

Lista 3

Classes e objetos

1. Baseado na classe **Data**, crie a classe **Hora** que representa uma hora qualquer (usando valores para representar hora, minutos e segundos). Esta classe deverá ter um construtor para iniciar os atributos e sobrescreva o método `toString()` para mostrar seus objetos de forma adequada. Faça um método para validar se a hora é válida antes de inicializar os atributos.
2. Cria uma classe **DataHora** que represente simultaneamente uma data e uma hora. A classe deve ter como atributos instâncias das classes **Data** e **Hora**. O construtor da classe deverá inicializar data e hora, e o método `toString()` deverá ser sobrescrito para mostrar a data seguido da hora.
3. Escreva uma classe **Empregado** que represente um empregado de uma empresa qualquer. Os atributos de um empregado são **nome**, **nível** (dois níveis aceitos), **horasTrabalhadasMes** e **salarioPorHora**. A classe deve ter um construtor que inicialize os atributos, e o método `toString()` deve ser sobrescrito. A classe deve ter os métodos **calculaSalarioMensal**. Este método deve verificar o nível para calcular o salário, os empregados do maior nível tem um acréscimo de 20% sobre o valor normal.
4. Implemente a classe do modelo **Lampada** da questão 3 da Lista 1. Crie o construtor para inicializar o atributo como desligada. O método `toString()` deve mostrar o status da lâmpada.
5. Implemente a classe do modelo **ContaSimplificada** da questão 4 da Lista 1. A classe deve conter dois construtores, um para abrir conta especial e outro para conta simples. A operação de saque deve verificar se o saque é permitido. Contas especiais tem cheque especial de 1 mil reais. O método `toString()` deve mostrar a ficha do conta.
6. Identifique e explique o(s) erro(s) nas classes abaixo:

```
public class DemoImpressao{
    main(String[] args){
        System.out.println("7+2="+ (7+2));
        System.out.println("7-2="+ (7-2));
        System.out.println("7*2="+ (7*2));
        System.out.println("7/2="+ (7/2));
        return true;
    }
}
```

Considere a classe **Data** que foi implementada nas aulas anteriores. Considere que a classe já foi modificada e contém um construtor.

```
public static void main(String[] args){
    Data hoje = new Data();
    hoje.inicializaData(1,7,2013);
    System.out.println(hoje);
}

public class Atribuicoes{
    public static void main(String[] args){
        Data a;
        Data b = new Data();
        b = null;
        b = a;
    }
}
```