

# ACCESSO A FILE SEQUENZIALI

# Il file

2

- Struttura per la memorizzazione di informazioni su memoria di massa
  - ▣ File binari e file di testo
- Operazioni:
  - ▣ Apertura in lettura o scrittura
  - ▣ Lettura o scrittura di dati (binari o caratteri)
  - ▣ Chiusura del file
  - ▣ Gestione struttura (ridenominazione, cancellazione, spostamento)

# Ridenominazione e cancellazione di file

3

Utilizzo della classe **File** che rappresenta il file all'interno del filesystem ma non i suoi dati

- ▣ **File f=new File ("prova.txt")**
- ▣ Cancellazione
  - ▣ **f.delete(); //ritorna true se il file è stato cancellato, false altrimenti**
- ▣ rename
  - ▣ **File vecchioNome=new File ("prova.txt")**
  - ▣ **File nuovoNome=new File ("provaNuovo.txt")**
  - ▣ **vecchioNome.renameTo(nuovoNome)**
- ▣ **I metodi non rilanciano eccezioni**
  - ▣ **richiedono: import java.io.\***

# Scrittura di file di testo

4

## Utilizzo della classe **FileWriter** e **PrintWriter**

- Apertura del file in scrittura
  - ▣ **FileWriter f=new FileWriter("prova.txt")**
  - ▣ **PrintWriter out=new PrintWriter(f)**
- Scrittura
  - ▣ **out.println("testo");**
  - ▣ **out.print("testo");**
- Chiusura
  - ▣ **out.close();**
  - ▣ **f.close();**
- Apertura in append
  - ▣ **FileWriter f=new FileWriter("prova.txt",true)**
- **Tutti i metodi rilanciano IOException che quindi va gestita o propagata**
  - ▣ **richiedono: import java.io.\***

# Lettura di file di testo

5

## Utilizzo della classe **FileReader** e **BufferedReader**

- Apertura del file in lettura
  - ▣ **FileReader f=new FileReader("prova.txt")**
  - ▣ **BufferedReader in=new BufferedReader(f)**
- Lettura
  - ▣ **String linea=in.readLine();**//quando null vuol dire che non ci sono altre righe da leggere
- Chiusura
  - ▣ **out.close();**
  - ▣ **f.close();**
- **Tutti i metodi rilanciano IOException che quindi va gestita o propagata**
  - ▣ **richiedono: import java.io.\*;**

# Lettura di file di testo (cicli)

6

- Lettura di tutte le righe di un file
  - **FileReader f=new FileReader("prova.txt")**
  - **BufferedReader in=new BufferedReader(f)**
  - **String linea=in.readLine();**
  - **while (linea !=null){**
    - **//fai qualcosa con linea**
    - **linea=in.readLine();**
  - **}**
  - **in.close();**
  - **f.close();**

# Lettura da tastiera

7

- La tastiera è rappresentata dal campo **System.in** di tipo **InputStream**
- Per poter leggere testo dalla tastiera si usa la classe **InputStreamReader** e **BufferedReader**
  - **InputStreamReader isr=new InputStreamReader(System.in)**
  - **BufferedReader in=new BufferedReader(isr)**
  - **String linea=in.readLine();**

# Esercizi

8

- Scrivere un programma che scrive un file di nome **numeri.txt** che contiene tutti i valori da 0 a 1000 su righe differenti.
  - ▣ In caso di eccezioni il programma deve stampare il messaggio “errore” e terminare.
- Scrivere un programma che legge un valore  $n$  e il nome del File da tastiera (non grafica) e scrive nel file **triangolo.txt** un triangolo di asterischi alto  $n$