

Curso Técnico Superior Profissional  
Programação de Sistemas de Informação

## **Serviços e Interoperabilidade de Sistemas**

REVISÕES EM POO (JAVA)
<p><b>Objetivos:</b></p> <ul style="list-style-type: none"><li>- Noção de Objeto/Classe</li><li>- Linguagem JAVA e IDE</li></ul> <p><b>Duração: 1 aula</b></p> <p>©2020-2021: {bruno.madeira, mario.fernandes, nuno.costa, nuno.simoed}@ipleiria.pt</p>

## Introdução

### Programação Orientada a Objetos (POO)

- A POO é baseada na manipulação de objetos e pode conter:
  - Campos com informação, normalmente chamados de Atributos (Ex: Altura, Peso, Idade, etc.);
  - Código na forma de procedimentos, normalmente chamados Métodos (Ex: Crescer, Engordar, Envelhecer, etc.);
  - A estrutura da informação chamada Classe, na qual se criam “cópias” chamadas Instâncias;
  - Classes de um objeto, ou seja, indica o tipo de objeto (Ex: Pessoa);
  - Instâncias de um objeto, ou seja, a realização da classe do objeto (Ex: O João).
- As diferenças para o paradigma procedimental podem ser as seguintes:
  - A programação procedimental permite criar um programa passo a passo que segue uma sequência de instruções;
  - A programação procedimental assenta na ideia de que todos os algoritmos são executados com funções e dados que o programador tem acesso e é capaz de mudar;
  - A programação orientada a objetos é a mais próxima da forma como funciona o mundo real. Pode dizer-se que é comparável ao funcionamento do cérebro humano.

### JAVA (Linguagem e Framework)

- Orientação a objetos
- Sintaxe similar a C/C++/C#
- Portabilidade - write once run anywhere (escreve uma vez corre em todo o lado)
- Inclui várias bibliotecas
- Garbage Collector - Deslocação de memória (Não é necessário o utilizador estar preocupado em desalocar memória alocada por ele)

## Exercícios

1. Abra o IDE NetBeans (Instalar Java SE Runtime Environment e Java SE Development Kit)
2. Crie um novo projecto (File->New Project)
  - a. Escolha “Java with Ant” > “Java Application”
  - b. Defina o nome do projeto como “JavaApplicationFicha1”
  - c. Desmarque a caixa “Create Main Class”
  - d. Sobre “Source Packages-><default package>” carregue com o botão direito e escolha “New->Other”
  - e. Escolha “Swing GUI Forms->Application Sample Form”
  - f. Prima “Next”
  - g. Dê o nome “NewApplicationFicha1”
3. Crie uma nova “Java Class” com o nome “Pessoa” no “default package”
  - a. Sobre “Source Packages-><default package>” carregue com o botão direito e escolha “New->Java Class”
  - b. Defina as propriedades privadas:
    - i. Nome do tipo String (e.g. **Nome**)
    - ii. Data de Nascimento do tipo `LocalDate` faça “**import java.time.LocalDate**” (e.g. **DataDeNascimento**)
    - iii. Morada do tipo String (e.g. **Morada**)
  - c. Crie o construtor com parâmetros de entrada (com as propriedades anteriores)
  - d. Crie um método público para calcular a idade, chamado “**getIdade**”;

```

LocalDate agora;
agora = LocalDate.now() ;
int anos = Period.between(DataDeNascimento, agora).getYears() ;

```

**Nota:** faça “`import java.time.Period`”

- e. Faça o *Override* do método **toString()** para que devolva o nome e idade:  
Ex: João Miguel (21 anos)
4. No *Form* visual “NewApplicationFicha1” adicione os seguintes componentes visuais:
  - a. List (Swing Controls) com o nome: **jListPessoas**
  - b. 3 x `TextField` (para Nome, Morada e Data de Nascimento), com os nomes **jTextFieldNome**, **jTextFieldMorada** e **jTextFieldDataDeNascimento**, respetivamente.
  - c. Botão Adicionar – para adicionar uma pessoa à lista – chamado **jButtonAdicionar**
  - d. Botão Remover – para remover uma pessoa da lista – chamado **jButtonApagar**

- e. O Formulário deve semelhante à imagem abaixo:

- f. Adicione as seguintes propriedades à classe “NewApplicationFicha1”:
- `ArrayList<Pessoa> listaPessoas;`**
  - `DefaultListModel<String> model = new DefaultListModel<>();`**
- g. No construtor defina:
- `listaPessoas = new ArrayList<>();`**
  - `jListPessoas.setModel(model);`**
- h. Implemente os métodos para os botões:
- Adicionar – Para criar uma nova instância do objeto Pessoa com os dados dos campos. Adicionar a instância à lista **`listaPessoas`** e ao **`model`**.
  - Remover – Remover a instância da **`listaPessoas`** e do **`model`**.

**Extra:** para resolver se houver tempo na aula ou para casa:

- Adicionar uma mensagem informativa caso nenhum item esteja selecionado para ser apagado da lista.