

UNITAT 11

FITXERS

EXEMPLES

PROGRAMACIÓ
CFGs DAW

Autors:

Joan Vicent Cassany – jv.cassanycoscolla@edu.gva.es

Revisat per:

2022/2023

Llicència



[CC BY-NC-SA 3.0 ES](https://creativecommons.org/licenses/by-nc-sa/3.0/es/) Reconeixement – No Comercial – Compartir Igual (by-nc-sa) No

es permet un ús comercial de l'obra original ni de les possibles obres derivades, la distribució de les quals s'ha de fer amb una llicència igual a la que regula l'obra original. ~~NOTA:~~

~~Aquesta és una obra derivada de l'obra original realitzada per Carlos Cacho i Raquel Torres.~~

Exemple 01

Obtindre informació sobre la ruta.

PROGRAMA PRINCIPAL

```
package UF11_Exemple01;
import java.io.File;
/**
 * UF11 Exemple 01: Obtindre informació sobre la ruta
 */
public class UF11_Exemple01 {

    public static void main(String[] args) {

        // Dues rutes absolutes
        File carpetaAbs = new File("/home/lionel/fotos");
        File arxiuAbs = new File("/home/lionel/fotos/albania1.jpg");

        // Dues rutes relatives
        File carpetaRel = new File("treballs");
        File fitxerRel = new File("treballs/document.txt");

        // Mostrem les seues rutes
        mostrarRutes(carpetaAbs);
        mostrarRutes(arxiuAbs);
        mostrarRutes(carpetaRel);
        mostrarRutes(fitxerRel);
    }

    public static void mostrarRutes(File f) {
        System.out.println("getParent() : " + f.getParent());
        System.out.println("getName() : " + f.getName());
        System.out.println("getAbsolutePath(): " + f.getAbsolutePath() + "\n");
    }
}
```

Exemple 02

Obtindre informació sobre l'estat.

PROGRAMA PRINCIPAL

```
package UF11_Exemple02;
import java.io.File;
/**
 * UF11 Exemple 02: Obtindre informació sobre l'estat
 */
public class UF11_Exemple02 {

    public static void main(String[] args) {

        String rutabase="src/main/java/UF11_Exemple02";

        File temp = new File(rutabase + "/Temp");
        File fotos = new File(rutabase + "/Temp/Fotos");
        File document = new File(rutabase + "/Temp/Document.txt");

        System.out.println(temp.getAbsolutePath() + " existeix? " + temp.exists());
        mostrarEstat(fotos);
        mostrarEstat(document);
    }

    public static void mostrarEstat(File f) {
        System.out.println(f.getAbsolutePath() + " arxiu? "+f.isFile());
        System.out.println(f.getAbsolutePath() + " carpeta? "+f.isDirectory());
    }
}
```

Exemple 03

Obtindre informació sobre l'estat.

PROGRAMA PRINCIPAL

```
package UF11_Exemple03;
import java.io.File;
import java.util.Date;
/**
 * UF11 Exemple 03: Obtindre informació sobre les propietats de fitxers
 */
public class UF11_Exemple03 {

    public static void main(String[] args) {

        File document = new File("src/main/java/UF11_Exemple03/Document.txt");
        System.out.println(document.getAbsolutePath());
        long milisegons = document.lastModified();
        Date data = new Date(milisegons);

        System.out.println("Última modificació (ms) : " + milisegons);
        System.out.println("Última modificació (data): " + data);
        System.out.println("Grandària de l'arxiu: " + document.length());
    }
}
```

Exemple 04

Creació i eliminació de carpetes i arxius.

PROGRAMA PRINCIPAL

```
package UF11_Exemple04;
import java.io.File;
import java.io.IOException;
/**
 * UF11 Exemple 04: Creació i eliminació de carpetes i arxius
 * - En la primera execució crea carpeta i dins arxiu.
 * - En la segona esborra arxiu (la carpeta no perquè té contingut).
 * - En la tercera esborra carpeta (que ja està buida).
 */
public class UF11_Exemple04 {

    public static void main(String[] args) {

        String rutaBase="src/main/java/UF11_Exemple04";

        File carpeta = new File(rutaBase + "/Carpeta");
        File arxiu = new File(rutaBase + "/Carpeta/Arxiu.txt");

        boolean creaDir = carpeta.mkdir();
        // Crea el directori si no existeix.
        if (creaDir) {
            System.out.println("Creat directori " + carpeta.getName() + " en la ruta " + carpeta.getParent() + "? " +
creaDir);
            // Sols esborra el directori si no conté arxius.
        } else {
            if (!arxiu.exists()) {
                boolean delDir = carpeta.delete();
                System.out.println("Esborrat directori " + carpeta.getName() + " en la ruta " + carpeta.getParent() + "? " +
delDir);
            }
        }

        try {
            boolean creaArx=arxiu.createNewFile();

            if (creaArx) {
                System.out.println("Creat arxiu " + arxiu.getName() + " en la ruta " + arxiu.getParent() + "? " + creaArx);
                // Si no ha creat l'arxiu és perquè ja existia i per tant l'esborra.
            } else {
                boolean delArx = arxiu.delete();
                if (delArx) {
```

```
        System.out.println("Esborrat arxiu " + arxiu.getName() + " en la ruta " + arxiu.getParent() + "? " + delArx);
    }
}
} catch (IOException ioe){
    System.out.println("Error en tractament d'arxiu.");
}
}
}
```

Exemple 05

Creació i eliminació de carpetes i arxius.

PROGRAMA PRINCIPAL

```
package UF11_Exemple05;
import java.io.IOException;
import java.io.File;
/**
 * UF11 Exemple 05: Creació i eliminació de carpetes i arxius
 */
public class UF11_Exemple05 {

    public static void main(String[] args) {

        String rutaBase="src/main/java/UF11_Exemple05";

        File carpeta = new File(rutaBase + "/Carpeta");
        File arxiu = new File(rutaBase + "/Carpeta/Arxiu.txt");

        creaEstructura (carpeta, arxiu);

        File carpetaNova = new File(rutaBase + "/CarpetaNova");
        File arxiuInt = new File(rutaBase + "/CarpetaNova/Arxiu.txt");
        File arxiuNou = new File(rutaBase + "/CarpetaNova/ArxiuNou.txt");

        boolean renCar = carpeta.renameTo(carpetaNova);
        System.out.println("S'ha mogut i canviat de nom la carpeta? " + renCar);
        boolean renArx = arxiuInt.renameTo(arxiuNou);
        System.out.println("S'ha mogut el document? " + renArx);
    }

    public static void creaEstructura (File carpeta, File arxiu) {

        boolean creaDir = carpeta.mkdir();
        // Crea el directori si no existeix.
        if (creaDir) {
            System.out.println("Creat directori " + carpeta.getName() + " en la ruta " + carpeta.getParent() + "? " +
creaDir);
            // Sols esborra el directori si no conté arxius.
        } else {
            if (!arxiu.exists()) {
                boolean delDir = carpeta.delete();
                System.out.println("Esborrat directori " + carpeta.getName() + " en la ruta " + carpeta.getParent() + "? " +
delDir);
            }
        }
    }
}
```

```
    }

    try {
        boolean creaArx=arxiu.createNewFile();

        if (creaArx) {
            System.out.println("Creat arxiu " + arxiu.getName() + " en la ruta " + arxiu.getParent() + "? " + creaArx);
            // Si no ha creat l'arxiu és perquè ja existia i per tant l'esborra.
        } else {
            boolean delArx = arxiu.delete();
            if (delArx) {
                System.out.println("Esborrat arxiu " + arxiu.getName() + " en la ruta " + carpeta.getParent() + "? " +
delArx);
            }
        }
    } catch (IOException ioe){
        System.out.println("Error en tractament d'arxiu.");
    }
}
```


Exemple 06

Llistat de carpetes i arxius continguts en una carpeta.

PROGRAMA PRINCIPAL

```
package UF11_Exemple06;
import java.io.IOException;
import java.io.File;
/**
 * UF11 Exemple 06: Llistat de carpetes i arxius continguts en una carpeta
 * - Si existeixen carpetes i arxius els elimna i mostra missatge de carpeta buida.
 * - Si la carpeta està buida crea carpetes i arxius i ho llista.
 */
public class UF11_Exemple06 {

    public static void main(String[] args) {

        // Creem una carpeta Contingut on es posarà la resta de carpetes i arxius
        String rutaBase="src/main/java/UF11_Exemple06";

        File carpeta = new File(rutaBase + "/Contingut");
        boolean creaDir = carpeta.mkdir();
        rutaBase+="Contingut";

        // Omplim o buidem la carpeta Contingut
        creaEstructura (rutaBase);

        // Creem un Array amb els elements que hi ha en la carpeta Contingut
        File dir = new File(rutaBase);
        File[] llista = dir.listFiles();

        System.out.println("Contingut de " + dir.getAbsolutePath() + " :");
        // Si la llista està buida mostrem missatge
        if (llista.length==0) {
            System.out.println("La carpeta està buida.");
        } // Si hi ha elements en la llista recorrem l'Array i es va mostrant el nom de cada element
        else {
            for (int i = 0; i < llista.length; i++) {
                File f = llista[i];
                if (f.isDirectory()){
                    System.out.println("[DIR] " + f.getName());
                } else {
                    System.out.println("[ARX] " + f.getName());
                }
            }
        }
    }
}
```

```
}

// Si la carpeta Contingut té carpetes i arxius els esborrem, si està buida es creen
public static void creaEstructura (String rutaBase) {

    // Creació o esborrat de 5 arxius
    for (int i=1; i<=5; i++){
        File arxiu = new File(rutaBase + "/Arxiu" + i + ".txt");
        try {
            boolean creaArx=arxiu.createNewFile();
            if (creaArx) {
                System.out.println("Creat arxiu " + arxiu.getName() + " en la ruta " + arxiu.getParent() + "? " +
creaArx);
                // Si no ha creat l'arxiu és perquè ja existia i per tant l'esborra.
            } else {

```

Exemple 07

Llegir valors enters d'un arxiu anomenat "Enters.txt"

Es provocarà diversos errors: arxiu no existeix, arxiu amb dades no numèriques, llegir després d'haver arribat al eof, llegir després de tancar l'arxiu.

PROGRAMA PRINCIPAL

```
package UF11_Exemple07;
import java.io.IOException;
import java.io.File;
import java.io.FileNotFoundException;
import java.util.InputMismatchException;
import java.util.NoSuchElementException;
import java.util.Scanner;
/**
 * UF11 Exemple 07: Llegir valors enters d'un arxiu anomenat "Enters.txt"
 * Es provocarà diversos errors: arxiu no existeix, arxiu amb dades no numèriques,
 * llegir després d'haver arribat al eof, llegir després de tancar l'arxiu.
 */
public class UF11_Exemple07 {

    public static void main (String [] args) {

        int valor;

        try {
            // Intentem obrir el fitxer
            File f = new File("src/main/java/UF11_Exemple07/Enters.txt");
            Scanner lector = new Scanner(f);

            // Bucle de lectura fins arribar al final d'arxiu
            while (lector.hasNext()) {
                valor = lector.nextInt();
                System.out.println("El valor llegit és: " + valor);
            }

            // Provoquem un error de lectura després d'haver arribat al final
            // valor = lector.nextInt();
        }
    }
}
```

```
        // Cal tancar el fitxer!  
        lector.close();  
  
        // Provoquem un error de lectura després d'haver tancat l'arxiu  
        // valor = lector.nextInt();  
  
        // Si l'arxiu no existeix  
    } catch (FileNotFoundException nf) {  
        System.out.println("Error 1: " + nf);  
        nf.printStackTrace();  
    }  
    // Si hi ha caràcters no numèrics  
    } catch (InputMismatchException im) {  
        System.out.println("Error 2: " + im);  
        im.printStackTrace();  
    }  
    } catch (NoSuchElementException eof) {  
        System.out.println("Error 3: " + eof);  
        eof.printStackTrace();  
    }  
    } catch (IllegalStateException fc) {  
        System.out.println("Error 4: " + fc);  
        fc.printStackTrace();  
    }  
    } catch (Exception e) {  
        System.out.println("Error 5: " + e);  
        e.printStackTrace();  
    }  
    }  
}
```

Exemple 08

S'escriuen 20 valors enters , començant per l'1 i cada vegada el doble de l'anterior.

- Si l'arxiu no existeix el crea.
- Si existeix el sobreescriu.

Després obrim de nou l'arxiu i afegim dades, en aquest el creem de forma que no sobreescriga el que ja hi havia.

PROGRAMA PRINCIPAL

```
package UF11_Exemple08;
import java.io.File;
import java.io.FileWriter;
import java.io.IOException;
/**
 * UF11 Exemple 08: S'escriuen 20 valors enters, començant per l'1 i cada vegada el
 * doble de l'anterior. Després obrim de nou l'arxiu i afegim dades, , en aquest el creem
 * de forma que no sobreescriga el que ja hi havia.
 */
public class UF11_Exemple08 {

    public static void main(String[] args) {

        // Escriptura inicial. Si no existeix l'arxiu el crea, si existeix el sobreescriu.
        try {
            File f = new File("src/main/java/UF11_Exemple08/Enters.txt");
            FileWriter escriptor = new FileWriter(f);

            int valor = 1;
            for (int i = 1; i <= 20; i++) {
                escriptor.write("" + valor); // escrivim valor
                escriptor.write(" "); // escrivim espai en blanc
                valor = valor * 2; // calculem pròxim valor
            }
            escriptor.write("\n"); // escrivim nova línia
            escriptor.write(65); // Escribim com bytes
            escriptor.close(); // tanquem el FileWriter
            // escriptor.write("" + valor); //Provocarem una excepció
            System.out.println("Fitxer escrit correctament");
        } catch (IOException e) {
```

```
        System.out.println("Error 1: " + e);
    }

    // Es torna a obrir l'arxiu i s'afeg informació
    try {
        File f = new File("src/main/java/UF11_Exemple07/Enters.txt");
        FileWriter escriptor = new FileWriter(f, true);

        escriptor.write("\n"); // escrivim nova línia
        int valor = 65;
        for (int i = 0; i < 20; i++) {
            escriptor.write( (valor + i) ); // escrivim valor
            escriptor.write(" "); // escrivim espai en blanc
        }
        escriptor.write("\n"); // escrivim nova línia
        escriptor.close(); // tanquem el FileWriter
        System.out.println("Fitxer escrit correctament");
    } catch (IOException e) {
        System.out.println("Error 2: " + e);
    }
}
}
```