



Serializar um objeto

Pelo que foi visto no vídeo, vamos analisar em profundidade a que se refere o conceito “serializar”.

Para que um programa Java converta um objeto com todos os seus atributos em um heap de bytes e depois o recupere, o objeto precisa ser serializável. Por ser capaz de converter o objeto em bytes, ele pode, por exemplo, ser enviado pela rede ou salvo em um arquivo e, em seguida, reconstruído do outro lado a partir dos bytes recebidos pela rede ou do arquivo.

Para que um objeto seja serializável, basta implementar a interface `<< Serializable >>`. Como essa interface não possui métodos, é muito fácil implementá-la. Basta digitar `implements Serializable` e pronto. Por exemplo, a classe `Pessoa` é serializável e o Java saberá como enviar um objeto do tipo `Pessoa` pela rede, gravá-lo em um arquivo ou reconstruí-lo a partir de um arquivo.

```
import java.io.*;

public class Pessoa implements Serializable {
    private String nome;
    private String sobrenome;
    private int CPF;
}
```

Se dentro de uma classe que queremos serializar existirem atributos de outras classes, estes, por sua vez, também devem ser serializáveis. A classe `Java ArrayList`, por exemplo, é serializável, portanto, se quisermos salvar uma lista

com todos os seus elementos em um arquivo, esses elementos devem ser objetos de classes que implementam a interface << Serializable >>.

ObjectOutputStream e ObjectInputStream

Para converter um objeto em uma matriz de bytes, precisamos usar a classe ObjectOutputStream e para poder reconstruir um objeto a partir de uma matriz de bytes, a classe ObjectInputStream. A seguir, veremos um exemplo em Java de como salvar o objeto em um arquivo e, em seguida, como recuperá-lo.

```
import java.io.*;
public class Teste{
    public static void main(String[] args) throws IOException {

        /*Salvar o objeto em um arquivo*/
        FileOutputStream fo = new FileOutputStream("OutputFile.txt");
        ObjectOutputStream oos = new ObjectOutputStream(fo);
        oos.writeObject(pessoa);

        /*Recuperar o objeto de um arquivo*/
        FileInputStream fi = new FileInputStream("OutputFile.txt");
        ObjectInputStream ois = new ObjectInputStream(fi);

        System.out.println(ois.readObject());
        ois.close();
    }
}
```

Vamos praticar!

Agora é a sua vez de colocar em prática o que aprendeu. Sugerimos que você resolva o seguinte exercício:

Você deverá manter uma lista de páginas favoritas em um arquivo. As páginas têm uma URL, que é o endereço da página do site na web e um nome.

Sugerimos que você utilize o método main para salvar toda a coleção de páginas em um arquivo e, em seguida, recupere o conteúdo do arquivo em outra coleção para exibir a lista no console.