



UNIVERSITÀ DEGLI STUDI DI SALERNO  
**DIPARTIMENTO DI INFORMATICA**  
**DIPARTIMENTO DI ECCELLENZA**

# Università di degli Studi di Salerno

## Dipartimento di Informatica

### Programmazione ad Oggetti

*a.a. 2023-2024*

### Gestione del Filesystem

Docente: Prof. Massimo Ficco

E-mail: *mficco@unisa.it*

# Gestione del filesystem

Il filesystem è la parte del sistema operativo che si occupa della gestione dei dati.

I dati sono organizzati in files e cartelle.

Java consente di gestire i files e le cartelle mediante la classe File del package java.io



# La classe File

- La classe File rappresenta un elemento del file system astratto
- Una istanza della classe viene creata a partire da una stringa che ne individua il path
- Su tale oggetto è possibile
  - Controllare se il file corrispondente esiste
  - Controllare se si tratta di file o directory
  - Verificarne le proprietà
  - Crearlo nel caso non esista
  - ....



# Costruttore della classe File

## Costruttori

**File**(File parent, String child)

Crea una nuova istanza dell'oggetto File utilizzando l'oggetto File genitore ed il nome

**File**(String pathname)

Crea una nuova istanza dell'oggetto File utilizzando il percorso completo

**File**(String parent, String child)

Crea una nuova istanza dell'oggetto File utilizzando il percorso della directory genitore ed il nome del file



# Metodi della classe File

Metodi della classe File	
boolean	<u><a href="#">canRead()</a></u> Controlla se si hanno i diritti di lettura sul file
boolean	<u><a href="#">canWrite()</a></u> Controlla se si hanno i diritti di scrittura sul file
boolean	<u><a href="#">delete()</a></u> Cancella il file o la directory corrispondene.
boolean	<u><a href="#">exists()</a></u> Controlla se la directory o il file esiste
<u><a href="#">String</a></u>	<u><a href="#">getName()</a></u> Ritorna il nome del file o della directory priva del path
<u><a href="#">String</a></u>	<u><a href="#">getPath()</a></u> ritorna il path completo come una stringa
boolean	<u><a href="#">isDirectory()</a></u> Controlla se l'oggetto file corrisponde ad una directory
boolean	<u><a href="#">isFile()</a></u> Controlla se l'oggetto File è corrisponde ad un file

# Metodi della classe File

long	<u><a href="#">length()</a></u> Ritorna la lunghezza del file.
<u><a href="#">String[]</a></u>	<u><a href="#">list()</a></u> Ritorna un array di stringhe che corrispondono ai files ed alle cartelle figli dell'oggetto File
boolean	<u><a href="#">mkdir()</a></u> Crea la cartella corrispondente all'oggetto File corrente.
boolean	<u><a href="#">mkdirs()</a></u> Crea la cartella corrispondente all'oggetto File corrente insieme a quelle genitore se non esistono.
boolean	<u><a href="#">renameTo()</a></u> ( <u><a href="#">File</a></u> dest) Rinomina il file
boolean	<u><a href="#">createNewFile()</a></u> Crea un file vuoto corrispondente all'oggetto File.
boolean	<u><a href="#">setReadOnly()</a></u> Rende il File a sola lettura.



# Esercizi di utilizzo della classe File

---

Creare una directory

Verificare che un file esiste

Stampare le proprietà di un oggetto File

Stampare l'elenco delle directory e dei file



# Un po' di grafica: FileDialog

È un oggetto che permette all'utente di selezionare visualmente un File.

Può esser utilizzato per:

- Selezionare un file da aprire
- Selezionare una cartella in cui salvare un file

Consente di navigare il filesystem in maniera visuale.

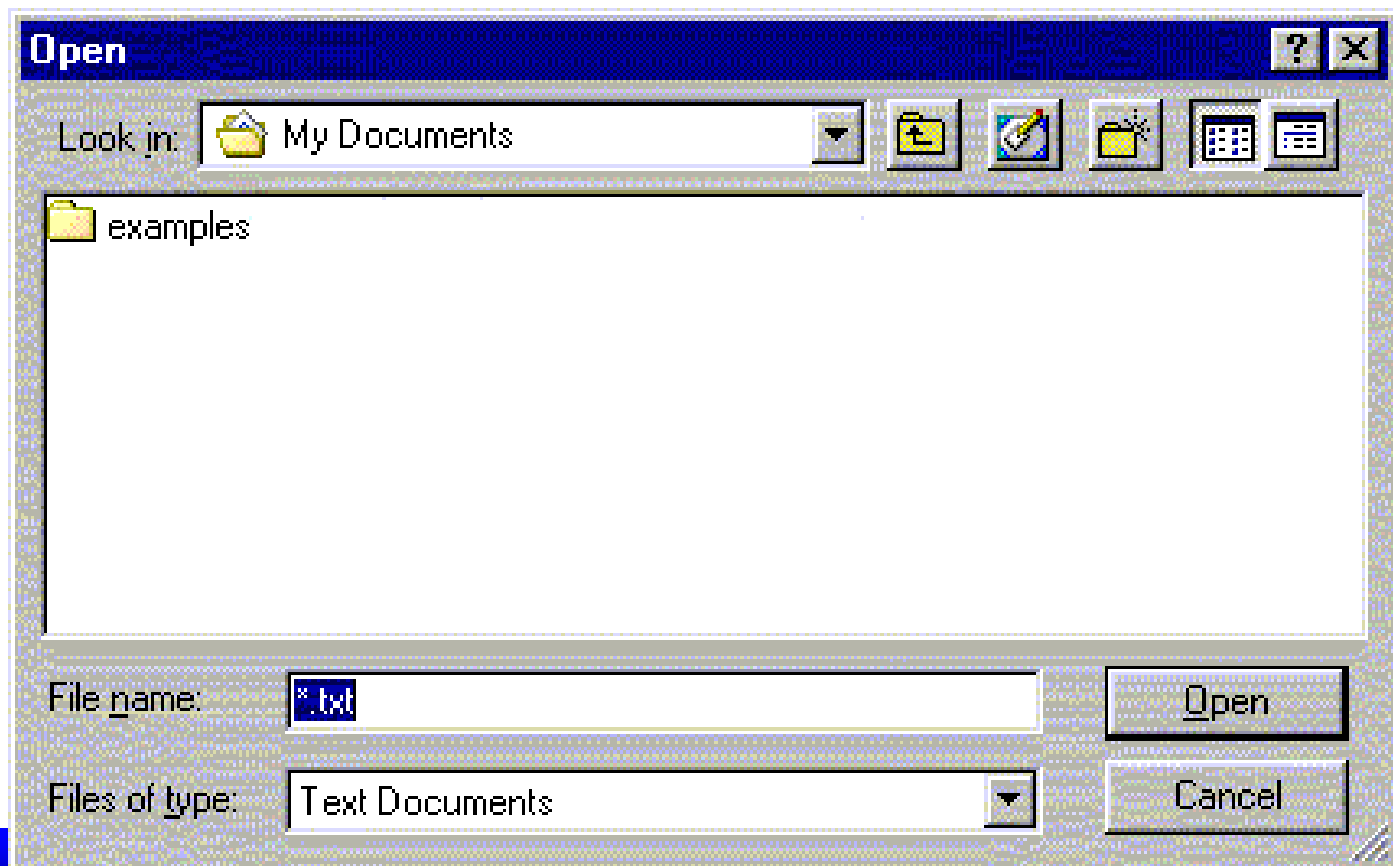




# About FileDialogs

La classe FileDialog visualizza una finestra dalla quale l'utente può selezionare un file.

La finestra FileDialog blocca l'applicazione fino alla sua chiusura



# Costruttori di FileDialog

## FileDialog(Frame f)

- Crea una FileDialog collegata ad una Frame f

## FileDialog(Frame f, String title)

- Crea una FileDialog collegata ad una Frame f con il titolo title

## FileDialog(Frame f, String title, int type)

- Crea una FileDialog collegata ad una Frame f con il titolo title;
- Il tipo di finestra può essere: **FileDialog.LOAD** o **FileDialog.SAVE**



# Metodi utili di FileDialog

String getDirectory()

- Ritorna la directory selezionata

String getFile()

- Ritorna il nome del file selezionato oppure null se nessun file è stato selezionato

int getMode()

- Ritorna FileDialog.LOAD o FileDialog.SAVE, a seconda del Tipo di finestra utilizzata



# Metodi utili di FileDialog

`void setDirectory(String directory)`

- Cambia la directory corrente

`void setFile(String fileName)`

- Cambia il file corrente

`void setMode(int mode)`

- Cambia il tipo di finestra (FileDialog.SAVE o FileDialog.LOAD)

