
Mòdul 6. Accés a dades

UF1. Persistència en fitxers

Persistència en fitxers

Fitxers i Persistència de dades

- **Què és un fitxer i com s'utilitza per emmagatzemar de manera persistent?**

És una unitat de dades emmagatzemada en un dispositiu de memòria. Aquestes dades poden ser de qualsevol tipus: text, imatges, ... qualsevol cosa que es pugui emmagatzemar en format binari.

Les dades que es desen en fitxers es mantenen després que s'atura un programa o s'atura el sistema. Això permet que siguin accessibles en futures sessions.

Els fitxers són una eina fonamental per a emmagatzemar, gestionar i compartir dades en la majoria de sistemes informàtics i aplicacions.

Fitxers i Persistència de dades

- Diferència entre emmagatzematge temporal i persistència de dades

Temporal: durada limitada, dades només disponibles mentre l'aplicació o sessió està en funcionament, ...

Exemples: variables temporals d'una aplicació, dades de sessió d'un usuari, ...

Persistència de dades: durada prolongada, les dades romanen intactes i estan disponibles fins que s'eliminen o modifiquen, ...

Exemples: arxius de text, bases de dades, imatges, documents, registres d'activitat, configuració d'aplicacions, ...

En resum, la principal diferència entre l'emmagatzematge temporal i la persistència de dades rau en la durada i l'accessibilitat de les dades.

Fitxers i Persistència de dades

- **Importància de la persistència de dades en aplicacions**

La persistència de dades és vital en aplicacions i sistemes per a assegurar-se que les dades es mantinguin segures, accessibles i útils a llarg termini.

- Conservació de les dades més enllà de la durada d'execució de l'aplicació.
 - Configuració i preferències.
 - Gestió d'usuaris i autenticació.
 - Històrics i registres.
 - Anàlisi de dades i informes.
 - Versions i control de canvis.
 - Còpies de seguretat.
 - Utilització de dades per altres aplicacions.
 - ...
-

Fitxers i Persistència de dades

- Tipus de fitxers més comuns

Dependrà de les necessitats específiques. Cada tipus de fitxer té el seu propi format, extensió i ús particular.

La varietat de fitxers utilitzats és molt àmplia i depèn de la tecnologia, el llenguatge de programació i les necessitats específiques del projecte de desenvolupament.

Exemples: fitxers de text, d'imatge, de configuració, executables, de compressió, de còpies de seguretat, de registre, ...

Fitxers i Persistència de dades

- **Què és un format de fitxer i com afecta la forma en què s'emmagatzemen les dades**

El format del fitxer defineix la manera en què s'estructuren les dades. Per tant, com es representen, codifiquen i organitzen les dades dins del fitxer per tal que es puguin llegir, escriure, interpretar i processar.

Un format de fitxer correcte garanteix que les dades es puguin gestionar de manera eficient i que es puguin intercanviar, si cal, amb altres sistemes o aplicacions de manera efectiva.

Operacions bàsiques en fitxers

- Crear fitxers i directoris.
 - Llegir i escriure dades en fitxers.
 - Eliminar fitxers i directoris.
 - Navegar pels directoris per obtenir informació.
 - ...
-

Accés a les dades d'un fitxer

- **Accés seqüencial**

És el mètode més senzill per accedir a un fitxer. Es llegeixen o s'escriuen les dades en un fitxer en el mateix ordre en el que apareixen, des de l'inici fins al final.

Característiques:

- **Lectura/escriptura contínua:** Les dades es processen en ordre seqüencial, una línia o un bloc a la vegada.
 - **Útil per a fitxers de text o fitxers que han de ser processats completament.**
 - **Fàcil d'implementar** amb classes de Java.
 - **No permet accedir directament a una posició específica del fitxer.** Si es vol accedir a una part específica, cal llegir tot el fitxer fins a la posició desitjada.
-

Accés a les dades d'un fitxer

Avantatges:

- **Senzill i fàcil d'implementar:** No requereix control complex sobre les posicions de les dades dins del fitxer.
- **Adequat per a processos lineals:** És ideal per a aplicacions que processen dades de principi a fi.

Inconvenients:

- **No és eficient per a cerques o modificacions puntuals:** Si es necessita accedir o modificar una part específica del fitxer, cal llegir tot el contingut abans.
 - **Temps de lectura més llarg en fitxers grans:** Com que el procés és lineal, els fitxers grans poden requerir més temps de lectura i processament.
-

Accés a les dades d'un fitxer

- **Accés aleatori**

Permet llegir o escriure en qualsevol posició dins d'un fitxer, sense necessitat d'haver de processar-lo tot en ordre seqüencial. Ja que ens situem directament a la posició específica dins el fitxer.

Característiques:

- **Posicionament aleatori:** És possible saltar a qualsevol part del fitxer i llegir o escriure dades sense seguir l'ordre seqüencial.
 - **Més flexible:** Permet la modificació de dades a qualsevol part del fitxer sense necessitat de llegir tot el contingut.
-

Accés a les dades d'un fitxer

- **Accés aleatori**

Avantatges:

- **Accés directe:** Molt més eficient per a operacions que requereixen accedir a parts específiques del fitxer.
- **Adequat per a bases de dades o fitxers amb estructures fixes** on es necessita llegir o modificar blocs de dades en posicions específiques.

Inconvenients:

- **Més complex de gestionar:** Requereix controlar explícitament les posicions dins del fitxer.
 - **No sempre és necessari:** Si les dades sempre s'han de processar en ordre seqüencial, aquest tipus d'accés pot ser excessiu i complicar la implementació.
-