

Общее описание работы

Разработать инструмент командной строки для учебного конфигурационного языка. Этот инструмент преобразует текст из входного формата (TOML) в выходной. Синтаксические ошибки выявляются с выдачей сообщений.

Используемые модули и язык программирования

Использую язык программирования Python версии 3.10 В качестве модулей, импортируемых в программу, использую

1. os - для работы с путями.
2. keyboard - для работы с клавиатурой.
3. threading - для создания многопоточности.
4. time - для создания задержек при получении строк от пользователя, чтобы не перегружать цп.
5. sys - для взаимодействия с аргументами из командной строки и получения строк пользователя.
6. re - для работы над регулярными выражениями.
7. pprint (функция pprint) - для красивого вида получившегося словаря.

Описание команд, функционирование программы

Пользователь через консоль вводит параметр для программы (путь текстового файла, в который и будет записан итоговый формат), затем до ввода сочетания ctrl+d вводит данные, затем они обрабатываются и выводится результат (при некорректных данных сообщение об ошибке) Сам процесс был разделён на 3 этапа. Пробежимся по каждому:

1. Получение строк от пользователя до нажатия сочетания клавиш(функция read_input(text) параллельно выполняется с частью main)
2. Парсинг этих данных (функция parse(value) и parse_array(value))
3. Перевод этих данных в учебный язык (при успешном вводе пользователя в файл). Имеется в виду обработка особенностей учебного языка, которые могут быть несовместимы с корректным TOML файлом.

Примечание: язык TOML это в своём роде язык таблиц, так как все данные воспринимаются как часть какой-то таблицы (по умолчанию существует корневая таблица), поэтому вполне справедливо, что результат на конфигурационном языке это один большой словарь с поправкой на его синтаксис. Все функции также обрабатывают случаи, когда аргументы некорректны.

Тестирование

Итак, пусть пользователь введёт в консоль следующий текст:

```
[Owner]
Name = "Tom Preston-Werner"
Dob = 1979-05-27T07:32:00Z
Is_admin = true
[Database]
Server = "192.168.1.1"
Ports = [8001, 8002, 8003]
Connection_max = 5000
Enabled = true
[Servers.Alpha]
Ip = "10.0.0.1"
Dc = "eqdc10"
Enabled = true
Ports = [80, 443]
[Servers.Beta]
Ip = "10.0.0.2"
Dc = "eqdc20"
Enabled = false
Ports = [8080, 443]
# Массив с различными типами данных
Strings = ["apple", "banana", "cherry"]
Booleans = [true, false, true]
Nested_array = [[1, 2, 3], ["a", "b", "c"], [true, false]]
[Comment_example] # Это просто комментарий
#a это прям большой комментарий, на всю строку
^Dфайл toml верный
ошибка записи, конвертирование невозможно
```

Возникла ошибка - "конвертирование невозможно", так как на учебный язык невозможно перевести булевы значения, даты и строки. Попытка номер 2:

```
C:\Users\Antua\PycharmProjects\config3>python language.py C:\Users\Antua\PycharmProjects\config3\testfile.txt
[Owner]
id = 228
[Owner]
id = 227
^DINVALID REDEFINITION OF TABLE 'Owner'
```

Возникла ошибка - текст оформлен не в соответствии с языком TOML. И наконец, попытка номер 3:

```
C:\Users\Antua\PycharmProjects\config3>python language.py C:\Users\Antua\PycharmProjects\config3\testfile.txt
[Owner]
ID = 228
Karandashi = [1, 2, 3, 4, 5]
[EDA.Vtornic]
Grade= 5
#комментарий
Price = 100 # что дорого
^Dфайл toml верный
```

Следующий текст запишется в файл с окончанием testfile.txt:

Всё так, как и должно быть!

Тесты

В папке "tests" хранится файл с расширением .py, содержащий в себе модуль тестирования функций `parse(value)` и `parse_array(value)`

```
C:\Users\Antua\PycharmProjects\config3>python language.py C:\Users\Antua\PycharmProjects\config3\testfile.txt
[Owner]
Name = "Tom Preston-Werner"
Dob = 1979-05-27T07:32:00Z
Is_admin = true
[Database]
Server = "192.168.1.1"
Ports = [8001, 8002, 8003]
Connection_max = 5000
Enabled = true
[Servers.Alpha]
Ip = "10.0.0.1"
Dc = "eqdc10"
Enabled = true
Ports = [80, 443]
[Servers.Beta]
Ip = "10.0.0.2"
Dc = "eqdc20"
Enabled = false
Ports = [8080, 443]
# Массив с различными типами данных
Strings = ["apple", "banana", "cherry"]
Booleans = [true, false, true]
Nested_array = [[1, 2, 3], ["a", "b", "c"], [true, false]]
[Comment_example] # Это просто комментарий
#a это прям большой комментарий, на всю строку
```

Все они, как видно, из рисунка, пройдены успешно.

Url репозитория.

Домашнее задание размещено на гитхабе, ссылка на репозиторий: <https://github.com/Anton-Su/config3>