

Tugas 09

Nama : Sefina Ayudia Syauqi

NIM : 245150701111023

Prodi : Teknologi Informasi

Kelas : D

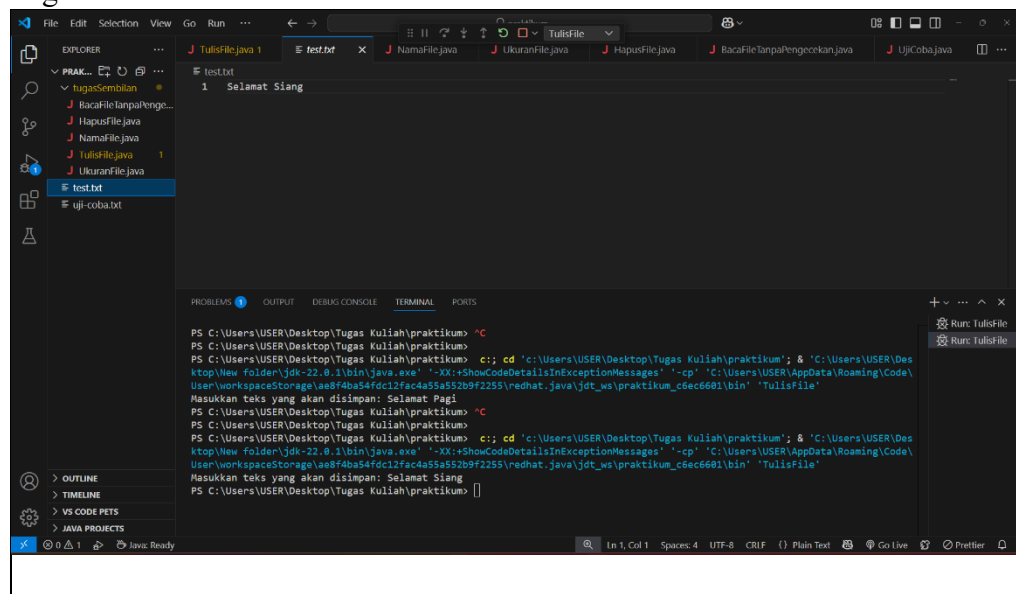
Operasi File

1. Jalankan kode TulisFile1.java beberapa kali dan amati yang terjadi pada *file* yang ditulis. Kemudian, pada baris 13, ubah parameter *false* menjadi *true*. Kemudian jalankan kode tersebut beberapa kali dan amati yang terjadi pada *file* yang ditulis.

- Saat parameter *false* setiap kali program dijalankan hanya teks yang dimasukkan terakhir kali yang tersimpan di file.

Contoh :

1. Saat pertama kali dijalankan dengan memasukan teks “Selamat Pagi” maka *file* akan berisi teks “Selamat Pagi”.
2. Saat kedua kali dijalankan dengan memasukan teks “Selamat Siang” maka *file* akan berganti dengan teks “Selamat Siang” tidak lagi berisi teks “Selamat Pagi”

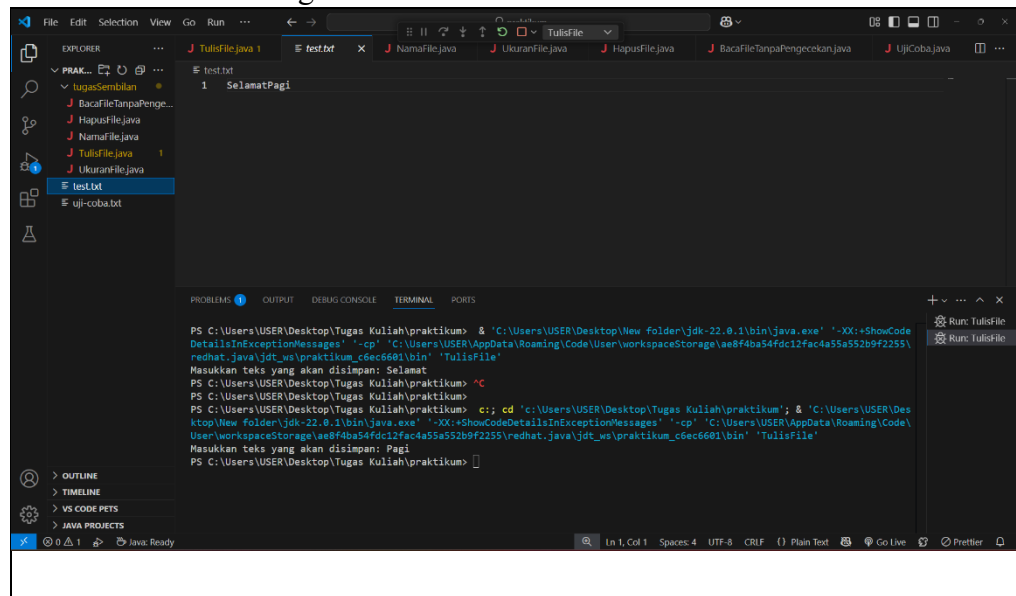


Saat parameter *true* setiap kali program dijalankan setiap teks yang pernah dimasukan akan tersimpan secara berurutan di dalam file.

Contoh:

1. Saat pertama kali dijalankan dengan memasukan teks “Selamat” maka *file* akan berisi teks “Selamat”.

2. Saat dijalankan kedua kalinya dengan memasukkan teks “Pagi” maka *file* akan berisi teks “SelamatPagi”.



2. Buat kode program untuk mendapatkan ukuran *file* dalam satuan KB jika ukuran *file* tersebut < 1 MB dan dalam satuan MB jika ukuran *file* tersebut >= 1 Mb.

- Kode Program

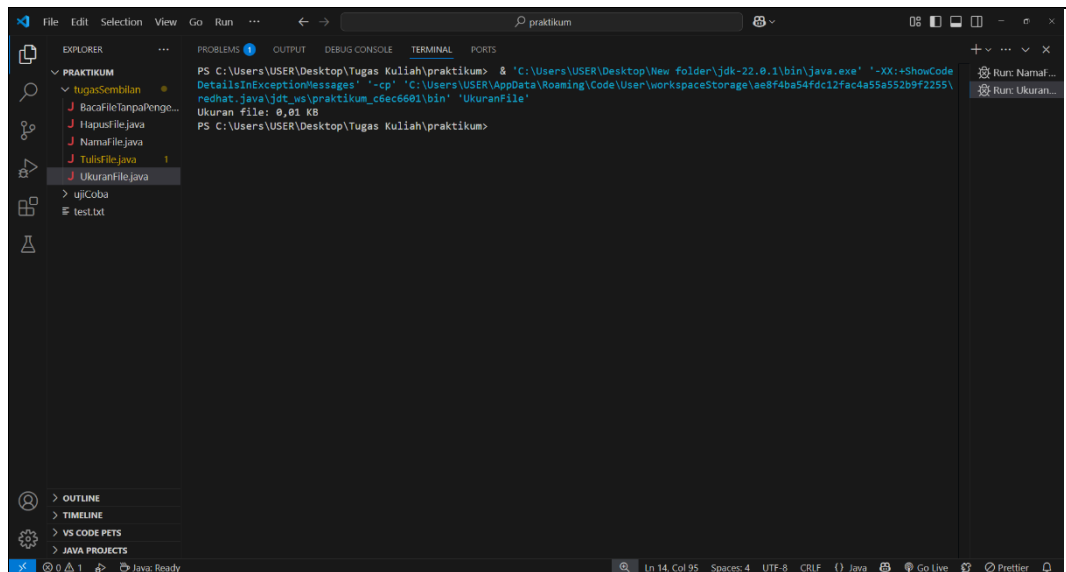
```
import java.io.File;;

public class UkuranFile {

    public static void main(String[] args) {
        var file = new File("test.txt");

        if(file.exists()) {
            long sizeInBytes = file.length(); // ini buat ukuran file dalam byte,
            menggunakan length() karena
            double sizeInKb = sizeInBytes / 1024.0; //dibagi 1024 karena 1 Kb =
            1024 byte
            double sizeInMb = sizeInKb / 1024.0;

            if(sizeInMb >= 1){
                System.out.printf("Ukuran file: %.2f MB\n", sizeInMb);
            }else{
                System.out.printf("Ukuran file: %.2f KB\n", sizeInKb);
            }
        }else{
            System.out.println("File tidak ditemukan!");
        }
    }
}
```



Menampilkan ukuran file yaitu sebesar 0,01KB, karena ukuran < 1 MB maka ukuran dalam satuan KB

3. Buat kode program untuk menampilkan nama dari semua *file* yang ada di dalam suatu direktori. Petunjuk gunakan perulangan dan method `list()` atau `listFiles()`.
 - Kode Program

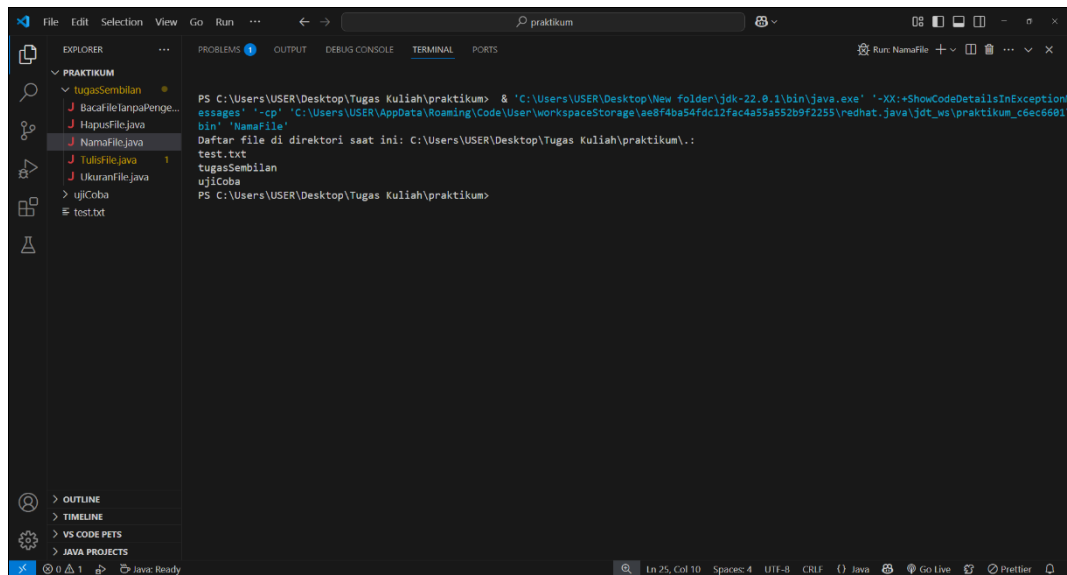
```
import java.io.File;

public class NamaFile {

    public static void main(String[] args) {
        var dir = new File(".");

        if(dir.exists() && dir.isDirectory()){
            System.out.println("Daftar file di direktori saat ini: " +
dir.getAbsolutePath() + ":");

            String[] list = dir.list();
            for(String fileName : list){
                System.out.println(fileName);
            }
        }else{
            System.out.println("Direktori tidak ditemukan!");
        }
    }
}
```



Menampilkan semua daftar nama file di direktori

4. Buat kode program untuk menghapus suatu direktori beserta semua *file* yang ada di dalamnya. Asumsi: di dalam direktori tersebut, hanya ada *file-file* saja, tidak ada subdirektori.

- Kode Program

```

import java.io.File;

public class HapusFile {

    public static void main(String[] args) {
        var dir = new File("ujiCoba");

        if(dir.exists() && dir.isDirectory()){
            File[] files = dir.listFiles();
            if(files != null){
                for(File file : files){
                    if(file.isFile()){
                        file.delete();
                    }
                }
            }

            boolean result = dir.delete();
            System.out.println(result ? "File berhasil dihapus" : "File gagal
dihapus");
        }else{
            System.out.println("File tidak ditemukan!");
        }
    }
}

```



5. Apakah yang salah dengan *statement* berikut? Berikan penjelasan.
`var file = new File("C:\Data\Java\teks.txt");`
 - Kesalahan pada *statement* ini adalah penggunaan *backslash* (\) dalam *path* tanpa di *escape*. Pada java (\) adalah karakter *escape* yang digunakan untuk menuliskan karakter khusus seperti \n, \t, \b dan sebagainya. Sedangkan, \D \J bukanlah *escape* atau tidak valid sehingga dapat menyebabkan kesalahan. Sehingga Solusi yang dapat dilakukan untuk menghindari kesalahan adalah penggunaan *double escape* (\\) atau dengan slash (/). Contoh :
`var file = new File("C:\\Data\\Java\\teks.txt");` atau `var file = new File("C:/Data/Java/teks.txt");`
6. Apa yang akan terjadi jika kita mencoba untuk membaca isi dari suatu *file* tetapi *file* tersebut tidak ada dan kita tidak melakukan pengecekan lebih dahulu?
 - Ketika mencoba membaca isi dari suatu *file* yang tidak ada tanpa pengecekan maka java akan melempar *exception FileNotFoundException*. Hal ini akan menyebabkan program berhenti jika *exception* tidak ditangani dengan *try-catch*. Sehingga Solusi yang dapat dilakukan dengan melakukan pengecekan terlebih dahulu menggunakan `exists()` dan `canRead()`.

