

File Stream

Stream I/O

- Dua cara menyimpan data ke dalam file pada JAVA
 - Binary File
 - TextFile
- Class untuk File Stream I/O
 - FileOutputStream → menggunakan method read()
 - FileInputStream → menggunakan method write()

Contoh FileOutputStream

```
import java.io.*;
public class TulisFile {
    public static void main (Strings []args)
        throws IOException{
        FileOutputStream fos;
        byte[] b = {'t', 'e', 's', 't'};
        int len = b.length;
        fos = new FileOutputStream("myfile.dat");
        fos.write(b,0,len);
        fos.close;
    }
}
```

Contoh FileInputStream

```
import java.io.*;
public class BacaFile {
    public static void main (Strings []args) throws
        IOException {
        FileInputStream fis;
        int len=1200;
        byte[] b = new byte[1200];
        fis = new FileInputStream("myfile.dat");
        fis.read(b,0,len);
        for (int i=0; i<len; i++)
        {
            System.out.print((char) b[i]);
        }
        fis.close;
    }
}
```

Class File

- Memberikan informasi suatu File
- Tidak berhubungan dengan membaca atau menulis file (PERHATIKAN!!!)
- Contoh:

```
File f = new File(namaFile)
If (f.exists() && f.isFile()) {
    System.out.println("OK");
} else {
    System.out.println("File not found");
    System.exit(-1);
}
```

Buffering

- Mempercepat performance baca-tulis file
- Class yang dipergunakan:
- `BufferedReader` → menggunakan method `readLine()` yang mempunyai nilai balik berupa `String`
- `PrintWriter` → menggunakan method `print()` dan `println()` untuk menulis `String` ke `File`

Contoh PrintWriter

```
File f;
FileWriter fw;
PrintWriter fOut;
String tulisan;
try {
    f = new File("sajak.txt");
    fw = new FileWriter(f);
    fOut = new PrintWriter(fw);
}
catch (IOException)
{
    System.out.println("File
    gagal dibuka");
}

tulisan = "Baris pertama";
fOut.println(tulisan);

tulisan = "Baris kedua";
fOut.println(tulisan);

fOut.print("tidak pindah
    baris");
fOut.println("setelah ini
    pindah baris");

fOut.flush();
fOut.close();
```

Contoh BufferedReader

```
File f;
FileReader fr;
BufferedReader fin;
try {
    f = new File("sajak.txt");
    fr = new FileReader(f);
    fin = new BufferedReader(fr);
}
catch (IOException)
{
    System.out.println("File
    gagal dibuka");
}
String tulisan;
```

```
for(;;)
{
    try {
        tulisan = fin.readLine();
    }
    catch (IOException ex) {
    }
    if (tulisan == null)
        break;
    System.out.println(tulisan);
    try {
        fin.close;
    }
    catch IOException ex) {
    }
}
```


Membaca Input dari Keyboard

- Java menyediakan `System.in` sebagai `InputStream` untuk mendapatkan input dari keyboard.
- (catatan: `System.in` tidak jalan pada platform lain yang tidak mempunyai standard input seperti MacIntosh)

- Enkapsulasi:

```
InputStreamReader isr = newInputStreamReader(System.in);  
BufferedReader buf = new BufferedReader(isr);
```

- Untuk mengambil `String` dari keyboard bisa didapat dengan method `readLine()` yang memiliki nilai balik `String`.

Contoh

```
String str = buf.readLine();
```