

檔案管理：紀錄檔案的存取介面



✿ 檔案寫入介面

◆ 檔案：Employee.data

宣告『可寫入』紀錄檔案的語法範例：	功能說明：
<code>import java.io.*;</code>	導入 java.io 套件
<code>String file = "Employee.data"</code>	檔案名稱（範例：Employee.data）
<code>FileWriter fw = new FileWriter(file);</code>	產生可供寫入的紀錄檔案
<code>BufferedWriter bw = new BufferedWriter(fw);</code>	產生可供寫入的緩衝器
<code>void bw.write()</code>	將字元、字串或位元寫入檔案。
<code>void bw.flush()</code>	強迫寫入檔案。
<code>void bw.newLine()</code>	於檔案內開啟新行（行分隔）。
<code>void bw.close()</code>	關閉檔案。



檔案管理：紀錄檔案的存取介面



✿ 檔案讀取介面，檔案：**Employee.data**

宣告『可讀取』紀錄檔案的語法範例：	功能說明：
<code>import java.io.*;</code>	導入 java.io 套件
<code>String file = "Employee.data"</code>	檔案名稱（範例：Employee.data）
<code>File fileID = new File(file);</code>	開啟檔案，並回傳檔案識別碼
<code>FileReader fr = new FileReader(fileID);</code>	產生可供讀取的紀錄檔案
<code>BufferedReader br = new BufferedReader(fr);</code>	產生可供讀取的緩衝器
<code>br.read()</code>	緩衝器讀取方法之一



檔案管理：紀錄檔案的存取介面



✿ 檔案方法

檔案物件的方法：	說明：
boolean canRead()	測試檔案是否可讀取。
boolean canWrite()	測試檔案是否可寫入。
boolean exists()	測試檔案是否存在
boolean isFile()	測試是否是一般檔案。

✿ 緩衝器方法

『可讀取』檔案的方法：	說明：
void close()	關閉檔案。
void mark()	於檔案內標示某一位置。
int read()	讀取以字串或位元組格式讀取檔案。
String readLine()	由檔案讀取一行（或一筆）紀錄。
void reset()	重置檔案回標示位置。

