**LAPORAN PRAKTIKUM**

49

**ALGORITMA DAN PEMROGRAMAN LANJUT**

**Pertemuan Ke – 13**



**DISUSUN OLEH :**

**HELDA LUDYA SAFITRI**

**175410186**

**TEKNIK INFORMATIKA**

**STMIK AKAKOM**

**YOGYAKARTA**

**2018**

**PERTEMUAN KE-13**

**I/O FILE**

1. **TUJUAN**
2. Mahasiswa dapat menangani file dengan membuat dan menulis isi file.
3. **DASAR TEORI**

Penyimpanan data didalam variable dan array bersifat temporary atau sementara. Data akan hilang ketika variable lokal keluar dari scope atau ketika program dihentikan. Untuk pemakaian jangka panjang, bahkan setelah program dihentikan komputer dapat menggunakan file. Data yang tersimpan dalam file adalah data persistent. Persistent data adalah data yang akan tetap ada walaupun program tidak aktif.

1. **PEMBAHASAN PRAKTIK**

**Praktik 1**

|  |
| --- |
| import java.io.File;  import java.util.Scanner;  public class FileDemonstration {  public static void main(String[] args) {  Scanner input = new Scanner(System.in);  System.out.print("Enter file or directory name :");  analyzePath(input.nextLine());  }  public static void analyzePath(String path) {  File name = new File(path);  if(name.exists())  {  System.out.printf("%s%s\n%s\n%s\n%s\n%s\n%s\n%s\n%s\n%s",  name.getName(),"exists",  (name.isFile() ?"is a file":"is not a file"),  (name.isDirectory()?"is a directory":"is not a directory"),  (name.isAbsolute()?"is absolute path":"is not absolute path"),  "Last modified:",name.lastModified(),  "Length:",name.length(),  "Path:",name.getPath(),  "Absolute path:",name.getAbsolutePath(),  "Parent:",name.getParent());  if(name.isDirectory())  {  String[] directory = name.list();  System.out.println("\n\nDirectory contents:\n");  for(String directoryName : directory)  System.out.println(directoryName);  }  }  else  {  System.out.printf("%s%s", path, "does not exist.");  }  }  } |

Script diatas merupakan syntax program yang bertujuan untuk menampilkan detail atau rincian nama file atau direktori yang dicari oleh user yang diinputkan pada program. Path yang diinputkan akan dianalisis oleh program. Kemudian jika path ditemukan maka akan ditampilkan nama filenya, ditampilkan apakah itu merupakan file atau bukan, merupakan absolute path atau bukan, waktu last modified, panjang index file, nama pathnya, nama absolute pathnya, dan nama parentnya.

Tetapi jika path yang diinputkan merupakan directory maka akan ditampilkan isi dari directory tersebut. Tetapi jika selain keduanya maka akan ditampilkan nama path+ “does not exist”.

Output 1 :

|  |
| --- |
| Enter file or directory name :C:/Program Files/Java/jdk1.8.0\_102  jdk1.8.0\_102exists  is not a file  is a directory  is absolute path  Last modified:  1485839245628  Length:  4096  Path:  Directory contents:  bin  COPYRIGHT  db  include  javafx-src.zip  jre  lib  LICENSE  README.html  release  src.zip  THIRDPARTYLICENSEREADME-JAVAFX.txt  THIRDPARTYLICENSEREADME.txt  Press any key to continue . . . |

Terlihat pada output diatas bahwa program akan menampilkan rincian dari path yang diinputkan oleh user. Dari mulai exist atau tidak, merupakan file atau bukan, absolute path atau bukan, length, path, absolute path hingga parent.

Output 2 :

|  |
| --- |
| Enter file or directory name :C:/Program Files/Java/jdk1.8.0\_102/README.html  README.htmlexists  is a file  is not a directory  is absolute path  Last modified:  1485839141634  Length:  159 |

Terlihat program tetap mencari file bernama README.html dengan sistem yang sama seperti output pada bagian 1.

Output 3 :

|  |
| --- |
| Enter file or directory name :D:/Listing  D:/Listingdoes not exist.Press any key to continue . . . |

Karena praktikan mencari file yang tidak ada dalam path tersebut maka progam akan langsung menjalankan perintah pada else yaitu dengan menampilkan nama path kemudian ditambah dengan “does not exist”.

**Praktik 2**

|  |
| --- |
| import java.io.\*;  public class DemoStream1 {  public static void main(String[] args){  byte data;  String namaFile="test.txt";  FileOutputStream fout=null;  try{  fout=new FileOutputStream(namaFile, true);  System.out.print("Ketik : ");  data=(byte)System.in.read();  while(data!=(byte)'\r'){  fout.write(data);  data=(byte)System.in.read();  }  }  catch(FileNotFoundException e){  System.out.println("File "+namaFile+"tidak dapat dicreate");  }  catch(IOException e){  System.out.println("Terjadi Exception");  }  finally{  if(fout!=null){  try{  fout.close();  }  catch(IOException e){  System.out.println("Terjadi Exception");  }  }  }  }  } |

Script diatas merupakan program yang dibuat dengan tujuan agar user dapat menginputkan file dengan byte stream atau secara singkatnya user dapat membuat file melalui program ini. Terlihat pada gambar di atas nama file yang dibuat adalah text.html. Kemudian user nantinya akan menginputkan isi file yang ada di file text.html setelah program mengeluarkan tulisan “Ketik :”.

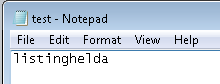
Kemudian terdapat 2 catch yang bernama FileNotFoundException e dan IOException e, masing-masing catch tersebut berisikan perintah penampil “File+namaFile+tidak dapat dicreate” dan “Terjadi Exception”. Terdapat finally yang nantinya perintah didalmnya harus dikerjakan tiap program dijalankan.

Output :

Ketik : listinghelda

Press any key to continue . . .

Terlihat output diatas praktikan membuat sebuah kalimat seperti yang tertera diatas. Kemudian jika praktikan membuka file text.html maka akan tampil gambar berikut :

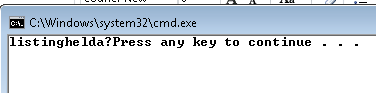


**Praktik 3**

|  |
| --- |
| import java.io.\*;  public class DemoStream2 {  public static void main(String[] args){  byte data;  String namaFile="test.txt";  FileInputStream fin=null;  try{  fin=new FileInputStream(namaFile);  do{  data=(byte)fin.read();  System.out.print((char)data);  }  while(data!=-1);  }  catch(FileNotFoundException e){  System.out.println("File "+namaFile+" tidak ditemukan");  }  catch(IOException e){  System.out.println("Terjadi Exception");  }  finally{  if(fin!=null){  try{  fin.close();  }  catch(IOException e){  System.out.println("Terjadi Exception");  }  }  } |

Script diatas bertujuan untuk menampilkan isi dari file test.txt yang sudah diinputkan user pada praktik sebelumnya, maka pada output akan menampilkan kata sesuai inputan user.

Output :



**Praktik 4**

|  |
| --- |
| import java.nio.file.\*;  public class PathTestCase1 {  public static void main(String[] args){  FileSystem fs=FileSystems.getDefault();  Path p=fs.getPath("D:/VNew/101.txt");  System.out.println("Default Directory ["+p+"]");  }  } |

Script diatas merupakan syntax progam untuk membuat menampilkan sebuah path. Dimana path yang akan ditampilkan adalah D:/VNew/101.txt.

Output :

Default Directory [D:\VNew\101.txt]

Press any key to continue . . .

**Praktik 5**

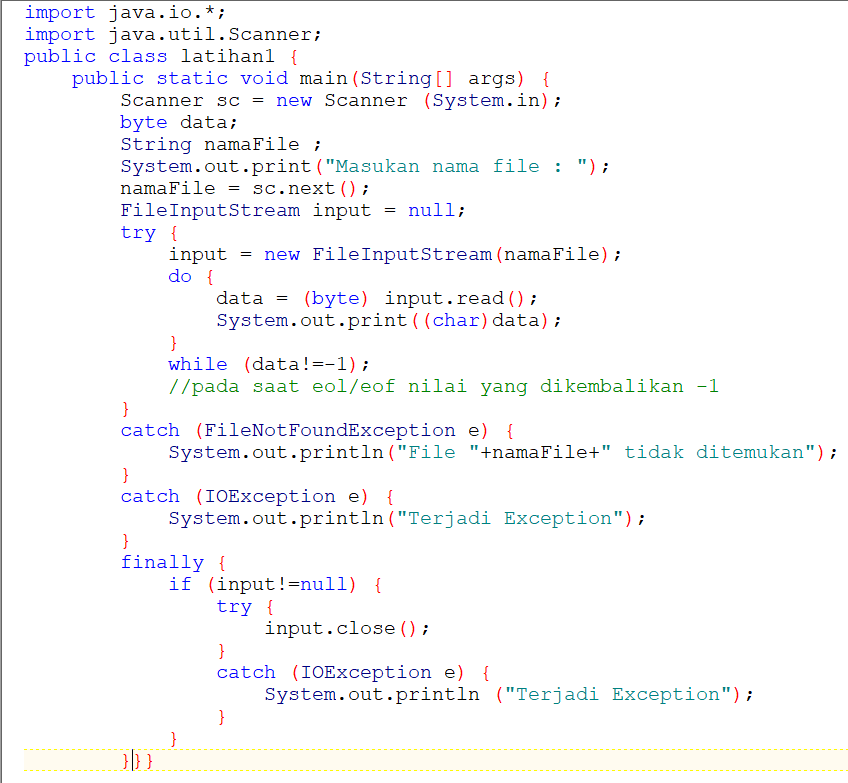
|  |
| --- |
| import java.io.\*;  public class ReadFileOld {  public static void main(String[] args) {  String line="",fileContent="";  try{  BufferedReader fileInput=new BufferedReader(new FileReader(new File ("D:/VNew/101.txt")));  line=fileInput.readLine();  fileContent=line + "\n";  while(line!=null){  line=fileInput.readLine();  if(line!=null)  fileContent=line + "\n";}  fileInput.close();}  catch(EOFException eofe){  System.out.println("No more lines to read.");  System.exit(0);}  catch(IOException ioe){  System.out.println("Error reading file.");  System.exit(0);}  System.out.println(fileContent);  }  } |

Program diatas merupakan progam yang memiliki kemampuan untuk menampilkan isi file text. File yang dimaksud adalah 101.txt . dimana sebelumnya praktikan diharuskan mengisi atau membuat file tersebut sebelum dapat dibaca oleh pragram. Pada program di atas terdapat 2 catch yakni “EOFException” dan “IOException” yang masing-masing memiliki perintah sendiri yakni masing-masing catch berisi perintah penampil “No more lines to read” dan “Error reading file”.

Output :

|  |
| --- |
| Error reading file.  Press any key to continue . . . |

1. **LATIHAN**



Terlihat pada script diatas merupakan syntax program yang memiliki kemampuan untuk mencetak isi file yang akan diinputkan oleh user melalui Scanner. Untuk inputnya praktikan menggunakan input berupa path sehingga progam akan menvari file berdasarkan path yang diinputkan.

1. **KESIMPULAN**

Pada java terdapat cara untuk memembuat, mngambil dan menghapus file yakni menggunakan I/O. Data yang disimpan dalam file tidak akan hilang ketika program dihentikan tetapi jika data disimpan dalam array maka ketika program dihentikan maka datany akan hilang.

1. **LISTING**

Terlampir.