

```
import java.io.Serializable;

public class Pessoa implements Serializable {
    private String nome;
    private double pc; // peso corporal
    private double alt; // altura em metros

    public Pessoa(String nome, double pc, double alt) {
        this.nome = nome;
        this.pc = pc;
        this.alt = alt;
    }

    public String getNome() {
        return nome;
    }

    public void setNome(String nome) {
        this.nome = nome;
    }

    public double getPC() {
        return pc;
    }

    public void setPC(float pc) {
        this.pc = pc;
    }

    public double getAlt() {
        return alt;
    }

    public void setAlt(float alt) {
        this.alt = alt;
    }

    public double IMC() {
        return(getPC() / (getAlt() * getAlt()));
    }

    public String interpretaIMC() {
        double vlrIMC = IMC();
        if (vlrIMC < 18.5)
            return("baixo peso");
        else if (vlrIMC < 25.0)
            return("peso adequado");
        else if (vlrIMC < 30.0)
```

```

        return("sobrepeso");
    else return("obesidade");
}
}

```

```

import java.io.File;
import java.io.IOException;
import java.util.ArrayList;
import java.io.FileInputStream;
import java.io.FileOutputStream;
import java.io.ObjectInputStream;
import java.io.ObjectOutputStream;

public class Empacotamento {

    // serialização: gravando o objetos no arquivo binário "nomeArq"
    public static void gravarArquivoBinario(ArrayList<Object> lista, String nomeArq) {
        File arq = new File(nomeArq);
        try {
            arq.delete();
            arq.createNewFile();

            ObjectOutputStream objOutput = new ObjectOutputStream(new
FileOutputStream(arq));
            objOutput.writeObject(lista);
            objOutput.close();

        } catch(IOException erro) {
            System.out.printf("Erro: %s", erro.getMessage());
        }
    }

    // desserialização: recuperando os objetos gravados no arquivo binário "nomeArq"
    public static ArrayList<Object> lerArquivoBinario(String nomeArq) {
        ArrayList<Object> lista = new ArrayList();
        try {
            File arq = new File(nomeArq);
            if (arq.exists()) {
                ObjectInputStream objInput = new ObjectInputStream(new FileInputStream(arq));
                lista = (ArrayList<Object>)objInput.readObject();
                objInput.close();
            }
        }
    }
}

```

```
    }  
    } catch(IOException erro1) {  
        System.out.printf("Erro: %s", erro1.getMessage());  
    } catch(ClassNotFoundException erro2) {  
        System.out.printf("Erro: %s", erro2.getMessage());  
    }  
  
    return(lista);  
}  
  
}
```