|  |
| --- |
| МИНОБРНАУКИ РОССИИ |
| **Федеральное государственное бюджетное образовательное учреждение**  **высшего образования**  **«МИРЭА – Российский технологический университет»**  **РТУ МИРЭА** |

Институт кибербезопасности и цифровых технологий

Кафедра КБ-2 «Информационно-аналитические системы кибербезопасности»

**ПРАКТИЧЕСКАЯ РАБОТА №3**

по дисциплине

**«Проектирование и разработка безопасного программного обеспечения информационно-аналитических систем**»

**Выполнил**

Студент 5 курса: Полянская П.А.

Группы: БИСО-01-20

Специальности: 10.05.04

Москва, 2024 г.

**1 Код по заданиям**

**1.1 Задание 1**

import psutil

inf\_log\_disk = psutil.disk\_partitions()

print("Информация о диске, и т.д.", inf\_log\_disk)

**1.2 Задание 2**

import os, shutil

filename = "text.txt"

text\_file = open(filename, "w")

text\_file.close()

text\_file = open(filename, "w")

stroka = "KB-2 luchshe, chem KB-3"

text\_file.write(stroka)

text\_file.close()

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

    for line in file:

        print(line)

text\_file.close()

os.remove(filename)

**1.3 Задание 3**

import json

import os

d = {

"data": "Click Here",

"size": 36,

"style": "bold",

"name": "text1",

"hOffset": 250,

"vOffset": 100,

"alignment": "center",

"onMouseUp": "sun1.opacity = (sun1.opacity / 100) \* 90;"

}

filename = "sample.json"

with open(filename, "w") as outfile:

    json.dump(d, outfile, indent=4, ensure\_ascii=False)

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

    for line in file:

        print(line)

file.close()

os.remove(filename)

**1.4 Задание 4**

import xml.etree.ElementTree as ET

import os

element1=ET.Element('config')

element1\_1=ET.SubElement(element1, 'database')

element1\_1\_1=ET.Element('host')

element1\_1\_1.text='localhost'

element1\_1.append(element1\_1\_1)

element1\_1\_2=ET.SubElement(element1\_1, 'port')

element1\_1\_2.text = '3306'

element1\_1\_3 = ET.SubElement(element1\_1, 'password')

element1\_1\_3.text = '12345'

ET.dump(element1)

filename = 'test.xml'

ET.ElementTree(element1).write(filename)

tree = ET.parse(filename)

root = tree.getroot()

password\_element = root.find('.//password')

new\_znach = input("Введите: ")

if password\_element is not None:

    password\_element.text = new\_znach *# Новое значение*

tree.write(filename)

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

    for line in file:

        print(line)

file.close()

os.remove(filename)

**1.5 Задание 5**

import zipfile

import os

filename = 'my\_archive.zip'

*# Создать архив в формате zip*

with zipfile.ZipFile(filename, 'w') as zip\_file:

*# Добавить файл, выбранный пользователем, в архив*

    file\_path = input("Введите путь к файлу для добавления в архив: ")

    zip\_file.write(file\_path)

*# Разархивировать файл*

with zipfile.ZipFile(filename, 'r') as zip\_file:

*# Вывести данные о файле*

    file\_info = zip\_file.getinfo("errorsound.mp3")

    print(**f**"Имя файла: {file\_info.filename}")

    print(**f**"Размер файла: {file\_info.file\_size} байт")

    print(**f**"Дата создания файла: {file\_info.date\_time}")

*# Разархивировать файл*

    zip\_file.extractall()

os.remove(filename)

**2 Обобщенный код**

import sys

import psutil

from pathlib import Path

import os

import shutil

import json

import xml.etree.ElementTree as ET

import zipfile

**def** main():

    while True:

        print("Выберите операцию:")

        print("1. Информация о логических дисках.")

        print("2. Работа с файлами.")

        print("3. Работа с форматом JSON.")

        print("4. Работа с форматом XML.")

        print("5. Создание ZIP архива.")

        choice = input("Ваш выбор: ")

        if choice == '1':

            print\_disk\_info()

        elif choice == '2':

            file\_operations()

        elif choice == '3':

            json\_operations()

        elif choice == '4':

            xml\_operations()

        elif choice == '5':

            create\_zip\_archive()

        else:

            break

**def** print\_disk\_info():

    inf\_log\_disk = psutil.disk\_partitions()

    print("Информация о диске, и т.д.", inf\_log\_disk)

**def** file\_operations():

*# Получение имени файла от пользователя*

    filename = input('Введите имя файла: ')

*# Создание файла*

    try:

        with open(filename, 'w'):

            pass

        print(**f**'Создали файл {filename}.')

    except FileExistsError:

        print(**f**'Файл {filename} уже существует.')

*# Запись строки в файл*

    string\_to\_write = input('Введите строку для записи в файл: ')

    with open(filename, 'a+') as f:

        f.write(string\_to\_write + '\n')

        print(**f**'Записали строку "{string\_to\_write}" в файл {filename}.')

*# Чтение файла в консоль*

    with open(filename, 'r') as f:

        content = f.read()

    print(content)

*# Удаление файла*

    while True:

        w = int(input("0-удалить файл, 1-НЕ удалять"))

        if (w in {0,1}):

            break

        else:

            print("Неверное значение. введите ещё раз.\n")

    if w==0:

        try:

            os.remove(filename)

            print(**f**'Удалили файл {filename}.')

        except OSError:

            print(**f**'Ошибка при удалении файла {filename}. Возможно, файл отсутствует.')

**def** json\_operations():

*# Создание файла в формате JSON*

    filename = input('Введите имя файла: ')

    data = {}

    data['key'] = input('Введите значение ключа: ')

*# Сериализация данных в формат JSON*

    with open(filename, 'w') as f:

        json.dump(data, f)

    print(**f**'Записали данные в файл {filename}.')

*# Чтение файла*

    with open(filename, 'r') as f:

        loaded\_data = json.load(f)

    print(loaded\_data)

*# Удаление файла*

    while True:

        w = int(input("0-удалить файл, 1-НЕ удалять"))

        if (w in {0,1}):

            break

        else:

            print("Неверное значение. введите ещё раз.\n")

    if w==0:

        try:

            os.remove(filename)

            print(**f**'Удалили файл {filename}.')

        except OSError:

            print(**f**'Ошибка при удалении файла {filename}. Возможно, файл отсутствует.')

**def** xml\_operations():

*# Создание файла в формате XML*

    filename = input('Введите имя файла: ')

    root = ET.Element('root')

    tree = ET.ElementTree(root)

    tree.write(filename)

    print(**f**'Записали данные в файл {filename}.')

*# Ввод новых данных и запись их в файл*

    new\_element = input('Введите элемент для добавления: ')

    element = ET.SubElement(root, new\_element)

    element.text = input('Введите значение элемента: ')

    tree.write(filename, encoding='utf-8', method="xml")

    print(**f**'Добавили элемент "{new\_element}" в файл {filename}.')

*# Чтение файла*

    tree = ET.parse(filename)

    root = tree.getroot()

    for child in root:

        print(child.tag, child.text)

*# Удаление файла*

    while True:

        w = int(input("0-удалить файл, 1-НЕ удалять"))

        if (w in {0,1}):

            break

        else:

            print("Неверное значение. введите ещё раз.\n")

    if w==0:

        try:

            os.remove(filename)

            print(**f**'Удалили файл {filename}.')

        except OSError:

            print(**f**'Ошибка при удалении файла {filename}. Возможно, файл отсутствует.')

**def** create\_zip\_archive():

    archive\_name = input('Введите имя архива: ')

    archive\_path = Path(archive\_name)

*# Добавление файлов в архив*

    files\_to\_add = []

    while True:

        file\_to\_add = input('Введите путь к файлу для добавления в архив: ')

        if not file\_to\_add or file\_to\_add == 'q':

            break

        files\_to\_add.append(Path(file\_to\_add))

    with zipfile.ZipFile(str(archive\_path), 'w') as zf:

        for file in files\_to\_add:

            zf.write(file, arcname=file.name)

    print(**f**'Создан архив {archive\_path}.')

*# Разархивирование файла*

    unzip\_dir = input('Введите путь для разархивирования: ')

    if unzip\_dir:

        archive\_path.rename(unzip\_dir)

        print(**f**'Архив распакован в {unzip\_dir}.')

*# Удаление файла*

    while True:

        w = int(input("0-удалить файл, 1-НЕ удалять"))

        if (w in {0,1}):

            break

        else:

            print("Неверное значение. введите ещё раз.\n")

    if w==0:

        try:

            os.remove(archive\_path)

            print(**f**'Удалили архив {archive\_path}.')

        except OSError:

            print(**f**'Ошибка при удалении архива {archive\_path}. Возможно, архив отсутствует.')

if \_\_name\_\_ == '\_\_main\_\_':

    main()

**3 Скриншоты работы**

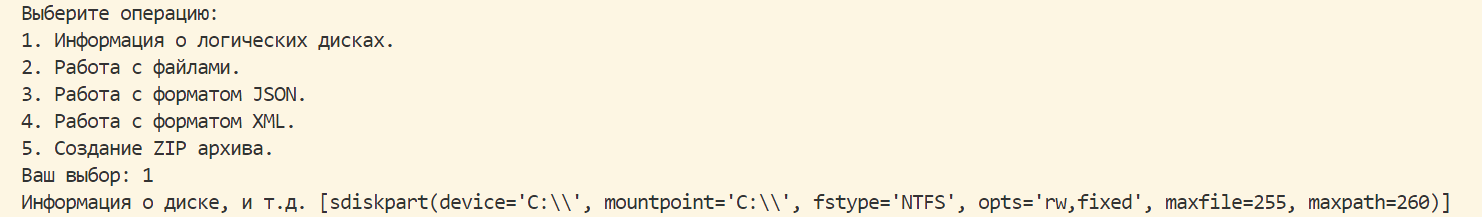


Рисунок 1 – Информация о диске

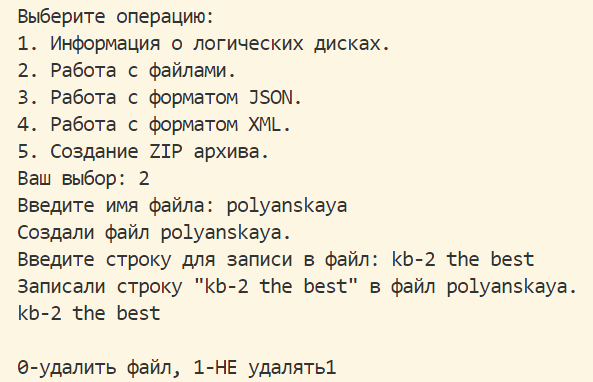


Рисунок 2 – Работа с файлами

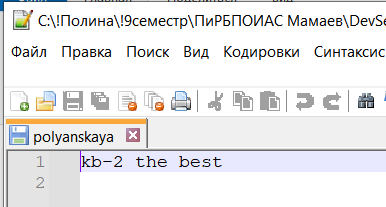


Рисунок 2 – Файл

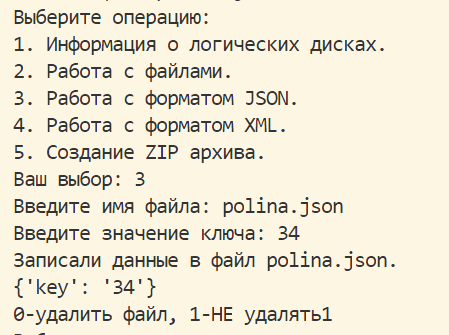


Рисунок 3 – Работа с json

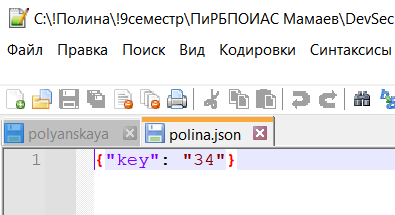


Рисунок 4 – Json-файл

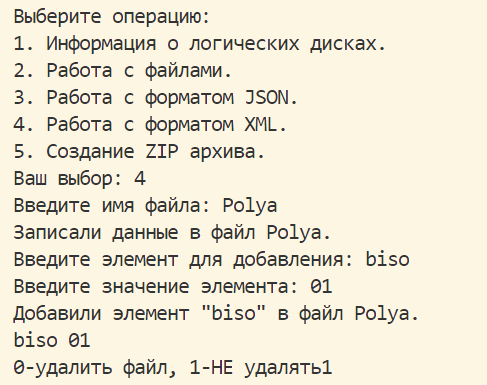


Рисунок 5 – Работа с XML

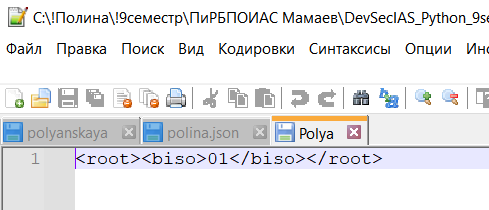


Рисунок 6 – XML-файл

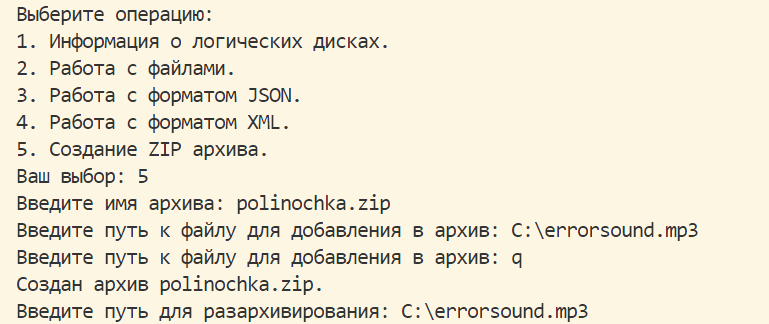


Рисунок 7 – Работа с архивом

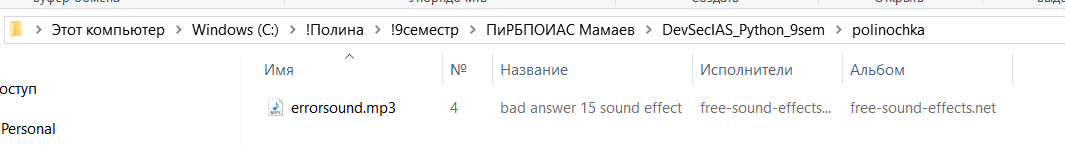


Рисунок 8 – Разархивировано