

netology-code / cppl-diplom Public

1 star 4 forks Branches Tags Activity

Star

Notifications

<> Code Issues Pull requests Actions Projects Security Insights



1 Branch 0 Tags



o to file

Go to file

<> Code



vikki-ya Update README.md

e4a1ccb · 2 years ago



README.md

Update README.md

2 years ago

README



Курсовой проект «Парсер INI-файлов»

Курсовой проект — простой парсер файлов формата INI.

Содержание

1. Введение

INI-файлы — это обычные текстовые файлы, которые можно редактировать и просматривать при помощи любого текстового редактора. INI-файлы приобрели популярность в Windows, хотя могут использоваться в любой ОС. Несложная структура этого формата позволяет легко обрабатывать их программно и имеет достаточно понятный вид для чтения и изменения человеком.

2. Формат INI-файлов

INI-файлы могут содержать типы строк:

- Строка имени секции — обозначает имя новой секции переменных. Имеет формат: `[имя_секции]`.

Пробелы и символы табуляции, стоящие до открывающей и после закрывающей скобки, игнорируются.

- Строка задания переменной — задаёт значение переменной. Имеет формат:
`название_переменной = значение_переменной` .
Количество пробелов между символами равенства может быть произвольным. Для простоты будем считать, что значения могут быть либо строкой, либо числом. Несколько значений не допускается.
- Строка комментариев. Имеет формат: `; строка комментариев` .
Парсер должен игнорировать такие строки, с ними ничего делать не нужно.

3. Общая структура INI-файла

Пример структуры INI-файла:

```
[Section1]
; комментарий о разделе
var1=5.0 ; иногда допускается комментарий к отдельному параметру
var2=какая-то строка

[Section2]
var1=1
var2=значение_2

; Иногда значения отсутствуют, это означает, что в Section3 нет переменных
[Section3]
[Section4]
Mode=
Vid=

; Секции могут повторяться
[Section1]
var3=значение
var1=1.0 ; переприсваиваем значение
```

4. Пример работы парсера

Вам нужно реализовать класс парсера INI-файлов, который предоставляет для пользователя одну шаблонную функцию, позволяющую получить значение переменной в определённой секции.

```
ini_parser parser("filename");
auto value = parser.get_value<int>("section.value")
```

Если в файле нет значения для этой переменной, то нужно бросить исключение. Если есть проблемы со считыванием файла или файл некорректный, то нужно сообщить об этом в исключении.

5. Дополнительное задание *

- Если файл содержит некорректный синтаксис, нужно указать в исключении, на какой строке встретилась проблема.
- Если искомого значения переменной нет, нужно вывести подсказку для пользователя — названия других переменных из этой секции. Возможно, пользователь опечатался.

Как задавать вопросы руководителю по курсовой работе

1. Если у вас возник вопрос, попробуйте сначала самостоятельно найти ответ в интернете. Навык поиска информации пригодится вам в любой профессиональной деятельности. Если ответ не нашёлся, можно уточнить у преподавателя.
2. Если у вас набирается несколько вопросов, присылайте их в виде нумерованного списка. Так преподавателю будет проще отвечать на каждый из них.
3. Для лучшего понимания контекста прикрепите к вопросу скриншоты и стрелкой укажите, что именно вызывает вопрос. Программу для создания скриншотов можно скачать [по ссылке](#).
4. По возможности задавайте вопросы в комментариях к коду.
5. Формулируйте свои вопросы чётко, дополняя их деталями. На сообщения «Ничего не работает», «Всё сломалось» преподаватель не сможет дать комментарии без дополнительных уточнений. Это затянет процесс получения ответа.
6. Постарайтесь набраться терпения в ожидании ответа на свои вопросы. Преподаватели Нетологии — практикующие разработчики, поэтому они не всегда могут отвечать моментально. Зато их практика даёт возможность делиться с вами не только теорией, но и ценным прикладным опытом.

Рекомендации по работе над курсовой работой:

1. Не откладывайте надолго начало работы над курсовым проектом. В таком случае у вас останется больше времени на получение рекомендаций от преподавателя и доработку проекта.
2. Разбейте работу над курсовым проектом на части и выполняйте их поочерёдно. Вы будете успевать учитывать комментарии от преподавателя и не терять мотивацию на полпути.

Инструкция по выполнению курсовой работы

1. Выполняйте работу в [GitHub](#).
2. Скопированную ссылку с вашей курсовой работой нужно отправить на проверку. Для этого перейдите в личный кабинет на сайте [netology.ru](#), в поле комментария к курсовой вставьте скопированную ссылку и отправьте работу на проверку.
3. Работу можно сдавать частями.

Критерии оценки курсовой работы

Releases

No releases published

Packages

No packages published

Contributors 3



vikki-ya



alexeycas6 Alexey Sinitskiy



akulm Аня Кульмаметова