

Разбор домашнего задания

Роман Булгаков

Спикер курса

Skillbox

Задача «Список списков»

Входные данные:

многомерный список

```
[[[1, 2, 3], [4, 5, 6], [7, 8, 9]], [[10, 11, 12], [13, 14, 15], [16, 17, 18]]]
```

Выходные данные:

одномерный список

```
[1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18]
```

Задача «Шифр Цезаря»

Условия задачи

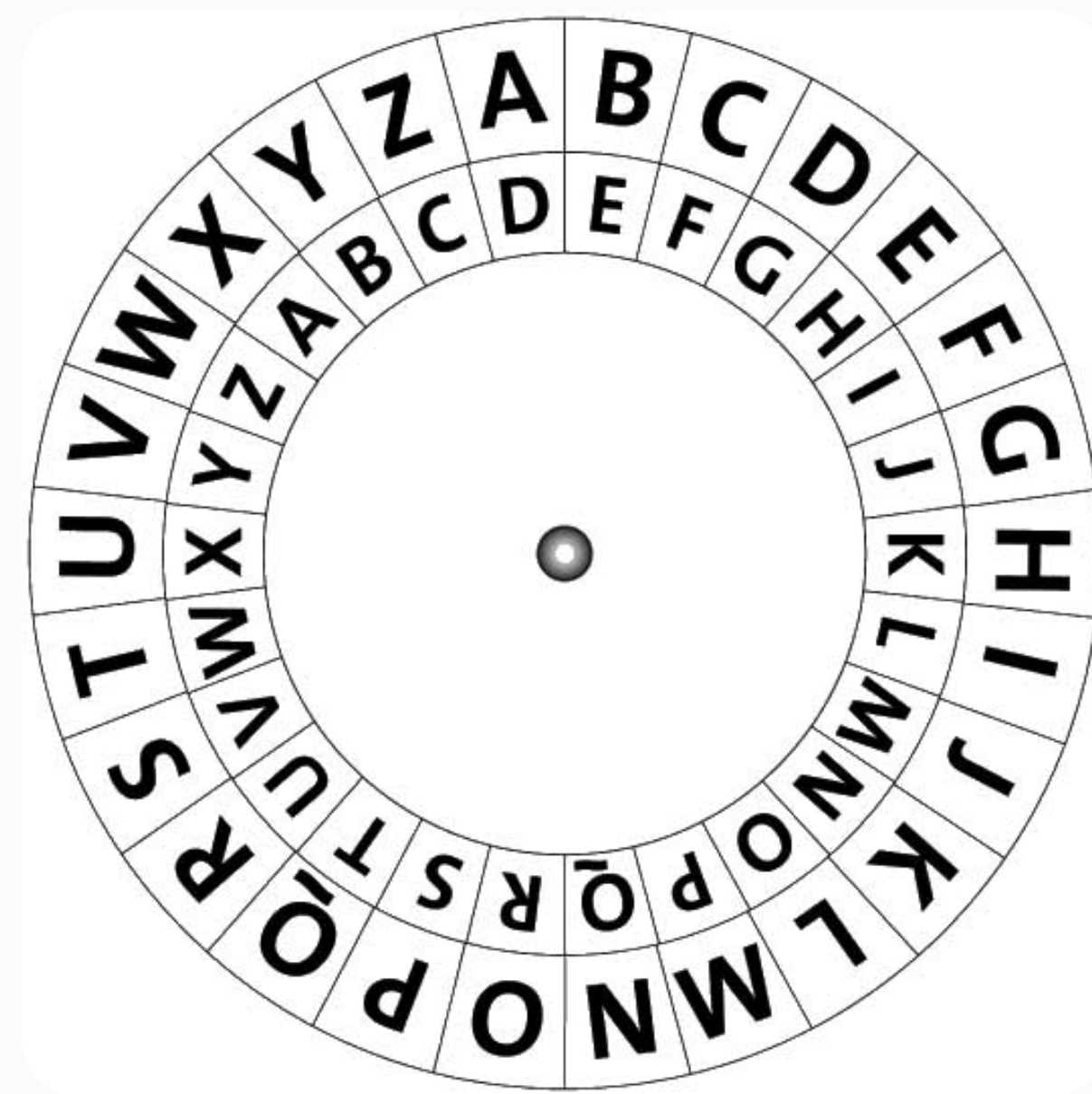
Строка; сдвиг (k).

При $k = 3$:

- "а" -> "г"
- "б" -> "д" ...

Выходные данные:

- зашифрованная строка



Разбор задачи

alphabet = 'абвгдеёжзийклмнопрстуфхцчшщъыьэюя'



Пример 1

Строка: 'питон'

Сдвиг: 3

Зашифрованная строка: 'тлхср'

Алгоритм:

'п' -> индекс('п') -> индекс('п') + 3 = буква шифра ('т')

Разбор задачи

alphabet = 'абвгдеёжзийклмнопрстуфхцчшщъыь^эюя'₃ а

Пример 2

Строка: 'это питон'

Сдвиг: 3

'э' -> индекс('э') -> индекс('э') + 3 = ???

индекс('э') = 30

$30 + 3 = 33$

$33 \% 33 = 0$

0 — это индекс('а')

Новый алгоритм:

'э' -> индекс('п') -> $(\text{индекс('п')} + 3) \% 33 = \text{буква шифра ('а')}$

Если ' ', то вставить ' ' в шифр.

Форматирование строк: `format` и `f-strings`

Роман Булгаков

Преподаватель

Skillbox

Задача «Путь к файлу»

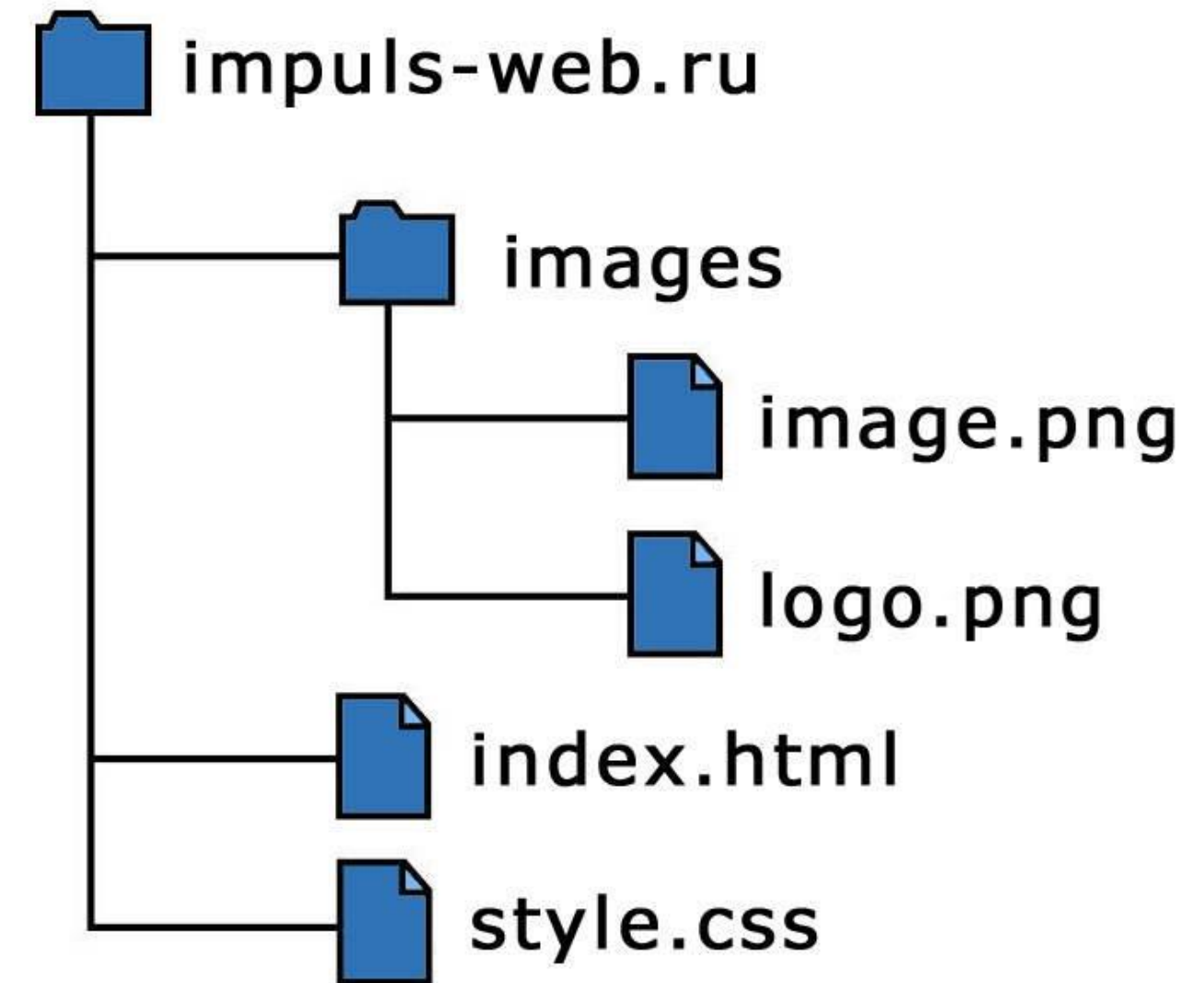
Входные данные:

путь: 'C:/user/docs/folder/new_file.txt'

user и new_file вводит пользователь.

Выходные данные:

НОВЫЙ ПУТЬ.



Форматирование строк

f-strings:

```
path_3 = f'C://{user_name}/docs/folder/{file_name}.txt'
```

Методом format:
(предпочтительный)

```
path = 'C://{user}/docs/folder/{new_file}.txt'.format(  
    user=user_name,  
    new_file=file_name  
)
```


Задача «Поздравления»

Входные данные:

- шаблон поздравления
- список имён (end — конец списка)

Выходные данные:

- поздравление на каждое имя



Итоги урока

- `path = 'C:/{user}/'.format()`
- `path = 'C:/{user}/'.format(user=user_name)`
- `path = 'C:/{0}/{1}'.format(user_name, file)`
- `path = f'C:/{user_name}/'`
- `print(f'Ответ: {1 + 1 + abs(x)}')` — не надо

Методы строк: `split` и `join`

Роман Булгаков

Преподаватель

Skillbox

Соединение строк

```
str_1 = "I love "  
str_2 = "Python."  
  
print(str_1 + str_2)
```

Конкатенация

```
str_1 = "I love "  
str_2 = "Python."  
  
print(''.join([str_1, str_2]))
```

Метод join

Задача «Поздравления»

Входные данные:

- шаблон поздравления (Ф. И. и возраст)
- список из имён и фамилий
- список возрастов

Выходные данные:

- поздравление на каждое имя
- именинники в одну строку



Итоги урока

- `names_lst = input().split()`
- `names_lst = input().split(',')`
- `people_str = ', '.join(people)`
- `age_str = ', '.join(str(5))`

Методы строк: `startswith`, `endswith`, `upper`, `lower`

Роман Булгаков

Преподаватель

Skillbox

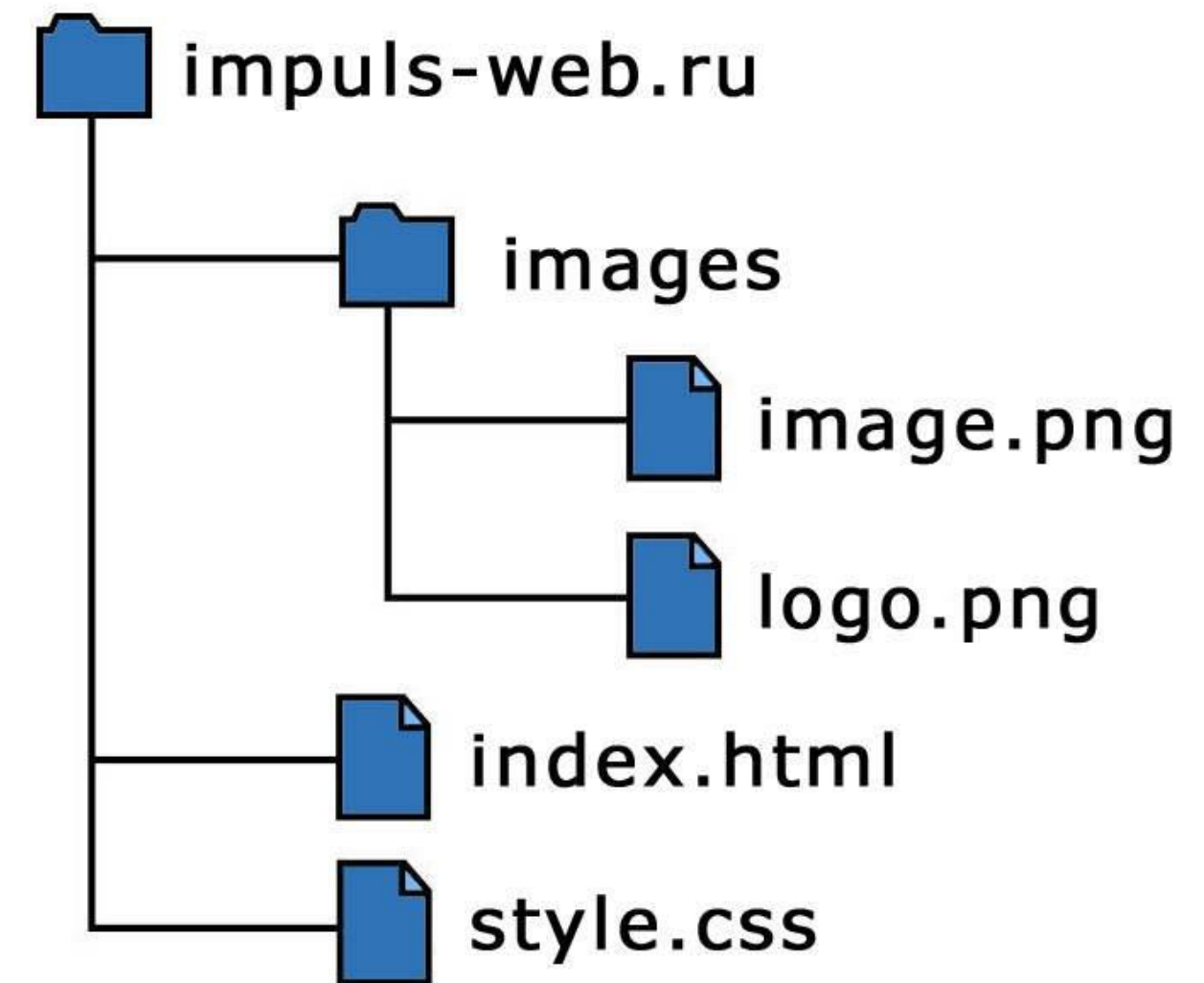
Задача «Путь к файлу»

Входные данные:

- путь: 'C:/user/docs/folder/new_file.txt'
- user
- new_file (с расширением)

Выходные данные:

- если заканчивается на '.txt', то новый путь, иначе ошибка



Задача «Подсчёт слов»

Условия задачи:

- три слова
- текст (в одну строку)
- независимость от регистра

Выходные данные:

- количество каждого из трёх слов в тексте

```
words_list = ['', 0], ['', 0], ['', 0]]

for i_num in range(3):
    print('Введите', i_num+1, 'слово:', end=' ')
    word = input()
    words_list[i_num][0] = word

text = input('Слово из текста: ')
while text != 'end':
    for index in range(3):
        if words_list[index][0] == text:
            words_list[index][1] += 1
    text = input('Слово из текста: ')

print('\nПодсчёт слов в тексте')
for index in range(3):
    print(words_list[index][0], ': ', words_list[index][1])
```

Методы строк: startswith, endswith, upper, lower

Методы строк

```
m find(self, sub, __start, __end)      str
m join(self, __iterable)              str
m capitalize(self)                    str
m casefold(self)                      str
m center(self, __width, __fillchar)   str
m count(self, x, __start, __end)      str
m encode(self, encoding, errors)      str
m endswith(self, suffix, start, end)   str
m expandtabs(self, tabsize)            str
m format_map(self, map)               str
m index(self, sub, __start, __end)    str
m isalnum(self)                       str
m isalpha(self)                       str
m isascii(self)                       str
m isdecimal(self)                     str
m isdigit(self)                       str
m isidentifier(self)                  str
m islower(self)                       str
m isnumeric(self)                     str
m isprintable(self)                   str
m isspace(self)                       str
m istitle(self)                       str
m isupper(self)                       str
m ljust(self, __width, __fillchar)    str
m lower(self)                         str
m lstrip(self, __chars)                str
m maketrans(__x)                       str
m partition(self, __sep)               str
m removeprefix(self, __prefix)         str
m removesuffix(self, __suffix)        str
m replace(self, __old, __new, __count) str
```

Итоги урока

- `path.endswith('.txt')`
- `path.startswith('C:/')`
- `word = input().lower()`
- `word = input().upper()`
- `word = input().lower().split()`

Форматирование строк: подстановки

Роман Булгаков

Преподаватель

Skillbox

Задача «Детали»

Условия задачи:

- количество деталей (int)
- цена детали (float)
- увеличение цены (float)

Выходные данные:

```
На складе 500,000,000 деталей
Каждая деталь стоит 23.86 рублей
Цена увеличилась на 4.567800%
```

Итоги модуля

- `path = 'C:/{user}/'.format(user=user_name)`
`print("{:.2f} рублей".format(price))`
- `names_lst = input().split(',')`
`people_str = ' & '.join(people)`
- `path.endswith('.txt')`
`path.startswith('C:/')`
- `word = input().lower()`
`word = input().upper()`

