

**МИНИСТЕРСТВО ЦИФРОВОГО РАЗВИТИЯ, СВЯЗИ И МАССОВЫХ
КОММУНИКАЦИЙ РОССИЙСКОЙ ФЕДЕРАЦИИ**

**Ордена Трудового Красного Знамени федеральное государственное бюджетное
образовательное учреждение высшего образования**

«Московский технический университет связи и информатики»

Кафедра “Математическая кибернетика и информационные технологии”

ОТЧЕТ

по лабораторной работе №3

по дисциплине «Введение в информационные технологии»

Тема: «Работа с файлами в Python: открытие, чтение, запись, работа с исключениями»

Выполнил: студент группы БВТ2505

Вакалюк А.А.

Проверил: Павликов. А.Е.

Москва, 2025

Цель работы:

Освоить принципы определения и использования функций в языке программирования Python, понять механизмы передачи аргументов в функции, научиться применять функции для решения практических задач, а также изучить базовые алгоритмические конструкции

Задание 1:

1. Создать текстовый файл example.txt и заполнить его несколькими строками текста.
2. Написать функцию на Python, которая открывает файл example.txt в режиме чтения и выводит его содержимое на экран.
3. Использовать разные методы чтения файла: чтение всего файла сразу, построчное чтение, реализуйте выбор типа чтения в принимаемых аргументах функции.

Задание 2:

1. Написать программу, которая запрашивает у пользователя текст и записывает его в новый файл user_input.txt.
2. Реализовать функционал добавления текста в существующий файл, не удаляя его предыдущее содержимое.

Задание 3:

1. Модифицировать программу из Задания 1 так, чтобы она корректно обрабатывала исключение, возникающее при попытке открыть несуществующий файл. Вместо вывода ошибки программа должна выводить пользователю понятное сообщение. Использовать в блоке try except следующий класс исключений: FileNotFoundError.

Скриншоты выполнения:

```
idhjkkgkdhgkjdsghkfdgh
lhgjkdfhgkdnucl
ppdfpdpfpdf
cpdpcpdsc

Строка: idhjkkgkdhgkjdsghkfdgh
Строка: lhgjkdfhgkdnucl
Строка: ppdfpdpfpdf
Строка: cdpdpcpdsc
idhjkkgkdhgkjdsghkfdgh
lhgjkdfhgkdnucl
ppdfpdpfpdf
cpdpcpdsc

['idhjkkgkdhgkjdsghkfdgh', 'lhgjkdfhgkdnucl', 'ppdfpdpfpdf', 'cpdpcpdsc']
Введите текст: ало
Введите текст для добавления: лол
idhjkkgkdhgkjdsghkfdgh
lhgjkdfhgkdnucl
ppdfpdpfpdf
cpdpcpdsc
```

Исходный код программы:

1 ЗАДАНИЕ

Чтение всего файла

```
with open('C:/Users/andrey/Desktop/example.txt', 'r') as file:  
    content = file.read()  
    print(content)
```

Чтение строк

```
with open('C:/Users/andrey/Desktop/example.txt', 'r') as file:  
    for line in file:  
        print(f"Строка: {line.rstrip()}")
```

Чтение по выбору

```
def read_file(filename, method='all'):  
    with open(filename, 'r') as file:  
        if method == 'all':  
            return file.read()  
        elif method == 'line':  
            return [line.strip() for line in file]  
  
print(read_file('C:/Users/andrey/Desktop/example.txt', 'all'))  
print(read_file('C:/Users/andrey/Desktop/example.txt', 'line'))
```

2 ЗАДАНИЕ

def write_new_text():

```
    text = input("Введите текст для записи в файл (оставьте пустым, чтобы не менять файл): ")  
    if text.strip() != "":  
        with open('user_input.txt', 'w', encoding='utf-8') as file:  
            file.write(text)  
        print("Текст записан в файл user_input.txt")
```

```

else:

    print("Пустой ввод — файл не изменён.")
write_new_text()

# Если добавить запись
filename2 = 'user_input.txt'

text = input("Введите текст для добавления в файл(Добавится на следующей строке,
старый текст сохраниться. Если ничего не хотите добавлять, то просто пропустите,
нажав ENTER): ")

def append_text():

    with open(filename2, 'a', encoding='utf-8') as file:

        file.write('\n' + text)

    print("Текст добавлен в файл user_input.txt")
append_text()

```

3 ЗАДАНИЕ

```

def read_file_safe(filename, method='all'):

    try:

        if method == 'all':

            with open(filename, 'r') as file:

                content = file.read()

                print(content)

            elif method == 'line':

                with open(filename, 'r') as file:

                    for line in file:

                        print(line.strip())

        except FileNotFoundError:

            print(f"Ошибка: файл {filename} не найден!")

read_file_safe('C:/Users/andrey/Desktop/example.txt', 'line')

```

Заключение

В ходе лабораторной работы были изучены основные операции работы с файлами в Python: открытие, чтение, запись и добавление данных. Были рассмотрены различные режимы работы с файлами и способы чтения содержимого, включая построчное чтение и чтение всего файла целиком.

Также были получены навыки обработки ошибок с использованием конструкции try-except. Реализована корректная реакция программы на попытку открыть несуществующий файл, что повышает надежность и устойчивость программного кода.