

CSV serializier

CSV файлът е текстов файл, в който се съхраняват стойности разделени помежду си със запетая. Всеки ред от файла се нарича “data record” – запис. Линк с повече информация:

https://en.wikipedia.org/wiki/Comma-separated_values

Сериализацията е процес, при който обекти се конвертират в поредица от байтове с цел да бъдат съхранени или изпратени по мрежата. Десериализацията е обратният процес по създаване на обекти от поредица от байтове. Линк с повече информация: <https://hazelcast.com/glossary/serialization/>

Напишете програма на C, която обработва csv файл с информация за продажбите на верига магазини и има възможност да:

1. Приема следните аргументи от командния ред

Опция	Значение	Задължителна	Аргумент	Ефект
s	silent	Не	Няма	Спира всички логове от програмата
f	file	Да	Пълен път до файл	Програмата обработва файла
c	command	Да	serialize/deserialize	Определя вида на файла, който ще се обработва – бинарен или текстов

2. Сериализира данните от подадения csv файл ако подадената команда като аргумент е “serialize”. Сериализираните данни се записват в нов бинарен файл. **Основната цел е той да има по-малък размер от оригиналния csv файл.**

3. Десериализира данните от подаден бинарен файл ако подадената команда като аргумент е “deserialize” и ги възстановява до оригиналния им вид в нов csv файл. Бинарният файл е такъв, който програмата е сериализирала.

Примерен файл за тестване на програмата е sales.csv файлът.

Примерен запис от файла:

```
1 "Row ID","Order ID","Order Date","Customer ID","City","State","Postal Code","Region","Product ID","Category","Sub-Category","Price"
2 1,"CA-2016-152156",11/8/2016,"CG-12520","Henderson","Kentucky",42420,"South","FUR-BO-10001798","Furniture","Bookcases",261.96
3 2,"CA-2016-152156",11/8/2016,"CG-12520","Henderson","Kentucky",42420,"South","FUR-CH-10000454","Furniture","Chairs",731.94
```

Допълнителни изисквания и насоки:

- Първият ред от csv файла показва формата му - имената на полетата във всеки запис
- Полетата от тип символен низ са оградени в двойни кавички
- Програмата спира изпълнение ако е подаден файл в друг формат или такъв с невалиден формат
- Използвайте подходящи типове данни – с правилни дължина и знак
- Организирайте кода на програмата си в отделни файлове и функции
- Когато се очаква потребителят да въведе стойност от клавиатурата, изведете подходящо съобщение на стандартния изход
- Логовете, които отпечатва програмата на стандартния изход ясно показват от коя функция са, и кои са релевантните данни при изпълнението ѝ (аргументи, връщани стойности и т.н) изход
- За обработката на опциите и аргументите от командния ред използвайте функцията “getopt”
- Не забравяйте, че в текстов файл се съхраняват ASCII символи
- Проучете възможността за преобразуване на символни низове в числени стойности чрез функции от стандартната библиотека

Бъдете готови да демонстрирате работата на програмата, както и да отговаряте на допълнителни въпроси. Възможно е от вас да бъде поискано да модифицирате части от програмата на място в деня на защитата.

При наличие на въпроси с удоволствие ще се радвам да ги обсъдим

Кристиян Златков

kristiqnz11@gmail.com