

LISTA 2: INTERFACES (AULA 4)

1. Faça primeiramente os exercícios do capítulo 20 da apostila (Kon, F.; Goldman, A.; Silva e Silva, P.; Introdução à Ciência da Computação com Java e Orientação a Objetos, IME-USP, São Paulo, 2005.).
2. O que são interfaces? Em que situações as interfaces são úteis considerando a linguagem Java?
3. Implemente uma Interface denominada SalárioFuncionário, com os seguintes métodos:
 - defineNome(String nome): define nome do funcionário
 - valorHoraBase (double hora): define o valor base da hora trabalhada
 - calculaHE (int quantidade): calcula o valor que o funcionário receberá referente a horas-extras, considerando uma determinada quantidade de horas extras
 - calculaSalarioFinal ():
4. Construa as classes FuncionárioComum e Gerente, que implementam a interface do exercício 2, observando:
 - o valor base da hora (tanto do funcionário quanto do gerente devem ser informados quando os objetos forem instanciados);
 - o valor base da hora do funcionário não pode ser maior que 200. Se for fornecido um valor maior que este o sistema deve emitir uma mensagem de erro;
 - o valor de hora-extra do gerente é calculado considerando 20% sobre o da hora base;
 - o valor de hora-extra do funcionário é calculado considerando 100% sobre o valor da hora base, mas ele pode fazer no máximo 40 horas-extras.
 - o salário final do funcionário é calculado sobre 200 horas mensais.
 - o salário final do gerente é calculado sobre 180 horas mensais mais uma comissão fixa no valor de R\$2000,00.
 - a classe FuncionarioComum deve ter métodos para definir o cargo do funcionário (String) e a data de admissão.
5. Construa uma classe para instanciar 2 funcionários e 3 gerentes usando as especificações dos exercícios 3 e 4. Imprima o valor da hora base, o nome do funcionário, a quantidade de horas-extras, o valor total de horas-extras e o salário final. Para os funcionários comuns, imprima também o cargo e a data de admissão.
6. Construa uma classe que forneça métodos para:
 - Criar e inicializar um array de funcionários com 10 posições.
 - Imprimir listas de funcionários ordenadas por valor da hora-base, nome e data de admissão (use interfaces para evitar repetição de código).
7. Considere as classes Livro e Biblioteca, dadas a seguir. Deseja-se imprimir duas listagens distintas, sendo a primeira em ordem alfabética crescente de título e a segunda em ordem crescente de ano. Utilizando a interface ComparadorDeLivro dada a seguir, faça:
 - Duas classes que implementem a interface fornecida e que sejam usadas para imprimir as listagens solicitadas;
 - Um método para ordenar um vetor de Livros. Este método deve utilizar obrigatoriamente objetos das classes criadas no item anterior, com conceito de interface, para ordenar o vetor de acordo com o atributo solicitado (título ou ano);
 - complete o método main da classe Biblioteca nos locais indicados para imprimir as listagens solicitadas.

```

class Livro
{
    private String titulo;
    private int ano;
    private int tipo; // valor de 1 a 20 que classifica a categoria de cada livro

    void setTitulo (String t) {titulo= t;}
    void setAno (int a) {ano = a;}
    void setTipo (int ti) {tipo = ti;}
    String getTitulo () {return titulo;}
    int getAno () { return ano;}
    int getTipo () { return tipo;}
}

```

```

interface ComparadorDeLivro
{
    int livroMenor(Livro livro1, Livro livro2)
    // retorna: 1 se atributo específico de livro1 é maior que livro2
    //          -1 se atributo específico de livro1 é menor que livro2
    //          0 se atributo específico de livro1 é igual a livro2
}

```

```

class Biblioteca
{
    // método para criar vetor de Livros
    // aqui vai o código para criar o vetor de Livros (você não precisa completar)

    // método imprime
    // aqui vai o código para imprimir o vetor de Livros (você não precisa completar)

    // método para ordenar o vetor
    ... aqui vai código para ordenar o vetor (você deve apresentá-lo completo-)

    // método principal
    public static void main (String [] args)
    {
        Biblioteca bibl= new Biblioteca();

        System.out.println("Livros desordenados");
        bibl.imprime();

        System.out.println("Livros em ordem de título")
        (você deve completar aqui)

        System.out.println("Livros em ordem de ano")
        (você deve completar aqui)

    }
}

```