-loop.

- Applies basic aggregation operations: sum, average, and max.

- Highlights practical use of arrays in data processing.

**Task 2 – product of evens**

**Algorithm**1. Create an integer array containing numbers from 1 to 10.

2. Initialize a variable to store the product (start with 1).

3. Traverse the array using a loop.

4. For each number:

- Check if it is even (number % 2 == 0).

- If it is, multiply it to the product.

5. Print the final product.

**Psudocode**

DECLARE array of integers from 1 to 10

SET product = 1

FOR each number in array

IF number is even

MULTIPLY number with product

PRINT product

**Code(java)**

**Output (test case 1)**

**Output (test case 2)**

**Output (test case 3)**

**Observations**

**Task 1 –**

**Algorithm**

**Psudocode**

**Code(java)**

**Output (test case 1)**

**Output (test case 2)**

**Output (test case 3)**

**Observations**

**Task 1 –**

**Algorithm**

**Psudocode**

**Code(java)**

**Output (test case 1)**

**Output (test case 2)**

**Output (test case 3)**

**Observations**