

|  |  |
| --- | --- |
| МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ  федеральное государственное бюджетное образовательное учреждение высшего образования  **«МОСКОВСКИЙ АВТОМОБИЛЬНО-ДОРОЖНЫЙ ГОСУДАРСТВЕННЫЙ**  **ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ (МАДИ)»** |  |

**Кафедра «Высшая математика»**

**Лабораторная работа № 2**

по дисциплине

«Программирование для ЭВМ»

на тему:

**«**Работа с файлами**»**

**Выполнил:**

Учебная группа: 1бПМ   
ФИО: Кузнецов А.-С. О.

Подпись: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Руководитель лабораторной работы:**

Должность: старший преподаватель

Звание: б/з

ФИО: Кутейников И. А.

Подпись: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

«\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_20\_\_г.

Целью:

Реализовать простейший файловый менеджер. (Для Java использовать пакеты

java.io|java.nio, класс File|Files). Программа должна обладать следующим

функционалом:

1 Выбор файла или каталога для работы;

2 Вывод абсолютного пути для текущего файла или каталога;

3 Вывод содержимого каталога;

4 Вывод всей возможной информации для заданного файла;

5 Изменение имени файла или каталога;

6 Создание нового файла или каталога по заданному пути;

7 Создание копии файла по заданному пути;

8 Вывод списка файлов текущего каталога, имеющих расширение,

задаваемое пользователем;

9 Удаление файла или каталога;

10.Поиск файла или каталога в выбранном каталоге;

Результаты лабораторной работы оформить в виде отчета с результатами

работы программы.

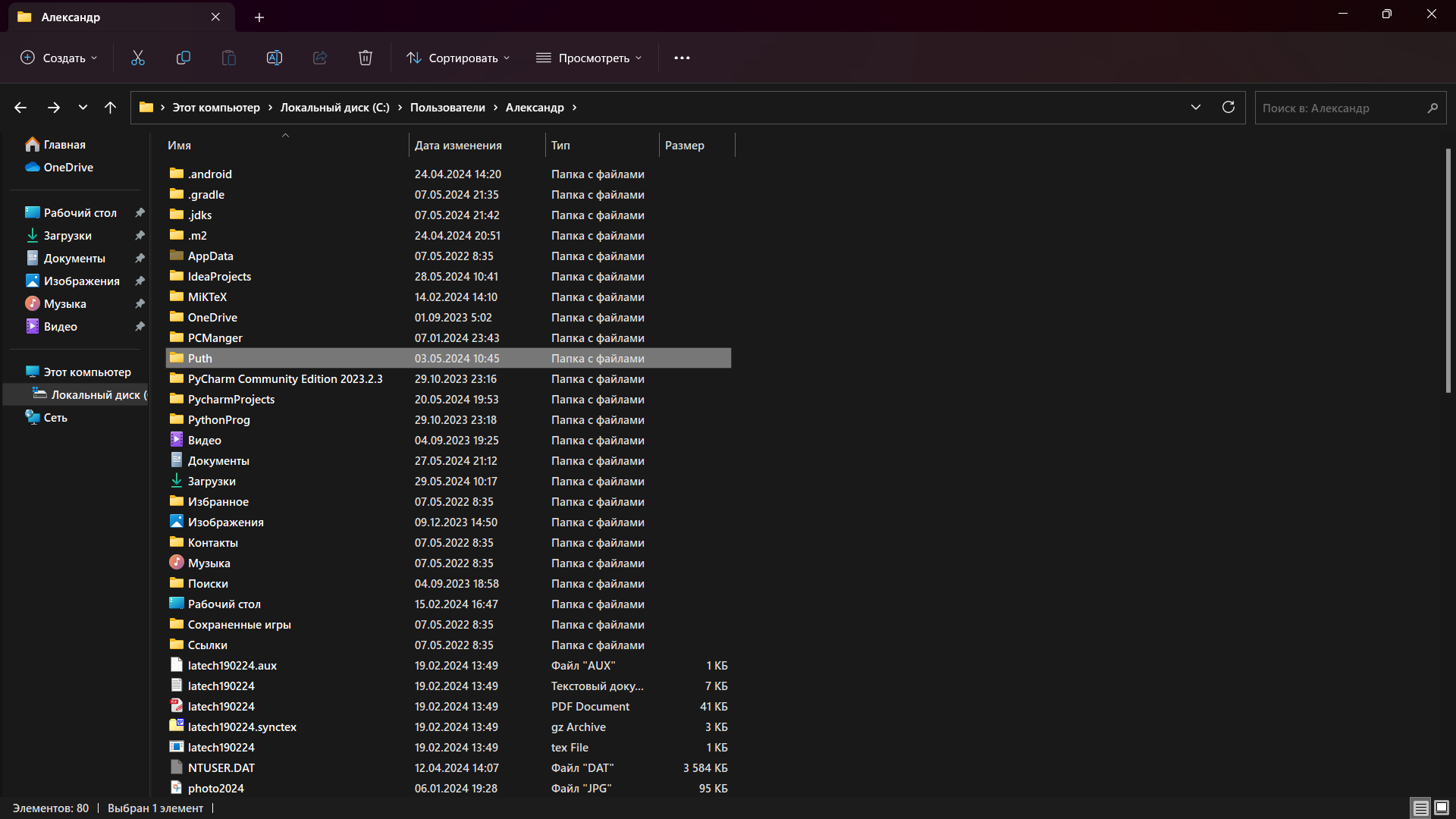
Код:

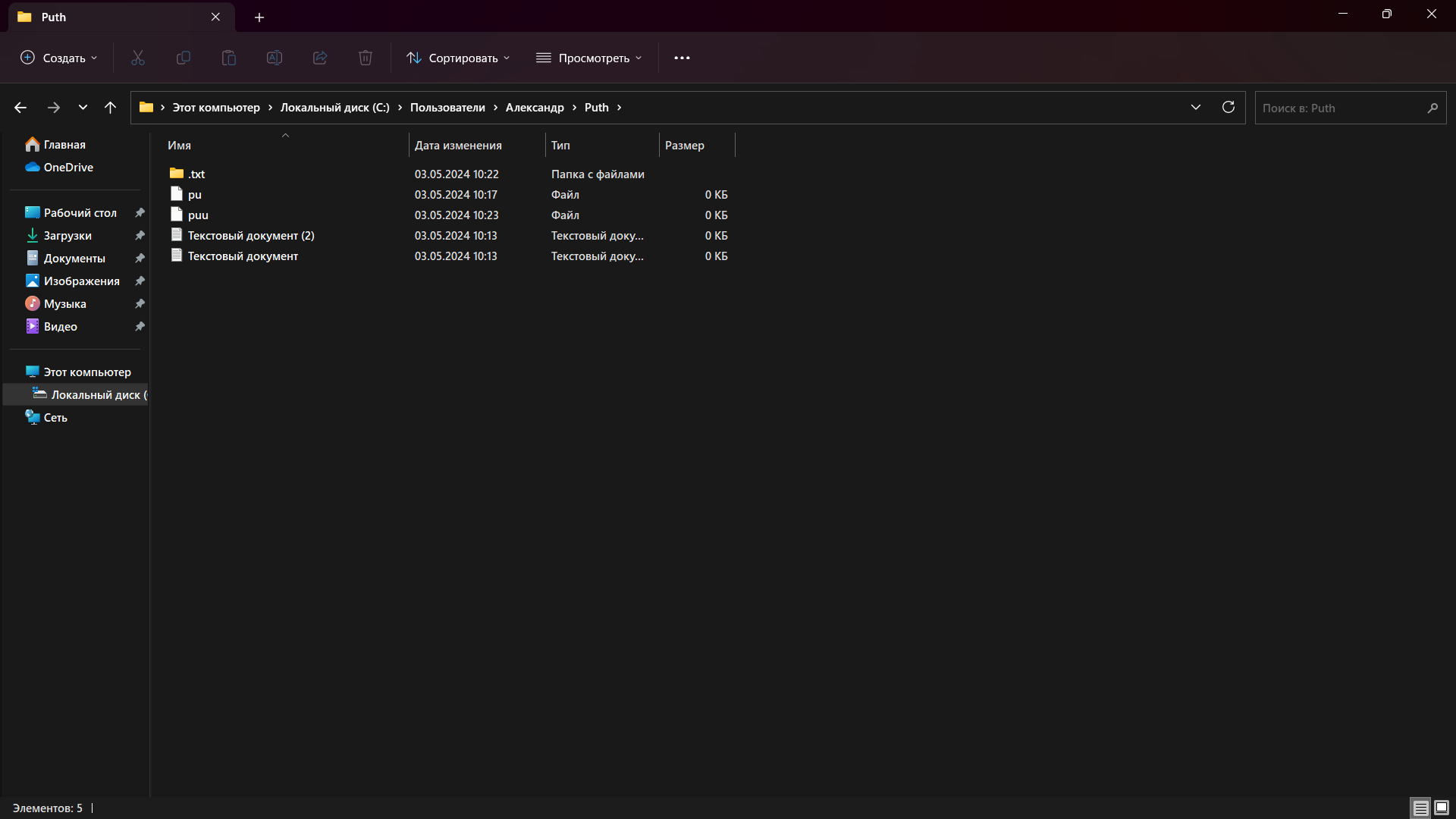
System.*out*.println("\nМеню файлового менеджера:");  
System.*out*.println("1. Показать абсолютный путь текущего файла/каталога");  
System.*out*.println("2. Показать содержимое каталога");  
System.*out*.println("3. Показать информацию о указанном файле/каталоге");  
System.*out*.println("4. Изменить имя файла/каталога");  
System.*out*.println("5. Создать новый файл/каталог в указанном пути");  
System.*out*.println("6. Создать копию файла в указанном пути");  
System.*out*.println("7. Перечислить файлы с указанным расширением в текущем каталоге");  
System.*out*.println("8. Удалить файл/каталог");  
System.*out*.println("9. Поиск файла/каталога в текущем каталоге");  
System.*out*.println("10. Выход");

…

private static void showAbsolutePath() {  
 System.*out*.println("Абсолютный путь: " + *currentFile*.getAbsolutePath());  
 }  
  
 private static void showContentOfDirectory() {  
 if (*currentFile*.isDirectory()) {  
 File[] files = *currentFile*.listFiles();  
 System.*out*.println("Содержимое каталога:");  
 if (files != null) {  
 for (File file : files) {  
 System.*out*.println(file.getName());  
 }  
 }  
 } else {  
 System.*out*.println("Текущий файл не является каталогом.");  
 }  
 }  
  
 private static void showInfoForFile() {  
 File file = new File(*currentFile*.getAbsolutePath());  
 if (file.exists()) {  
 System.*out*.println("Имя файла/каталога: " + file.getName());  
 System.*out*.println("Путь: " + file.getAbsolutePath());  
 System.*out*.println("Размер: " + file.length() + " байт");  
 if (file.isDirectory()) {  
 System.*out*.println("Тип: Каталог");  
 } else {  
 System.*out*.println("Тип: Файл");  
 }  
 } else {  
 System.*out*.println("Файл/каталог не существует.");  
 }  
 }  
  
 private static void changeFileName() {  
 System.*out*.println("Введите новое имя для файла/каталога:");  
 String newName = *currentFile*.getParent() + "\\" + *scanner*.nextLine();  
 File newFile = new File(newName);  
 if (*currentFile*.renameTo(newFile)) {  
 *currentFile* = newFile;  
 System.*out*.println("Имя файла/каталога успешно изменено.");  
 } else {  
 System.*out*.println("Ошибка при изменении имени файла/каталога.");  
 }  
 }  
  
 private static void createNewFileOrDirectory() {  
 System.*out*.println("Введите путь для нового файла/каталога:");  
 String newPath = *scanner*.nextLine();  
 File newFile = new File(newPath);  
  
 System.*out*.println("Выберите что создать (1 - файл, 2 - каталог):");  
 int choice = *scanner*.nextInt();  
 *scanner*.nextLine(); // Чтение символа новой строки после ввода числа  
  
 try {  
 if (choice == 1) {  
 if (newFile.createNewFile()) {  
 System.*out*.println("Новый файл успешно создан.");  
 } else {  
 System.*out*.println("Ошибка при создании нового файла.");  
 }  
 } else if (choice == 2) {  
 if (newFile.mkdir()) {  
 System.*out*.println("Новый каталог успешно создан.");  
 } else {  
 System.*out*.println("Ошибка при создании нового каталога.");  
 }  
 } else {  
 System.*out*.println("Некорректный выбор. Пожалуйста, выберите 1 для файла или 2 для каталога.");  
 }  
 } catch (IOException e) {  
 System.*out*.println("Произошла ошибка.");  
 e.printStackTrace();  
 }  
 }  
 private static void createCopyOfFile() {  
 System.*out*.println("Введите путь для копии файла:");  
 String copyPath = *scanner*.nextLine();  
 File copyFile = new File(copyPath);  
 try {  
 Files.*copy*(*currentFile*.toPath(), copyFile.toPath());  
 System.*out*.println("Файл успешно скопирован.");  
 } catch (IOException e) {  
 System.*out*.println("Произошла ошибка.");  
 e.printStackTrace();  
 }  
 }  
  
 private static void listFilesWithExtensionInCurrentDirectory(Scanner scanner) {  
 if (*currentFile*.isDirectory()) {  
 System.*out*.println("Введите расширение файла для поиска:");  
 String extension = scanner.nextLine();  
 File[] files = *currentFile*.listFiles((dir, name) -> name.toLowerCase().endsWith(extension));  
 if (files != null) {  
 System.*out*.println("Файлы с расширением '" + extension + "':");  
 for (File file : files) {  
 System.*out*.println(file.getName());  
 }  
 }  
 } else {  
 System.*out*.println("Текущий файл не является каталогом.");  
 }  
 }  
  
 private static void deleteFileOrDirectory() {  
 System.*out*.println("Введите путь к файлу или папке для удаления:");  
 String path = *scanner*.nextLine();  
 File file = new File(path);  
  
 if (!file.exists()) {  
 System.*out*.println("Файл или папка не существует.");  
 return;  
 }  
  
 if (file.isDirectory()) {  
 *deleteDirectory*(file);  
 } else {  
 if (file.delete()) {  
 System.*out*.println("Файл успешно удален.");  
 } else {  
 System.*out*.println("Ошибка при удалении файла.");  
 }  
 }  
 }  
  
 private static void deleteDirectory(File directory) {  
 File[] files = directory.listFiles();  
 if (files != null) {  
 for (File file : files) {  
 if (file.isDirectory()) {  
 *deleteDirectory*(file);  
 } else {  
 if (file.delete()) {  
 System.*out*.println("Файл успешно удален: " + file.getName());  
 } else {  
 System.*out*.println("Ошибка при удалении файла: " + file.getName());  
 }  
 }  
 }  
 }  
  
 if (directory.delete()) {  
 System.*out*.println("Папка успешно удалена: " + directory.getName());  
 } else {  
 System.*out*.println("Ошибка при удалении папки: " + directory.getName());  
 }  
 }  
  
 private static void searchForFileOrDirectory(Scanner scanner) {  
 if (*currentFile*.isDirectory()) {  
 System.*out*.println("Введите имя файла/каталога для поиска:");  
 String searchName = scanner.nextLine();  
 File[] files = *currentFile*.listFiles((dir, name) -> name.equals(searchName));  
 if (files != null && files.length > 0) {  
 System.*out*.println("Файл/каталог найден: " + files[0].getAbsolutePath());  
 } else {  
 System.*out*.println("Файл/каталог не найден.");  
 }  
 } else {  
 System.*out*.println("Текущий файл не является каталогом.");  
 }  
 }  
}

Результат:





Вывод:

В ходе данной лабораторной работы был построен простейший файловый менеджер. Появилось лучшее понимание того, как работать с библиотекой io, а также произошло знакомство с библиотекой nio. Также появилось лучшее понимание того, как работать с файлами и синтаксисом.