Source: <https://examples.javacodegeeks.com/core-java/writeread-csv-files-in-java-example/>

public class CsvFileWriter {

    //Delimiter used in CSV file

    private static final String COMMA\_DELIMITER = ",";

    private static final String NEW\_LINE\_SEPARATOR = "\n";

    //CSV file header

    private static final String FILE\_HEADER = "id,firstName,lastName,gender,age";

    public static void writeCsvFile(String fileName) {

        //Create new students objects

        Student student1 = new Student(1, "Ahmed", "Mohamed", "M", 25);

        Student student2 = new Student(2, "Sara", "Said", "F", 23);

        Student student3 = new Student(3, "Ali", "Hassan", "M", 24);

        Student student4 = new Student(4, "Sama", "Karim", "F", 20);

        Student student5 = new Student(5, "Khaled", "Mohamed", "M", 22);

        Student student6 = new Student(6, "Ghada", "Sarhan", "F", 21);

        //Create a new list of student objects

        List students = new ArrayList();

        students.add(student1);

        students.add(student2);

        students.add(student3);

        students.add(student4);

        students.add(student5);

        students.add(student6);

        FileWriter fileWriter = null;

        try {

            fileWriter = new FileWriter(fileName);

            //Write the CSV file header

            fileWriter.append(FILE\_HEADER.toString());

            //Add a new line separator after the header

            fileWriter.append(NEW\_LINE\_SEPARATOR);

            //Write a new student object list to the CSV file

            for (Student student : students) {

                fileWriter.append(String.valueOf(student.getId()));

                fileWriter.append(COMMA\_DELIMITER);

                fileWriter.append(student.getFirstName());

                fileWriter.append(COMMA\_DELIMITER);

                fileWriter.append(student.getLastName());

                fileWriter.append(COMMA\_DELIMITER);

                fileWriter.append(student.getGender());

                fileWriter.append(COMMA\_DELIMITER);

                fileWriter.append(String.valueOf(student.getAge()));

                fileWriter.append(NEW\_LINE\_SEPARATOR);

            }

            System.out.println("CSV file was created successfully !!!");

        } catch (Exception e) {

            System.out.println("Error in CsvFileWriter !!!");

            e.printStackTrace();

        } finally {

            try {

                fileWriter.flush();

                fileWriter.close();

            } catch (IOException e) {

                System.out.println("Error while flushing/closing fileWriter !!!");

                e.printStackTrace();

            }

        }

    }

}

public class CsvFileReader {

    //Delimiter used in CSV file

    private static final String COMMA\_DELIMITER = ",";

    //Student attributes index

    private static final int STUDENT\_ID\_IDX = 0;

    private static final int STUDENT\_FNAME\_IDX = 1;

    private static final int STUDENT\_LNAME\_IDX = 2;

    private static final int STUDENT\_GENDER = 3;

    private static final int STUDENT\_AGE = 4;

    public static void readCsvFile(String fileName) {

        BufferedReader fileReader = null;

        try {

            //Create a new list of student to be filled by CSV file data

            List students = new ArrayList();

            String line = "";

            //Create the file reader

            fileReader = new BufferedReader(new FileReader(fileName));

            //Read the CSV file header to skip it

            fileReader.readLine();

            //Read the file line by line starting from the second line

            while ((line = fileReader.readLine()) != null) {

                //Get all tokens available in line

                String[] tokens = line.split(COMMA\_DELIMITER);

                if (tokens.length > 0) {

                    //Create a new student object and fill his  data

                    Student student = new Student(Long.parseLong(tokens[STUDENT\_ID\_IDX]), tokens[STUDENT\_FNAME\_IDX], tokens[STUDENT\_LNAME\_IDX], tokens[STUDENT\_GENDER], Integer.parseInt(tokens[STUDENT\_AGE]));

                    students.add(student);

                }

            }

            //Print the new student list

            for (Student student : students) {

                System.out.println(student.toString());

            }

        }

        catch (Exception e) {

            System.out.println("Error in CsvFileReader !!!");

            e.printStackTrace();

        } finally {

            try {

                fileReader.close();

            } catch (IOException e) {

                System.out.println("Error while closing fileReader !!!");

                e.printStackTrace();

            }

        }

    }

}