

Serializando composições

Joana precisa serializar objetos das seguintes classes de seu projeto:

```
public class Livro {  
  
    private Autor autor;  
  
    // restante do código omitido...  
}  
  
public class Autor {  
  
    private String nome;  
  
    // restante do código omitido...  
}
```

E, para isso, escreveu o seguinte código:

```
public class TesteSerializacaoComposicao {  
  
    public static void main(String[] args) throws Exception {  
        Autor autor = new Autor();  
        autor.setNome("Joana");  
  
        Livro livro = new Livro();  
        livro.setTitulo("Serialização com Java IO");  
        livro.setAutor(autor);  
  
        ObjectOutputStream oos = new ObjectOutputStream(new FileOutputStream("livro.bin"));  
        oos.writeObject(livro);  
        oos.close();  
  
        ObjectInputStream ois = new ObjectInputStream(new FileInputStream("livro.bin"));  
        Livro livroSerializado = (Livro) ois.readObject();  
        ois.close();  
        System.out.println(livroSerializado);  
    }  
}
```

Marque as afirmativas corretas sobre o código de nossa colega:

Selecione 2 alternativas

A

Caso alguma das classes não implemente a interface `Serializable`, uma exceção do tipo `NotSerializableException` será lançada!

B Ao serializar um objeto do tipo `Livro`, a operação ocorrerá corretamente e tanto o livro quanto seu autor serão salvos.

C Ao serializar um objeto do tipo `Livro`, para que a operação ocorra sem erros e o autor também seja serializado, é necessário que a classe `Livro` implemente a interface `Serializable`.

D Ao serializar um objeto do tipo `Livro`, para que a operação ocorra sem erros e o autor também seja serializado, é necessário que ambas as classes `Livro` e `Autor` implementem a interface `Serializable`.