University of Hyderabad
School of Computer and Information Sciences
Java Lab, Assignment 1

Solve the following Questions in Java and submit the same.

Q1. Fibonacci Series

Problem Description:
Write a program that takes a number n as input and prints the first n Fibonacci numbers.

Functionality:

- Implement a function **generateFibonacci(int n)** to generate and print the Fibonacci series up to the nth term.
- Handle edge cases for values of n less than or equal to 0.

Q2. Arithmetic Progression

Problem Description:
Write a program that takes three numbers a, d, and n as input, where:

- a is the first term of the arithmetic progression (AP),
- d is the common difference,
- n is the number of terms.

Print the first n terms of the arithmetic progression.

Functionality:

- Implement a function **generateAP(int a, int d, int n)** to generate and print the arithmetic progression.

Q3. Squares and Cubes

Problem Description:
Write a program that takes an integer n as input and prints a table displaying the first n numbers along with their squares and cubes.

Functionality:

- Implement a function **generateSquareCubeTable(int n)** to generate and display the table of numbers, their squares, and cubes.

Q4. Binary to Decimal Conversion

Problem Description:
Write a program that takes a binary number as input and converts it to its decimal equivalent.

Functionality:

- Implement a function **binaryToDecimal(binary)** to convert and return the decimal equivalent of the binary input.

Q5. Marks Analysis

Problem Description:
Write a program that accepts the number of students n and then takes the marks of n students as input. Calculate and print the maximum, minimum, and average marks.

Functionality:

- Implement a function **analyzeMarks(int[] marks)** to find and return the maximum, minimum, and average marks.

Instructions:

1. **Modularization:** Ensure that your solution is divided into functions for better structure and reusability.
2. **Error Handling:** Implement basic error handling for edge cases such as invalid inputs or data out of expected ranges.
3. **Documentation:** Write meaningful comments to describe the functionality of your code.
4. **Input/Output:** Prompt the user for inputs, and display the output in a user-friendly format.

Instructions for Submission:

- ➢ Once you've completed the exercises, you must upload your Program Source files. Please ensure that you also include screenshots of the outputs generated by your programs.
- ➢ Create a directory and name it **<yourRollNumber_Java1>,** and place all your files in it. Upload the compressed folder (zip) to Google Classroom as your solution to the assignment.