

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ  
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ  
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ «СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ  
УНИВЕРСИТЕТ» ИНСТИТУТ ЦИФРОВОГО РАЗВИТИЯ

Отчет о лабораторной работе №7  
по дисциплине основы программной инженерии

Выполнил: Духно Михаил  
Александрович,  
2 курс, группа ПИЖ-б-о-20-1,  
Проверил: Доцент кафедры  
инфокоммуникаций, Воронкин  
Р.А.

Ставрополь, 2022 г

Код программы тестового задания 1

```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-

import json
from jsonschema import validate
from pathlib import Path
import pathlib
import os

FILE_NAME = pathlib.PureWindowsPath(r'C:\Users\я\json_file.json')
SETTINGS_FILE = 'settings.json'

def add_element():
    name = input('Конечный пункт: ')
    num = input('Номер поезда: ')
    tm = input('Время отправления: ')
    trains = {}
    trains['name'] = name
    trains['num'] = int(num)
    trains['tm'] = tm
    schema = ''
    with open('schema.json', 'r') as f:
        schema = json.loads(f.read())
    validate(instance=trains, schema=schema)
    with open(FILE_NAME, 'a') as f:
        f.write(json.dumps(trains) + '\n')

def find_train(num):
    with open(FILE_NAME, 'r') as f:
        trains = f.readlines()
        for dcts in trains:
            dcts = json.loads(dcts)
            if dcts['num'] == int(num):
                print(
                    f'Конечный пункт: {dcts["name"]} \n'
                    f'Номер поезда: {dcts["num"]} \n'
                    f'Время отправления: {(dcts["tm"])}'
                )
                return
    print('Поезда с таким номером нет')

if __name__ == '__main__':
```

```

print('LOADING...')
with open(SETTINGS_FILE, 'r') as f:
    settings = json.loads(f.read())
    if settings['gitignore'] == False:
        path = Path(__file__).resolve()
        print(path.parents[1])
        par_path = path.parents[1]
        with open(str(par_path) + '\\.gitignore', 'a') as gig:
            gig.write('\n' + '*.json' + '\n' + '*.env')
with open(SETTINGS_FILE, 'w') as f:
    f.write(json.dumps({'gitignore': True}))

print('Hello!')

flag = True
while flag:
    print('1. Добавить новый поезд')
    print('2. Вывести информацию о поезде')
    print('3. Выход из программы')
    com = int(input('введите номер команды: '))
    if com == 1:
        add_element()
    elif com == 2:
        train_num = input('Введите номер поезда: ')
        find_train(train_num)
    elif com == 3:
        flag = False

```

Код программы тестового задания 2

```

#!/usr/bin/env python3
# -*- coding: utf-8 -*-

import os

def show_all_subdir(path, space_factor=0):
    file_list = os.listdir(path)
    for file in file_list:
        space = ' ' * space_factor
        print(space + file)
        try:
            show_all_subdir(path + f'/{file}',
space_factor=space_factor + 1)
        except:
            continue

```

```
def main():
    current_path = 'C:/Users'
    while True:
        comand = input(current_path + '>').split()
        match comand[0]:
            case 'cd':
                current_path = comand[1]
            case 'list':
                show_all_subdir(current_path)

if name == 'main':
    main()
```

1. Какие существовали средства для работы с файловой системой до Python 3.4?

До Python 3.4 работа с путями файловой системы осуществлялась либо с помощью методов строк, либо с помощью модуля `os.path`.

2. Что регламентирует PEP 428?

С путями, представленными строками, возможно, но обычно плохая идея, использовать обычные строковые методы. Например, вместо того, чтобы соединять два пути с `+`, как обычные строки, вы должны использовать `os.path.join()`, который соединяет пути, используя правильный разделитель пути в операционной системе. Напомним, что Windows использует `\`, а Mac и Linux используют `/` в качестве разделителя. Это различие может привести к трудно обнаруживаемым ошибкам, таким как наш первый пример во введении, работающий только для путей Windows. Модуль `pathlib` был введен в Python 3.4 (PEP 428) для решения этих проблем. Он объединяет необходимые функции в одном месте и делает его доступным через методы и свойства простого в использовании объекта `Path`.

3. Как осуществляется создание путей средствами модуля `pathlib`?

Все, что вам действительно нужно знать, это класс `pathlib.Path`. Есть несколько разных способов создания пути. Прежде всего, существуют `classmethods` наподобие `.cwd()` (текущий рабочий каталог) и `.home()` (домашний каталог вашего пользователя). Путь также может быть явно создан из его строкового представления.

4. Как получить путь дочернего элемента файловой системы с помощью модуля `pathlib`?

5. Как получить путь к родительским элементам файловой системы с помощью модуля `pathlib`?

`.parent`: каталог, содержащий файл, или родительский каталог, если путь является каталогом

6. Как выполняются операции с файлами с помощью модуля `pathlib`?

Чтобы переместить файл, используйте `.replace()`. Обратите внимание, что если место назначения уже существует, `.replace()` перезапишет его.

7. Как можно выделить компоненты пути файловой системы с помощью модуля `pathlib`?

8. Как выполнить перемещение и удаление файлов с помощью модуля `pathlib`?

9. Как выполнить подсчет файлов в файловой системе?

Есть несколько разных способов перечислить много файлов. Самым простым является метод `.iterdir()`, который перебирает все файлы в данном каталоге. В следующем примере комбинируется `.iterdir()` с классом `collection.Counter` для подсчета количества файлов каждого типа в текущем каталоге.

10. Как отобразить дерево каталогов файловой системы?

```
def tree(directory):  
    print(f'+ {directory}')  
    for path in sorted(directory.rglob('*')):  
        depth = len(path.relative_to(directory).parts)  
        spacer = ' ' * depth  
        print(f'{spacer}+ {path.name}')
```

11. Как создать уникальное имя файла?

```
def unique_path(directory, name_pattern):  
    counter = 0  
    while True:  
        counter += 1  
        path = directory/name_pattern.format(counter)  
        if not path.exists():  
            return path
```

```
path = unique_path(pathlib.Path.cwd(), 'test{:03d}.txt')
```

12. Каковы отличия в использовании модуля `pathlib` для различных операционных систем?