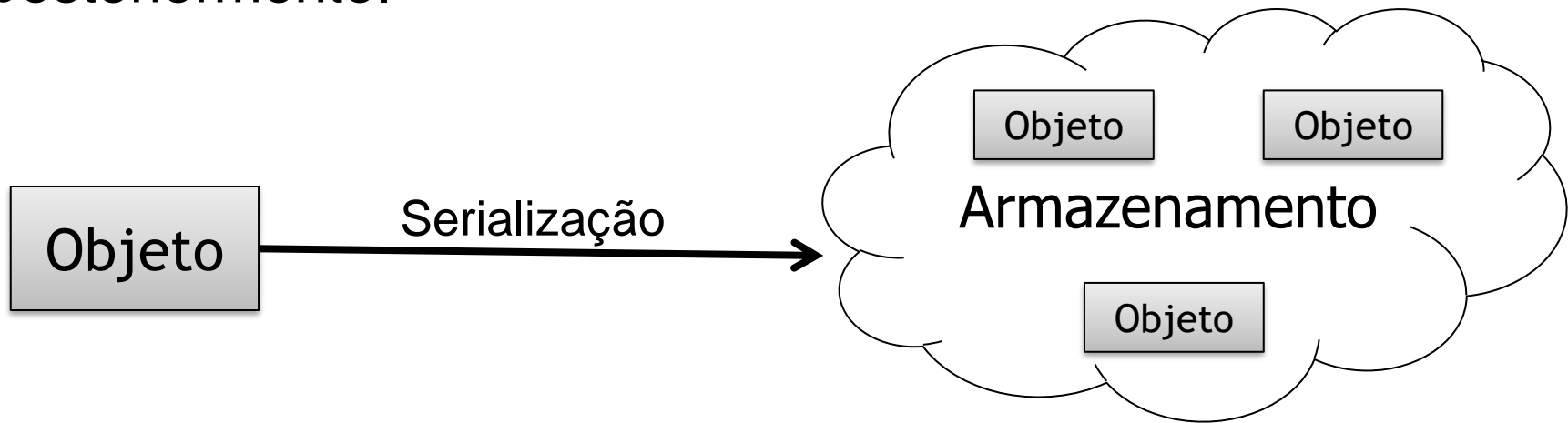


Persistência de Objetos

Persistência

Persistência de dados em Java é utilizada quando se deseja guardar objetos ou estrutura de dados por um tempo indeterminado, sendo possível recuperá-los posteriormente.



Para persistir dados os mesmos precisam ser serializados.

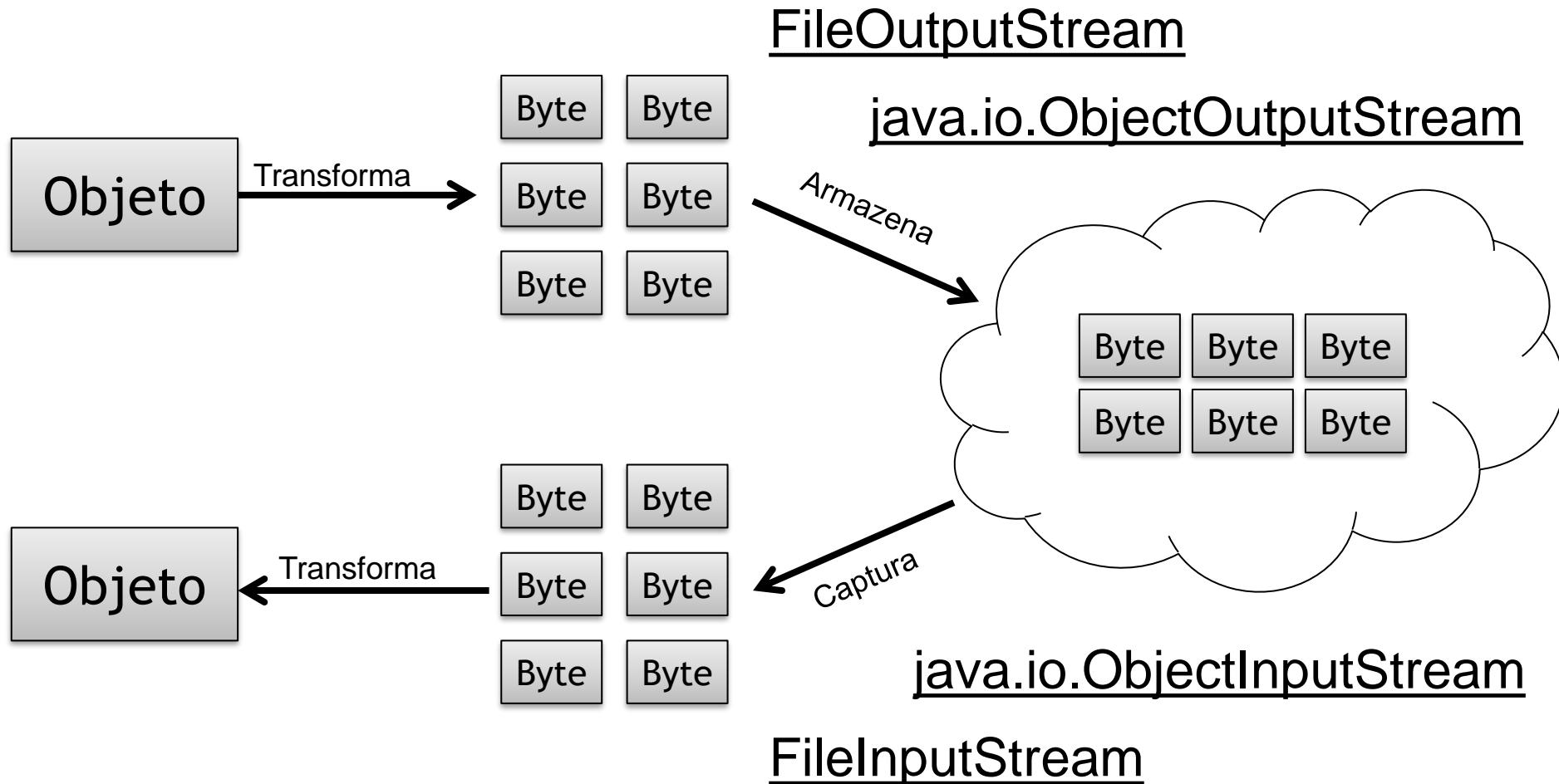
Serialização

- ✓ Captura o estado do objeto ou a estrutura de dados, e transforma em uma cadeia de bytes.
- ✓ Recupera os bytes persistidos para fazer o processo inverso, e ter os dados de volta para a aplicação em execução.
- ✓ Serializar um objeto, em Java, só é possível caso sua classe esteja definida como ***serializável***.

Sintaxe:

```
import java.io.Serializable;  
  
public class NomeDaClasse implements Serializable  
{...}
```

Serialização



Serialização

- ✓ **java.io.FileOutputStream** é um fluxo de arquivo que permite a gravação em disco. Trabalham com arquivos.
- ✓ **java.io.FileInputStream** é justamente o contrário, permitindo a leitura de um arquivo em disco. Trabalham com arquivos
- ✓ **java.io.ObjectOutputStream** semelhante ao **FileOutputStream** e trabalham com objetos.
- ✓ **java.io.ObjectInputStream** semelhante ao **FileInputStream** e trabalham com objetos.

Exercícios

1. Faça uma classe chamada Cliente que implemente Serializable e que contenha os atributos privados cpf, nome e fone. Implemente nessa classe:

- a) Métodos get e set.
- b) Construtor com todos os atributos.
- c) Método toString para retornar o conteúdo dos atributos.
- d) Um método chamado gravar para persistir os dados no disco – o método deve gravar o arquivo cujo nome seja o cpf do cliente.
- e) Um método que recebe o cpf, realiza a leitura do arquivo no disco e apresenta os dados do cliente em tela.

2. Crie uma aplicação para testar as funcionalidades da classe Cliente. A aplicação deve conter um pequeno menu conforme abaixo:

O que você deseja fazer?

1. *Gravar um cliente*
2. *Consultar um cliente*
3. *Sair*

Opção 1: solicita os dados do cliente (use JOptionPane) e armazene os dados em disco.

Opção 2: solicita o cpf e apresenta os dados do cliente em tela. Caso o cliente não exista, a aplicação deve enviar uma mensagem informando ao usuário.

Opção 3: encerra a aplicação.