

## COMO GRAVAR DADOS NO DISCO

1. A classe de dados precisa implementar a serialização. Isso vai permitir converter os dados em um fluxo que será gravado ou lido no hd.

Basta adicionar **implements java.io.Serializable** na declaração da classe.

Ex.:

```
public class Contato implements java.io.Serializable {  
  
    private String nome, fone;  
  
    Contato (String n, String f){  
        nome = n; fone = f; }  
  
    ...  
}
```

2. O próximo passo é adicionar o código de gravação, que é composto pelos seguintes passos:

a) indicar o nome do arquivo onde os dados serão gravados usando a classe **FileOutputStream**.

b) criar um objeto da classe **ObjectOutputStream** (que é capaz de escrever dados) e informar o arquivo onde será gravado.

c) pedir pra o objeto criado no passo b gravar no disco o objeto desejado.

d) fechar o arquivo com o método **close()** de cada objeto.

```
try {  
a)    FileOutputStream arquivo =  
        new FileOutputStream("dados.dat");  
  
b)    ObjectOutputStream gravador =  
        new ObjectOutputStream(arquivo);  
  
c)    gravador.writeObject(lista);  
  
d)    gravador.close();  
        arquivo.close();  
  
    JOptionPane.showMessageDialog(null,  
        "GRAVADO COM SUCESSO");  
} catch (Exception e) {  
    e.printStackTrace();  
}
```

O objeto que está sendo gravado é chamado **lista**, e neste exemplo é um ArrayList de objetos do tipo Contato.

O arquivo gravado ficará na mesma pasta do projeto. Caso queira mudar o local de armazenamento, indique o caminho completo do arquivo, usando duas contra-barras para separar as pastas.

Ex.: ... `new FileOutputStream("c:\\pasta\\subpasta\\arquivo.dat");`

**Obs1.:** esses comandos precisam ficar dentro de uma estrutura try-catch. Isso permite *tentar (try)* executar e *capturar (catch)* eventuais erros que aconteçam.

**Obs2.:** será preciso adicionar `import java.io.*;` no início do código, na linha acima da declaração `public class NomeDaClasse` para poder usar as classes destacadas.

Ex.:

```
package agendadecontatos;

import java.io.*;
import java.util.ArrayList;
import javax.swing.JOptionPane;

public class TelaCadastro extends javax.swing.JFrame {
```

---

3. Finalmente iremos carregar os dados a partir do arquivo gravado no disco e coloca-los de volta na memória, na mesma variável que indicamos na gravação (lista).

Aqui será preciso:

- a) indicar o arquivo a ser lido, através da classe **FileInputStream**.
- b) criar um objeto que lê a partir do disco, com a classe **ObjectInputStream**, e indicar qual arquivo será lido.
- c) colocar na variável os dados lidos a partir do disco, dizendo qual o tipo de objeto que esses dados irão formar.
- d) fechar o arquivo com o método **close()** de cada objeto.

Ex.:

```
try {  
a)   FileInputStream arquivo =  
       new FileInputStream("dados.dat");  
  
b)   ObjectInputStream leitor =  
       new ObjectInputStream(arquivo);  
  
c)   lista = (ArrayList<Contato>  
             leitor.readObject());  
  
d)   leitor.close();  
       arquivo.close();  
  
       JOptionPane.showMessageDialog(null,  
                                     "ARQUIVO LIDO COM SUCESSO");  
} catch (Exception e) {  
    e.printStackTrace();  
}
```

---

Observe que para gravar usamos classe com *Output* (saída) e para ler, classes com *Input* (entrada).