

# Aula Prática 2

## Resumo:

- Classes e objectos em Java.

## Exercício 2.1

Analise o código dos ficheiros `Complex.java` e `TestComplex.java`.

- Compile e execute o programa em modo de linha de comando:

```
$ javac TestComplex.java
$ java TestComplex
```

- Altere o programa para que possa aceitar parâmetros na forma “`java TestComplex 5 6`” ou lendo os dois parâmetros através da consola (usando o `Scanner`) e o seguinte formato:

```
Re: 5
Im: 6
```

## Exercício 2.2

A classe `Contacto` representa um contacto telefónico. Já inclui campos privados para guardar o nome e o telefone de um contacto. Acrescente à classe uma forma de guardar um endereço de email e os métodos necessários para que o programa `TestContacto.java` funcione corretamente.

## Exercício 2.3

Crie uma cópia da classe `Contacto` num pacote `pt.ua.prog2` e altere-lhe o método `nome` para devolver o nome em maiúsculas. Copie o programa anterior para `TestContacto2.java` e altere-o de forma a funcionar com a nova classe. Note que o ficheiro do programa não deve pertencer ao mesmo pacote da classe `Contacto`.

## Exercício 2.4

A classe `Data` permite criar objetos que representam datas. Esta classe define três campos

inteiros privados para registrar o dia, o mês e o ano de uma data e já tem um construtor, um método que devolve a data no formato normalizado "AAAA-MM-DD" e um método de classe que permite verificar se um ano é bissexto, mas falta completar vários outros métodos. O ficheiro `TestData1.java` tem um programa que usa objetos de tipo `Data` e testa vários métodos.

Comece por tentar compilar este programa. Terá de completar vários métodos da classe `Data` com instruções `return` que devolvem valores do tipo adequado, ainda que incorretos. Quando conseguir compilar, execute o programa e analise o resultado. A seguir, complete progressivamente os métodos que faltam.

- Acrescente à classe `Data` os métodos `dia`, `mes` e `ano` e descomente algumas linhas do programa para os testar. Esses métodos devem ser `static` ou não? Devem ser declarados `public` ou `private`?
- Complete e teste o método `diasDoMes`, que deve devolver o número de dias de um dado mês (de dado ano). Faça uso do método `bissexto` e da tabela `diasMesComum`, que já estão definidos. Repare que essa tabela é um atributo de classe (`static`): não ocupa memória no contexto de qualquer objeto que venha a ser criado.
- Complete o método `mesExtenso` e teste-o no programa. Sugestão: pode criar uma tabela de nomes e indexá-la com o mês. Essa tabela deve ser um atributo de classe ou de instância?
- Complete o método `extenso` para devolver uma string com a data por extenso, como em "25 de Abril de 1974".
- Complete o método estático `dataValida`, que deve verificar se um terno de inteiros (dia, mês, ano) forma uma data válida, isto é, se o dia e o mês estão dentro dos limites esperados.
- Complete o construtor `Data(dia, mes, ano)`, que inicializa o objecto com essa data.
- Complete o método `seguite` que deve avançar a data para o dia seguinte.

## Exercício 2.5

Usando a classe `Data`, faça um programa `DatasPassadas.java` que escreva, por extenso, todas as datas desde o Natal do ano passado até à data atual.