

1. Desenvolva um programa que toda vez que for executado, peça para o usuário digitar uma mensagem (use JOptionPane). Esta mensagem deve ser armazenada no final de um arquivo texto. Na próxima vez que o programa for executado ele deve mostrar a mensagem anterior e pedir uma nova mensagem que também será armazenada. Note que as mensagens devem ser armazenadas juntamente com a data/hora que ela foi digitada.
2. Converta o arquivo cidades.txt disponível no Aprender em um arquivo html 5 de forma que cada cidade apareça em um parágrafo. Deve ser inserido um título a cada troca de iniciais do nome. Teste no navegador:

```
Adamantina
Adolfo
Aguai
Aguas da Prata
Aguas de Lindoia
Aguas de Santa Barbara
Aguas de Sao Pedro
Agudos
Alambari
Alfredo Marcondes
Altair
Altinopolis
Alto Alegre
```



```
<!DOCTYPE html>
<html lang="pt-br">
  <head>
    <title>Cidades do Estado de São Paulo</title>
  </head>
  <body>
    <h1>A</h1>
    <p> Adamantina </p>
    <p> Adolfo </p>
    <p> Aguai </p>
    <p> Aguas da Prata</p>
    <p> Aguas de Lindoia</p>
    ...
    <h1>B</h1>
    <p>Bady Bassitt </p>
    ...
  </body>
</html>
```

Abra automaticamente o arquivo html gerado com o seguinte código:

```
try
{
    Desktop.getDesktop().open(new File("cidades.html"));
}
catch(Exception e){System.out.println(e);}
```

3. Desenvolva um programa para enumerar um arquivo fonte em Java. O programa deverá ler o nome um arquivo-fonte Java válido e copiá-lo com numeração de linhas para outro arquivo com o mesmo nome, concatenado com a extensão <.enum>.

livro.java

```
import java.io.*;

class livro
{
    public String autor;
    public String titulo;
```

livro.java.enum

```
1 import java.io.*;
2 class livro
3 {
4     public String autor;
5     public String titulo;
```

O código fonte abaixo utiliza a classe **File** para realizar uma busca exaustiva por arquivos com a extensão .java a partir de uma pasta qualquer e suas sub-pastas. A solução implementa recursividade. Entenda este código fonte e faça alterações para que a busca seja feita por **conteúdo de arquivo**, ou seja, o usuário informa uma palavra-chave e uma pasta inicial para busca e o programa lista todos os arquivos que possuem em seu **conteúdo** tal palavra-chave.

```
1  import java.io.*;
2
3  class searchfile
4  {
5      public static void rec(String direc)
6      {
7
8          File dir;
9          File dirfiles[];
10         dir = new File(direc);
11         dirfiles = dir.listFiles();
12         try
13         {
14             if(dirfiles.length > 0)
15             {
16                 for(int i=0; i<dirfiles.length; i++)
17                 {
18                     if(dirfiles[i].isFile())
19                         if(dirfiles[i].getName().endsWith(".java"))
20                             System.out.println(dirfiles[i].getAbsolutePath());
21                     if(dirfiles[i].isDirectory())
22                     {
23                         rec(dirfiles[i].getAbsolutePath());
24                     }
25                 }
26             }
27         } catch (Exception e)
28         { System.out.println("ERRO:" + e.toString()); }
29     }
30     public static void main(String args[])
31     {
32         rec("g:\\");
33     }
34 }
```