ESQUEMA DA AULA PRÁTICA

- **Herança**

1 – Considere o esquema da classe abaixo:

a) Implemente a classe construindo cada um dos seus métodos.

```
public class Base {
  private int a;
  private int b;

  public Base(int a, int b) {
    this.a=a;    this.b=b;
 }
 public int soma(){
   return a+b;
}
 public int soma(int x){
    return a+b+x;
 }
public int getA(){...}
public int getB(){...}
public void setA(int a){...}
public void setB(int b){...}
public String toString {…}
public boolean equals (Object o ){ …}
public Object clone () {…}
}
```

b) Construa uma classe teste para testar a classe Base.

2 - Defina uma classe, Derivada, como subclasse da classe Base. Na subclasse Derivada, defina um atributo **a do tipo int**, um atributo **c do tipo int**, e um construtor com a assinatura **Derivada (Base b, int a, int c)**.

- Na subclasse Derivada construa ainda os seguintes métodos:

a) Getters e setters;

b) Método toString.

c) Método que calcule a soma de todas as variáveis de instância de um objeto da classe Derivada.

d) Método equals.

e) Método clone.

3 - Construa uma classe teste para testar a classe Derivada.

4 - Modifique a classe Base declarando os atributos **a** e **b** com o modificador de acesso ***protected***. Explore o que pode agora mudar no exercício **2**?

5 - Implemente uma classe Disciplina com os atributos codigo (valor do tipo int) designacao (valor do tipo String) e nota (int) e com os seguintes métodos:

a) Construtor com os atributos codigo e designacao como parâmetros;

b) Getters e setters para os atributos da classe Disciplina

c) toString.

d) Construa você mesmo o método equals.

e) Construa o método clone .

6 - Implemente ainda uma classe Aluno tal que cada objecto do tipo Aluno tenha um número e um nome de aluno. Defina um construtor com o número e nome de aluno como parâmetros assim como os getters e setters para cada atributo.

7 – A partir da classe Aluno pretendem definir-se duas subclasses, AlunoLicenciatura e AlunoPosGraduacao.

A classe AlunoLicenciatura terá como atributos o curso frequentado (String) e uma lista de Disciplinas com objectos do tipo Disciplina. Esta lista (objecto do tipo ArrayList<Disciplina>) irá conter as disciplinas feitas pelo aluno.

- Defina o cabeçalho e atributos para a classe AlunoLicenciatura e os seguintes métodos:

a) Um construtor que tem como parâmetros um objecto do tipo Aluno e o curso. b)

Getters e Setters. Use o processo automático apenas para os tipos primitivos.
Para os tipos referenciados (arrays e objectos) deve construir os métodos de forma a não copiar apenas o endereço dos objectos.

c) toString

d) equals

e) clone

f) addDisciplina - dado um objecto do tipo Disciplina deverá adicioná-lo à lista de disciplinas caso este não exista ainda na lista.

f) getNotaDisciplina - dado um código de disciplina deve devolver a nota da disciplina. Se essa disciplina não constar da lista de disciplinas do aluno de licenciatura o método deve devolver o valor 0.

g) método que calcule a média das classificações obtidas pelo aluno de licenciatura.

8 - Numa classe de teste:

a) Construa um método público e estático que dado um array de objectos do tipo Aluno conte quantos desses alunos são alunos de licenciatura.

b) Construa um método público e estático que dada uma ArrayList de objectos do tipo Aluno conte quantos desses alunos são alunos de licenciatura.

b) Teste as operações:

- getNotaDisciplina da classe AlunoLicenciatura;

- o método que calcula a média das classificações do aluno de licenciatura;

- o método que, dado um array de objetos do tipo aluno, conta quantos desses alunos são alunos de licenciatura.

- o método que, dado uma ArrayList de objetos do tipo aluno, conta quantos desses alunos são alunos de licenciatura.

c) Indique qual o output do programa anterior.