

# Serialização

ELABORATA  
INFORMÁTICA



# Serialização

- **Serialização** é o nome do processo de **conversão de um objeto para uma sequência de caracteres**. Ao processo **inverso**, dá-se o nome de **desserialização**.
- Para que um **objeto** possa ser **serializado / desserializado**, é **necessário** que a **classe** deste objeto implemente a interface **Serializable**. Esta é uma das poucas interfaces do Java que **não declara nenhum método**. Sendo assim, nenhum método precisará ser implementado nas classes de objetos serializáveis.

```
public class Cliente implements Serializable {  
    ...  
}
```



# Fluxos de dados

- Existem fluxos de dados específicos para lidarmos com serialização e desserialização de objetos.

## ***ObjectInputStream***

- Possibilita a leitura de arquivos, em modo de bytes (objetos serializados)

```
//Constrói um leitor de arquivos com objetos serializados
```

- `ObjectInputStream input = new  
ObjectInputStream(FileInputStream("arquivo.dat"));`

```
//Desserializa um objeto
```

- `Object objeto = input.readObject();`



# Fluxos de dados

- ***ObjectOutputStream***

- Possibilita a gravação de dados em arquivos, em modo de bytes (objetos serializados).

```
//Constrói um gravador de arquivos com objetos serializados
```

- `ObjectOutputStream output = new  
ObjectOutputStream(new FileOutputStream("arquivo.dat"));`

```
//Serializa um objeto
```

- `output.writeObject(objeto);`



# Versão de serialização da classe

- O atributo **serialVersionUID** identifica a versão de serialização de uma classe.
- Objetos serializados com versões diferentes do **serialVersionUID** atual da classe não podem ser desserializados.
- É recomendada a utilização do **serialVersionUID** gerado pelo Eclipse.

```
public class Cliente implements Serializable {  
    private static final long serialVersionUID = 2806421523585360625L;  
    ...  
}
```

