

ESQUEMA AULA PRÁTICA #3

□ Procedimentos e Arrays

1 - No método main das nossas classes podemos invocar procedimentos (ou métodos) construídos na mesma classe, se esses procedimentos também tiverem o qualificador *static*. Uma variável global à classe também terá de ter o qualificador *static*.

Implemente e execute o exemplo abaixo:

```
public class Teste {  
  
    static int [] lista;  
  
    static int menu() {  
        System.out.println("Opções: ");  
        System.out.println("1 - Ler array ");  
        System.out.println("2 - opção 2 ");  
        System.out.println("3 - opção 3 ");  
        System.out.println("0 - Terminar ");  
        System.out.println("Qual a sua opção? ");  
        int op = Ler.umInt();  
        return op;  
    }  
  
    static void lerArray(int [] listaInteiros) {  
        System.out.println("Introduza os valores do array ");  
        for (int i = 0 ; i < listaInteiros.length; i++ ) {  
            System.out.println("Posição " + i);  
            listaInteiros[i] = Ler.umInt();  
        }  
    }  
  
    public static void main(String[] args) {  
        int escolha, dim;  
        do {  
            escolha = menu();  
            switch (escolha) {  
                case 1:  
                    System.out.println("Qual a dimensão do array? ");  
                    dim = Ler.umInt();  
                    lista = new int [dim];  
                    lerArray(lista);  
                    break;  
                case 2:  
                    System.out.println("Aqui vou invocar o método para a opção 2 ");  
                    break;  
                case 3:  
                    System.out.println("Aqui vou invocar o método para a opção 3 ");  
                    break;  
            }  
        } while (escolha != 0);  
    }  
}
```

Supondo que as opções são:

- 1 - Ler um array de valores do tipo int cuja dimensão é dada pelo utilizador.
- 2 – Escrever os valores de um array na consola;
- 3 – Calcular o produto interno entre dois arrays da mesma dimensão.

- Implemente os procedimentos para as opções 2 e 3 e complete o programa.

- 2 - Adicione uma opção que dado um array de valores inteiros dê como resultado o maior do seus valores.
- 3 - Adicione uma opção que dado um array de valores inteiros, e um valor inteiro x, verifique se x está no array. Se sim, deve devolver true, caso contrário devolver false.
- 4 - Adicione uma opção que dado um array de valores inteiros, devolva um array com os elementos desse array que são múltiplos de 3.
- 5 – Adicione uma opção para ler uma matriz (n x m) de valores inteiros do teclado.
- 6 - Adicione uma opção para escrever uma matriz (n x m) de inteiros na consola.
- 7 – Adicione uma opção que dada uma matrix devolva a sua transposta.

□ A classe `LocalDateTime`

8 - O programa₍₁₎ que se segue, está construído numa classe, `Tempo`. Contém uma função `main`, que constitui o código onde terá início a execução, e nessa função são invocadas as outras funções que estão definidas na mesma classe, `getHoras` e `periodoDia`. A classe contém ainda a declaração de um conjunto de variáveis (`saudacoes`, `nome`, `horas`, `minutos`) que são globais às funções definidas na classe.

A classe `LocalDateTime` existe pré-definida na linguagem Java, e é usada neste programa. Apesar de o programa conter alguns elementos que apenas estudaremos mais para a frente durante este curso, tente explorá-lo, e responder às questões que se seguem:

1) Adaptado de “A small cup of Java” de Pedro Guerreiro

```
import java.time.LocalDateTime;

public class Tempo {

    private static String[] saudacoes =
        {"Bom dia", "Boa tarde", "Boa noite"};

    private static String nome = "Escreva aqui o seu nome";
    private static int horas;
    private static int minutos;

    public static void getHoras( ){
        LocalDateTime dataComputador = LocalDateTime.now();
        horas = dataComputador.getHour();
        minutos = dataComputador.getMinute();
    }
    private static int periodoDia(int h){
        return (h+20) /8 % 3;
    }
    public static void main(String args[]){
        getHoras ( ) ;
        System.out.println(saudacoes[periodoDia(horas)] + ", " + nome);
        System.out.println("Passam " + minutos + " minutos das " + horas + " horas." );
    }
}
```

- Depois de executar e analisar o programa, tente modificá-lo para que:

- i) Em vez de escrever o seu nome no código, ser o programa que pede o nome ao utilizador.
- ii) Apresente as horas de um modo gramaticalmente correto (1 hora, 1 minuto! Passam, faltam! Horas exactas?).
- iii) Refaça o exercício anterior usando o operador ternário (ver TP02).
- iv) Apresente também a data.
- v) Refaça o exercício anterior mostrando o mês por extenso.
- vi) Indique o número de dias que faltam até ao fim-de-semana.
- vii) Apresente a listagem das datas das 13 próximas Sextas-feiras 13.