

AULA 12

Objectivos

Esta aula faz uma introdução à utilização de ficheiros para o armazenamento permanente de dados. Não se pretende fazer uma abordagem exaustiva ao assunto, mas apenas introduzir as noções necessárias à manipulação de ficheiros de texto. No final da aula, os alunos deverão ser capazes de:

- *declarar um descritor de acesso a um ficheiro e fazer a distinção entre o descritor do ficheiro num programa Java e o nome do ficheiro no sistema operativo;*
- *abrir um ficheiro de texto nos modos de leitura e escrita, reconhecendo que existem dois modos de escrita: um que permite criar um ficheiro novo e outro que permite acrescentar linhas no final de um ficheiro já existente;*
- *ler linhas a partir de um ficheiro de texto e saber detectar o final do ficheiro;*
- *escrever linhas num ficheiro de texto;*
- *fechar um ficheiro.*

1. Analisar o programa abaixo:

```
import pl.*;
import java.io.*;
class aulas12_lerFicheiro extends PlApp {
    static void main(String [] args) throws IOException {
        String s, nomeFx;
        // ler nome de um ficheiro de texto no sistema operativo
        nomeFx = readLine("Nome ficheiro: ");
        // declarar um descritor para leitura do ficheiro de texto
        BufferedReader f=null;
        try {
            // abrir o ficheiro de texto
            f = new BufferedReader(new FileReader(nomeFx));
        } catch(Exception e) {
            // ocorreu um erro na abertura do ficheiro, logo:
            println("Impossível abrir o ficheiro " + nomeFx);
            println("fim do programa...");
            return;
        }
        s = f.readLine(); // ler a 1ª linha
        while(s != null) { // o readLine devolve null no fim ficheiro
            println(s); // mostrar linha no ecrã
            s = f.readLine(); // ler a linha seguinte
        }
        // fechar o ficheiro
        f.close();
    }
}
```

- a) Criar um ficheiro usando um editor de texto à sua escolha (por exemplo o *Notepad* do *Windows*). Escrever algumas linhas de texto e gravar o documento na pasta de trabalho (onde guarda os programas em Java).
- b) Ver o conteúdo do documento criado na alínea anterior, usando o programa proposto acima.
- c) Ver o conteúdo de outros ficheiros de texto armazenados no *Windows* usando o mesmo programa. Introduzir também nomes inválidos (ficheiros que não existem) para verificar o resultado produzido.

2. Copiar, analisar e compilar o programa apresentado abaixo:

```

import pl.*;
import java.io.*;
class escreverFicheiro extends PlApp {
    static void main(String [] args) throws IOException {
        String s, nomeFx = "novo.txt";
        // declarar descritor para acesso ao ficheiro em modo escrita
        BufferedWriter f = null;
        try {
            // abrir o ficheiro de texto
            f = new BufferedWriter(new FileWriter(nomeFx, false));
        } catch(Exception e) {
            // ocorreu um erro na abertura do ficheiro, logo:
            println("Impossível criar o ficheiro " + nomeFx);
            println("fim do programa...");
            return;
        }
        s = readLine(); // ler uma linha do teclado
        while(s.length() > 0) { // termina quando digitar <enter>
            f.write(s); // escrever a linha no ficheiro
            f.newLine(); // inserir mudança linha no ficheiro
            s = readLine(); // ler outra linha do teclado
        }
        // fechar o ficheiro
        f.close();
        println("O conteúdo foi armazenado no ficheiro " + nomeFx);
    }
}

```

- a) Executar o programa e inserir algumas linhas de texto. Notar que o programa termina quando o utilizador inserir uma linha em branco. As linhas são guardadas num ficheiro de texto com o nome "novo.txt".
 - b) Ver o conteúdo do ficheiro usando o programa do exercício nº 1.
 - c) Repetir as alíneas anteriores e verificar o que aconteceu ao conteúdo inicial do ficheiro (que inserir na alínea a).
 - d) Alterar o valor *false* para *true* na abertura do ficheiro e repetir a alínea anterior.
3. Criar um programa que leia um ficheiro com o formato apresentado abaixo à esquerda com a listagem dos alunos de uma turma e cria um ficheiro novo com o formato apresentado abaixo à direita com as notas que obtiveram em Programação em Java. O programa deverá funcionar da seguinte forma:
- i. Pedir ao utilizador para introduzir via teclado os nomes do ficheiro inicial (à esquerda) e do ficheiro novo (à direita);
 - ii. Mostrar o conteúdo do ficheiro inicial linha a linha, pedindo a cada passo a nota obtida pelo aluno listado na disciplina de Programação em Java;
 - iii. Adicionar a nova informação no ficheiro novo.

```

a9845 João Silva Martins
a10200 Ana Maria Almeida
a15800 José Pedro Alves
...

```

```

a9845 João Silva Martins 16
a10200 Ana Maria Almeida 14
a15800 José Pedro Alves 17
...

```

Nota: o ficheiro à esquerda pode ser criado usando o programa do exercício nº 2.

4. Na terra do Alberto Alexandre (localmente conhecido por Auexande Aubeto), o dialecto local é semelhante ao português com duas excepções:

Não dizem os R's

Trocam os L's por U's

Por exemplo, "lar doce lar" deve ser traduzido para "ua doce ua".

Implementar um tradutor de ficheiros de texto em português para o dialecto do Alberto. O tradutor deve pedir ao utilizador o nome do ficheiro a traduzir e apresentar a tradução no ecrã.

5. Considerar o ficheiro de texto abaixo com o registo de uma sequência de operações de compra e venda de acções.

```
Cliente 0010254: compra 1000 acções WCT
Cliente 0017222: vende 500 acções ABC
Cliente 0012005: compra 2000 acções ABC
...
```

Escrever um programa que leia um ficheiro de texto com o formato exemplificado abaixo e que permita fazer as seguintes operações:

- i. Pedir ao utilizador o nome de uma empresa (ABC, WCT, etc.);
 - ii. Listar no ecrã as operações de compra ou venda de acções relativas a essa empresa;
 - iii. No final, apresentar o valor total das compras e o valor total das vendas associados a essa empresa.
6. Considerar um ficheiro de transacções, exemplificado abaixo, em que o carácter ':' permite separar as colunas do ficheiro. A 1ª coluna indica a data da transacção, as 2ª e 3ª colunas indicam o nº e o nome do vendedor, as 4ª, 5ª e 6ª colunas indicam o código, a designação e o preço unitário do produto vendido, a 7ª coluna indica a quantidade vendida e finalmente as duas últimas colunas indicam o nº e a designação do cliente.

```
2007/04/12:5:João Castro:hw25:HP deskjet 6500:125.00:1:1050:Abreu e irmãos Lda
2007/04/12:1:Ana Teles:cs07:Tinteiro HP nº51:30.00:8:1040: Joaquim Silva
2007/04/13:5: João Castro: cs07:Tinteiro HP nº51:30.00:2:1090:APgest
2007/04/13:6:Sónia Castro:hw05:Sony vayo:1600.00:1:1050:Abreu e Irmãos Lda
...
```

Escrever um programa com o seguinte menu de opções:

1. Listar vendas de um vendedor
2. Listar vendas de um produto
3. Pesquisar vendedor
4. Pesquisar produto
0. Sair

A opção 1 deve pedir ao utilizador o nº de um vendedor e em seguida listar as linhas do ficheiro correspondentes às vendas desse vendedor. No final deve indicar o nº total de vendas realizadas.

A opção 2 deve pedir ao utilizador o código de um produto e em seguida listar as linhas do ficheiro correspondentes a vendas desse produto. No final deve indicar o nº total de vendas realizadas.

A opção 3 deve pedir ao utilizador para indicar um ou mais termos correspondentes ao nome do vendedor e apresentar no ecrã todas as linhas em que o nome do vendedor contém esses termos. Por exemplo, se o utilizador introduzir "Castro" o programa deve mostrar no ecrã as linhas correspondentes aos vendedores João Castro, Sónia Castro, etc.

A opção 4 deve pedir ao utilizador para indicar um ou mais termos associadas à designação de um produto e apresentar no ecrã todas as linhas em que a designação do produto contém esses termos. Por exemplo, se o utilizador introduzir o termo "HP" o programa deve mostrar no ecrã as linhas correspondentes aos produtos HP deskjet 6500, Tinteiro HP nº51, etc.