```java
package lt.vtvpmc.ems.aidas;
/*
    log4j --> log4j2, j.u.l., logback, slf4j
    log4j: ALL, TRACE, DEBUG, INFO, WARN, ERROR, FATAL, OFF; // cia yra lygiai

 */

import org.apache.logging.log4j.LogManager; //naujas
import org.apache.logging.log4j.Logger;
//import org.slf4j.LoggerFactory; //senas

import java.io.File;
import java.io.IOException;

public class JavaLogging2 {

    //'prisukam' logeri:
    //private final static Logger logger = LoggerFactory.getLogger(JavaLogging.class); //senas
    private static final Logger logger = LogManager.getLogger(JavaLogging2.class.getName()); //naujas

    public static void main(String[] args) throws IOException{

        logger.trace("Message from TRACE level");
        logger.info("Message from INFO level");
        logger.error("Message from ERROR level");
        logger.info("Message from INFO level");

        createNewFile();
        createNewDirectory();
        createNewDirectories();
        createTempFile();
        renameFile();
        deleteFile();
    }

    private static void createNewFile() throws IOException {
        try {
            File file = new File("/abc/test.txt"); //naujo objekto sukurimas
                                                                  // veliau ivedam nesama direktorija homeris ir meta 'No such file or directory' - isisaiskinti kodel ?
            boolean bool = file.createNewFile();    //naujo failo sukurimas
            System.out.println("File created: " + bool);
        } catch (NullPointerException e) {
            //System.out.println("null" + e); //parodo kuris exception meta klaida
            e.printStackTrace(); //parodo klaidos atsiradimo kelia
        } catch (IOException e) {
            //System.out.println("io " + e); // parodo kuris exception meta klaida
            e.printStackTrace(); // versija su starcTrace
            logger.error("Sukuriant faila iskilo problemu" + e); //versija su logging
        }
    }

    private static void createNewDirectory() throws IOException {
        File file = new File("Java"); //naujo objekto sukurimas
        boolean bool = file.mkdirs(); //naujos direktorijos sukurimas
        System.out.println("Directory created: " + bool);
        logger.info("Direktorija sukurta" + bool);
    }

    private static void createNewDirectories() throws IOException {
        File file = new File("Java/Collections/List"); //naujo objekto sukurimas
        boolean bool = file.mkdir(); //nauju direktoriju sukurimas
        System.out.println("Directories created: " + bool);
    }

    private static void createTempFile() throws IOException {
        File tempFile = File.createTempFile("tmp", ".tmp");
        System.out.println("File path: " + tempFile.getAbsolutePath());
```

```java
        tempFile = File.createTempFile("tmp", null); //pakartota iliustravimui kad jei null vistiek sukuria tmp (zr. kode)
        System.out.println("File path: " + tempFile.getAbsolutePath());
    }

    private static void renameFile() throws IOException {
        File selectedFile = new File("test.txt");
        selectedFile.renameTo(new File ("new_test.txt"));
    }

    private static void deleteFile() throws IOException {
        File selectedFile = new File("new_test.txt");
        selectedFile.delete();
        System.out.println("File deleted: ");
        //logger.all();
    }

}

```