Q1. Read an Employee data with idno, name and mobilenumber (regular expression)

and compare the mobile number must have only 10 digits

name can consists of only alphabets , space character

idno number consists of 5 digits ?

ANS:

```
import java.util.Scanner;
public class EmployeeDataValidator {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);

        // Read employee data
        System.out.print("Enter ID number: ");
        String idno = scanner.nextLine();

        System.out.print("Enter name: ");
        String name = scanner.nextLine();

        System.out.print("Enter mobile number: ");
        String mobileNumber = scanner.nextLine();

        // Validate employee data
        boolean isValid = validateEmployeeData(idno, name, mobileNumber);

        if (isValid) {
            System.out.println("Employee data is valid.");
        } else {
            System.out.println("Invalid employee data.");
        }
    }

    public static boolean validateEmployeeData(String idno, String name, String
mobileNumber) {
        String idnoPattern = "^\\d{5}$";
        String namePattern = "^[A-Za-z ]+$";
        String mobilePattern = "^\\d{10}$";

        boolean idnoMatch = idno.matches(idnoPattern);
        boolean nameMatch = name.matches(namePattern);
        boolean mobileMatch = mobileNumber.matches(mobilePattern);

        return idnoMatch && nameMatch && mobileMatch;
    }
}
```

OUTPUT:

```
Enter ID number: 12345
Enter name: riya
Enter mobile number: 9866498094
Employee data is valid.
```

Q2. Write a mutithreading program,

thread 1 : to display all perfect numbers,

thread 2 : to display factorial value of numbers from 1 to 10 ?

ANS:

```
public class MultithreadingExample {
    public static void main(String[] args) {
        // Thread to display perfect numbers
        Thread perfectNumbersThread = new Thread(() -> {
            for (int i = 1; i <= 1000; i++) {
                if (isPerfectNumber(i)) {
                    System.out.println("Perfect Number: " + i);
                }
            }
        });

        // Thread to display factorial values
        Thread factorialThread = new Thread(() -> {
            for (int i = 1; i <= 10; i++) {
                long factorial = calculateFactorial(i);
                System.out.println("Factorial of " + i + ": " + factorial);
            }
        });

        // Start the threads
        perfectNumbersThread.start();
        factorialThread.start();
    }

    public static boolean isPerfectNumber(int number) {
        int sum = 0;
        for (int i = 1; i < number; i++) {
            if (number % i == 0) {
                sum += i;
            }
        }
        return sum == number;
    }

    public static long calculateFactorial(int number) {
        long factorial = 1;
        for (int i = 2; i <= number; i++) {
            factorial *= i;
        }
        return factorial;
    }
}
```

OUTPUT:

```
Perfect Number: 6
Perfect Number: 28
```

```
Perfect Number: 496
Factorial of 1: 1
Factorial of 2: 2
Factorial of 3: 6
Factorial of 4: 24
Factorial of 5: 120
Factorial of 6: 720
Factorial of 7: 5040
Factorial of 8: 40320
Factorial of 9: 362880
Factorial of 10: 3628800
```

Q3. Write a program to read the data from file?

ANS:

```
import java.io.BufferedReader;
import java.io.FileReader;
import java.io.IOException;
import java.util.Scanner;

public class ReadFile {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);

        System.out.print("Enter the file path: ");
        String filePath = scanner.nextLine();

        try (BufferedReader reader = new BufferedReader(new FileReader(filePath)))
{
            String line;
            while ((line = reader.readLine()) != null) {
                System.out.println(line);
            }
        } catch (IOException e) {
            System.out.println("Error reading file: " + e.getMessage());
        } finally {
            scanner.close();
        }
    }
}
```

OUTPUT:

```
Enter the file path:B.TECH.TXT //non-existent-file.txt
Error reading file: non-existent-file.txt (No such file or directory)
```

Q4. write a program to write the content to file in append mode ?

ANS:

```
import java.io.BufferedWriter;
import java.io.FileWriter;
import java.io.IOException;
import java.util.Scanner;
```

```
public class WriteToFile {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);

        System.out.print("Enter the file path: ");
        String filePath = scanner.nextLine();

        System.out.print("Enter the content to append: ");
        String content = scanner.nextLine();

        try (BufferedWriter writer = new BufferedWriter(new FileWriter(filePath,
true))) {
            writer.write(content);
            writer.newLine();
            System.out.println("Content appended to the file successfully.");
        } catch (IOException e) {
            System.out.println("Error writing to file: " + e.getMessage());
        } finally {
            scanner.close();
        }
    }
}
```

OUTPUT:

```
Enter the file path: LAB4.txt
Enter the content to append: This is the new content
Content appended to the file successfully.
```