Работа со строковыми значениями

Автор задач: Блохин Н.В. (NVBlokhin@fa.ru)

Материалы:

- Макрушин С.В. Лекция "Работа со строковыми значениям"
- https://pyformat.info/
- https://docs.python.org/3/library/re.html
  - https://docs.python.org/3/library/re.html#flags
  - https://docs.python.org/3/library/re.html#functions
- https://pythonru.com/primery/primery-primeneniya-regulyarnyh-vyrazheniy-v-python
- https://kanoki.org/2019/11/12/how-to-use-regex-in-pandas/
- https://realpython.com/nltk-nlp-python/

In [914…
```
import pandas as pd
import nltk
from nltk import word_tokenize
from nltk.tokenize import sent_tokenize
from nltk.tag import pos_tag
from bs4 import BeautifulSoup
import re

nltk.download('punkt_tab')
nltk.download('averaged_perceptron_tagger')
```

```
[nltk_data] Downloading package punkt_tab to
[nltk_data]     /Users/egorsipilov/nltk_data...
[nltk_data]   Package punkt_tab is already up-to-date!
[nltk_data] Downloading package averaged_perceptron_tagger to
[nltk_data]     /Users/egorsipilov/nltk_data...
[nltk_data]   Package averaged_perceptron_tagger is already up-to-
[nltk_data]       date!
```

Out[914… True

Задачи для совместного разбора

1. Вывести на экран данные из словаря `obj` построчно в виде `k = v`, задав формат таким образом, чтобы знак равенства оказался на одной и той же позиции во всех строках. Строковые литералы обернуть в кавычки.

In [918…
```
obj = {
    "home_page": "https://github.com/pypa/sampleproject",
    "keywords": "sample setuptools development",
    "license": "MIT",
}
```

In [920…
```
def print_dict(obj: dict) -> None:
    max_len = max(len(key) for key in obj) + 2
    for key, value in obj.items():
        k = f'"{key}"'
        print(f'{k:<{max_len}} = "{value}"')
```

In [922…
```
print_dict(obj)
```

```
"home_page" = "https://github.com/pypa/sampleproject"
"keywords"  = "sample setuptools development"
"license"   = "MIT"
```

2. Написать регулярное выражение,которое позволит найти номера групп студентов.

In [925…
```
obj = pd.Series(["Евгения гр.ПМ19-1", "Илья пм 20-4", "Анна 20-3"])
obj
```

Out[925…
```
0    Евгения гр.ПМ19-1
1         Илья пм 20-4
2            Анна 20-3
dtype: object
```

In [927…
```
pattern = r'\d\d-\d'
obj.str.findall(pattern)
```

Out[927…
```
0    [19-1]
1    [20-4]
2    [20-3]
dtype: object
```

3. Разбейте текст формулировки задачи 2 на слова.

In [932…
```
text  = "Написать регулярное выражение,которое позволит найти номера групп студентов."
word_tokenize(text)
```

Out[932…
```
['Написать',
 'регулярное',
 'выражение',
 ',',
 'которое',
 'позволит',
 'найти',
 'номера',
 'групп',
 'студентов',
 '.']
```

Лабораторная работа 6.1

Форматирование строк

6.1.1. Загрузите данные из файла `recipes_sample.csv` (**ЛР2**) в виде `pd.DataFrame` `recipes` При помощи форматирования строк выведите информацию об id рецепта и времени выполнения 5 случайных рецептов в виде таблицы следующего вида:

```
|      id      |  minutes  |
|--------------------------|
|    61178     |    65     |
|    202352    |    80     |
|    364322    |    150    |
|    26177     |    20     |
|    224785    |    35     |
```

Обратите внимание, что ширина столбцов заранее неизвестна и должна рассчитываться динамически, в зависимости от тех данных, которые были выбраны.

In [937…
```
recipes = pd.read_csv("recipes_sample.csv")
recipes.head()
```

Out[937…

	name	id	minutes	contributor_id	submitted	n_steps	description	n_ingredients
0	george s at the cove black bean soup	44123	90	35193	2002-10-25	NaN	an original recipe created by chef scott meska...	18.0
1	healthy for them yogurt popsicles	67664	10	91970	2003-07-26	NaN	my children and their friends ask for my homem...	NaN
2	i can t believe it s spinach	38798	30	1533	2002-08-29	NaN	these were so go, it surprised even me.	8.0
3	italian gut busters	35173	45	22724	2002-07-27	NaN	my sister-in-law made these for us at a family...	NaN
4	love is in the air beef fondue sauces	84797	25	4470	2004-02-23	4.0	i think a fondue is a very romantic casual din...	NaN

In [939…
```
def recipes_sample(obj: pd.DataFrame) -> None:
    col = ['id', 'minutes']
    n_rows = 5
    samp = obj[['id', 'minutes']].sample(5)

    n1 = samp.id.astype(str).str.len().max() + 8
    n2 = samp.minutes.astype(str).str.len().max() + 8

    print(f'|{'id':^{n1}}|{'minutes':^{n2}}|')
    print('|' + "-"*(n1+n2+1) + '|')
    for i in range(n_rows):
        print(f"|{samp.id.iloc[i]:^{n1}}|{samp.minutes.iloc[i]:^{n2}}|")
```

In [941…
```
recipes_sample(recipes)
```

```
|     id      |  minutes  |
|-------------------------|
|    38672    |    370    |
|    54865    |    30     |
|    90193    |    40     |
|    44922    |    20     |
|    58935    |    18     |
```

In [943…
```python
def recipes_sample(obj: pd.DataFrame, cols: list) -> None:
    n_rows = 5
    samp = obj[cols].sample(5)

    header = ""
    data  = ""
    paddings = {}
    for v in cols:
        n = max(samp[v].astype(str).str.len().max(), len(v)) + 8
        paddings[v] = n
        header += f"|{v:^{n}}"

    header += "|"

    print(header)
    print('|' + "-"*(len(header) - 2) + '|')

    for i in range(n_rows):
        row = ""
        for v in cols:
            row += f"|{str(samp[v].iloc[i]):^{paddings[v]}}"
        row += "|"
        print(row)
```

In [945…
```python
recipes_sample(recipes, ["id", "n_ingredients", "minutes", "contributor_id"])
```

```
|      id      |    n_ingredients    |    minutes    |    contributor_id    |
|-------------------------------------------------------------------------|
|    447922    |        21.0         |      80       |       1744326        |
|    530211    |         7.0         |       5       |      2001375952      |
|    316435    |         nan         |      180      |        450004        |
|    57655     |         nan         |      15       |        27783         |
|    236900    |         9.0         |      120      |        37779         |
```

6.1.2. Напишите функцию `show_info`, которая по данным о рецепте создает строку (в смысле объекта python) с описанием следующего вида:

```
"Название Из Нескольких Слов"

1. Шаг 1
2. Шаг 2
----------
Автор: contributor_id
Среднее время приготовления: minutes минут
```

Данные для создания строки получите из файлов `recipes_sample.csv` (**ЛР2**) и `steps_sample.xml` (**ЛР3**). Вызовите данную функцию для рецепта с id `170895` и выведите (через `print`) полученную строку на экран.

In [948…
```python
def parse(recipe_id: int, recipes_df: pd.DataFrame, steps_file: str):
    recipe = recipes_df[recipes_df['id'] == recipe_id].iloc[0]
    name = recipe['name'].title()
    contributor_id = recipe['contributor_id']
    minutes = recipe['minutes']

    with open(steps_file, 'r', encoding='utf-8') as file:
        soup = BeautifulSoup(file, 'xml')


    steps = []
    for recipe in soup.find_all('recipe'):
        if recipe.find('id').text == str(recipe_id):
            steps = [step.text.capitalize().strip() for step in recipe.find_all('step')]
            break

    return name, steps, minutes, contributor_id
```

In [950…
```python
def show_info(name, steps, minutes, author_id) -> str:
    steps_cap = []
    for step in steps:
        steps_cap.append(step.capitalize())
    name = name.title()
    result = f'"{name}"\n\n'
    for i, step in enumerate(steps_cap, 1):
```

```
        result += f"{i}. {step}\n"
    result += "----------\n"
    result += f"Автор: {author_id}\n"
    result += f"Среднее время приготовления: {minutes} минут\n"

    return result
```

In [952…
```
name, steps, minutes, contributor_id = parse(170895, recipes, "steps_sample.xml")
print(show_info(name, steps, minutes, contributor_id))
```

```
"Leeks And Parsnips  Sauteed Or Creamed"

1. Clean the leeks and discard the dark green portions
2. Cut the leeks lengthwise then into one-inch pieces
3. Melt the butter in a medium skillet , med
4. Heat
5. Add the garlic and fry 'til fragrant
6. Add leeks and fry until the leeks are tender , about 6-minutes
7. Meanwhile , peel and chunk the parsnips into one-inch pieces
8. Place in a steaming basket and steam 'til they are as tender as you prefer
9. I like them fork-tender
10. Drain parsnips and add to the skillet with the leeks
11. Add salt and pepper
12. Gently sautee together for 5-minutes
13. At this point you can serve it , or continue on and cream it:
14. In a jar with a screw top , add the half-n-half and arrowroot
15. Shake 'til blended
16. Turn heat to low under the leeks and parsnips
17. Pour in the arrowroot mixture , stirring gently as you pour
18. If too thick , gradually add the water
19. Let simmer for a couple of minutes
20. Taste to adjust seasoning , probably an additional 1 / 2 teaspoon salt
21. Serve warm
----------
Автор: 8377
Среднее время приготовления: 27 минут
```

In [954…
```
assert (
    show_info(
        name="george s at the cove black bean soup",
        steps=[
            "clean the leeks and discard the dark green portions",
            "cut the leeks lengthwise then into one-inch pieces",
            "melt the butter in a medium skillet , med",
        ],
        minutes=90,
        author_id=35193,
    )
    == '"George S At The Cove Black Bean Soup"\n\n1. Clean the leeks and discard the dark green portions\n2. Cu
)
```

Работа с регулярными выражениями

6.1.3. Напишите регулярное выражение, которое ищет следующий паттерн в строке: число (1 цифра или более), затем пробел, затем слова: hour или hours или minute или minutes. Произведите поиск по данному регулярному выражению в каждом шаге рецепта с id 25082. Выведите на экран все непустые результаты, найденные по данному шаблону.

In [958…
```
def find_time(recipe_id: int, steps_file: str) -> list:
    with open(steps_file, 'r', encoding='utf-8') as file:
        soup = BeautifulSoup(file, 'xml')

    steps = []
    for recipe in soup.find_all('recipe'):
        if recipe.find('id').text == str(recipe_id):
            steps = [step.text.strip() for step in recipe.find_all('step')]
            break

    pattern = r'\d+\s+(?:hour|hours|minute|minutes)'

    matches = []
    for step in steps:
        found = re.findall(pattern, step, re.IGNORECASE)
        if found:
            matches.extend(found)

    return matches
```

In [960…
```
matches = find_time(25082, "steps_sample.xml")
for match in matches:
    print(match)
```

```
20 minute
10 minute
2 hour
10 minute
20 minute
30 minute
```

6.1.4. Напишите регулярное выражение, которое ищет шаблон вида "this..., but" *в начале строки* . Между словом "this" и частью ", but" может находиться произвольное число букв, цифр, знаков подчеркивания и пробелов. Никаких других символов вместо многоточия быть не может. Пробел между запятой и словом "but" может присутствовать или отсутствовать.

Используя строковые методы `pd.Series` , выясните, для каких рецептов данный шаблон содержится в тексте описания. Выведите на экран количество таких рецептов и 3 примера подходящих описаний (текст описания должен быть виден на экране полностью).

In [963…
```python
pattern = r'^this[a-zA-Z0-9_\s]*,(?:\s)?but'

matches = recipes['description'].str.contains(pattern, case=False, na=False, regex=True)

count = matches.sum()
count
```

Out[963… 133

In [965…
```python
tests = recipes[matches][['id', 'description']].sample(3)
for _, row in tests.iterrows():
    print(f"\nid: {row['id']}\nОписание: {row['description']}\n")
```

```
id: 169770
Описание: this drink is rich, but believe it or not, it is low fat!  definitely satisfies a chocolate craving.


id: 218825
Описание: this is pretty easy, but oh-so-fancy. great for a girl party.


id: 37942
Описание: this recipe originally from a taste of home magazine, but i made a few changes over the last few years
. it is a wonderfully moist bread and little ones in the family can't figure out how
```

Лабораторная работа 6.2

6.2.1. В текстах шагов рецептов обыкновенные дроби имеют вид "a / b". Используя регулярные выражения, уберите в тексте шагов рецепта с id 72367 пробелы до и после символа дроби. Выведите на экран шаги этого рецепта после их изменения.

In [969…
```python
def remove(recipe_id: int, steps_file: str):
    with open(steps_file, 'r', encoding='utf-8') as file:
        soup = BeautifulSoup(file, 'xml')

    steps = []
    for recipe_tag in soup.find_all('recipe'):
        if recipe_tag.find('id').text == str(recipe_id):
            steps = [step.text.strip() for step in recipe_tag.find_all('step')]
            break

    pattern = r'(\d+)\s*/\s*(\d+)'

    modified_steps = []
    for step in steps:
        modified_step = re.sub(pattern, r'\1/\2', step)
        modified_steps.append(modified_step)

    return modified_steps
```

In [971…
```python
modified_steps = remove(72367, "steps_sample.xml")
for i, step in enumerate(modified_steps, 1):
    print(f"Шаг {i}: {step}")
```

```
Шаг 1: mix butter , flour , 1/3 c
Шаг 2: sugar and 1-1/4 t
Шаг 3: vanilla
Шаг 4: press into greased 9" springform pan
Шаг 5: mix cream cheese , 1/4 c
Шаг 6: sugar , eggs and 1/2 t
Шаг 7: vanilla beating until fluffy
Шаг 8: pour over dough
Шаг 9: combine apples , 1/3 c
Шаг 10: sugar and cinnamon
Шаг 11: arrange on top of cream cheese mixture and sprinkle with almonds
Шаг 12: bake at 350 for 45-55 minutes , or until tester comes out clean
```

Сегментация текста

6.2.2. Разбейте тексты шагов рецептов на слова при помощи пакета `nltk`. Посчитайте и выведите на экран кол-во уникальных слов среди всех рецептов. Словом называется любая последовательность алфавитных символов (для проверки можно воспользоваться `str.isalpha`). При подсчете количества уникальных слов не учитывайте регистр.

In [975…
```python
def count_unique(steps_file: str):
    with open(steps_file, 'r', encoding='utf-8') as file:
            soup = BeautifulSoup(file, 'xml')

    all_steps = []
    for recipe_tag in soup.find_all('recipe'):
        steps = [step.text.strip() for step in recipe_tag.find_all('step')]
        all_steps.extend(steps)

    unique_words = set()
    for step in all_steps:
        tokens = word_tokenize(step.lower())
        words = [token for token in tokens if token.isalpha()]
        unique_words.update(words)

    return len(unique_words)
```

In [977…
```python
unique_words = count_unique("steps_sample.xml")
unique_words
```

Out[977… 14926

6.2.3. Разбейте описания рецептов из `recipes` на предложения при помощи пакета `nltk`. Найдите 5 самых длинных описаний (по количеству *предложений*) рецептов в датасете и выведите строки фрейма, соответствующие этим рецептами, в порядке убывания длины.

In [980…
```python
def count_sentences(description):
    if pd.isna(description):
        return 0
    return len(sent_tokenize(description))


recipes['sentence_count'] = recipes['description'].apply(count_sentences)
top_5 = recipes.nlargest(5, 'sentence_count').sort_values(by=['sentence_count'], ascending=False)

top_5
```

Out[980…

	name	id	minutes	contributor_id	submitted	n_steps	description	n_ingredients	sentence_count
18408	my favorite buttercream icing for decorating	334113	30	681465	2008-10-30	12.0	this wonderful icing is used for icing cakes a...	NaN	76
481	alligator claws avocado fritters with chipot...	287008	45	765354	2008-02-19	NaN	a translucent golden-brown crust allows the gr...	9.0	27
22566	rich barley mushroom soup	328708	60	221776	2008-10-03	NaN	this is one of the best soups i've ever made a...	10.0	24
6779	chocolate tea	205348	6	428824	2007-01-14	NaN	i wrote this because there are an astounding l...	NaN	23
16296	little bunny foo foo cake carrot cake with c...	316000	68	689540	2008-07-27	14.0	the first time i made this cake i grated a mil...	NaN	23

6.2.4. Напишите функцию, которая для заданного предложения выводит информацию о частях речи слов, входящих в

предложение, в следующем виде:

```
   PRP   VBD   DT      NNS     CC   VBD      NNS        RB
    I  omitted the raspberries and added strawberries instead
```

Для определения части речи слова можно воспользоваться `nltk.pos_tag`.

Проверьте работоспособность функции на названии рецепта с id 241106.

Обратите внимание, что часть речи должна находиться ровно посередине над соотвествующим словом, а между самими словами должен быть ровно один пробел.

In [100…
```python
def print_pos_tags(sentence):
    tokens = word_tokenize(sentence)
    pos_tags = pos_tag(tokens)

    tags_line = []
    words_line = []
    for word, tag in pos_tags:
        padding = max(len(word), len(tag))
        tags_line.append(f"{tag:^{padding}}")
        words_line.append(f"{word:^{padding}}")

    print(" ".join(tags_line))
    print(" ".join(words_line))
```

In [100…
```python
recipe_name = recipes[recipes['id'] == 241106]['name'].iloc[0]

print(f"Recipe name: {recipe_name.capitalize()}\n")
print_pos_tags(recipe_name)
```

```
Recipe name: Eggplant steaks with chickpeas  feta cheese and black olives

   JJ     NNS    IN     NNS    VBP    JJ   CC   JJ    NNS
eggplant steaks with chickpeas feta cheese and black olives
```