# Aula Prática 2

**Resumo:**
- Classes e objectos em Java.

**Exercício 2.1**
Analise o código dos ficheiros `Complexo.java` e `p21.java`.

a. Compile e execute o programa em modo de linha de comando:

```
$ javac p21.java
$ java p21
```

b. Altere o programa para que possa aceitar parâmetros na forma "`java p21 5 6`" ou lendo os dois parâmetros através da consola (usando o `Scanner`) e o seguinte formato:

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

**Exercício 2.2**
Crie uma classe para representar um contacto telefónico (`Contacto`). Deve incluir atributos sobre o nome do contacto, o telefone e o email. Ao construir esta classe assegure-se de que o código seguinte (ficheiro `p22.java`) funciona.

```
public class p22 {

   public static void main(String[] args) {
      Contacto[] cl = new Contacto[4];
      cl[0] = new Contacto("Ana", "978676760");
      cl[1] = new Contacto("Rita", "867367834", "rita@gmail.com");
      cl[2] = new Contacto("Paulo", "897476388", "paulo@hotmail.com");
      cl[3] = new Contacto("Carlos", "674767867");
      for (int i = 0; i < cl.length; i++) {
         System.out.println(cl[i].nome() +
```

```
            ": " + cl[i].telefone() +
            "; " + cl[i].eMail());
      }
   }
}
```

**Exercício 2.3**

Coloque a classe `Contacto` num pacote `pt.ua.prog2` e ponha o programa a funcionar (note que o ficheiro do programa principal não deve pertencer ao pacote).

**Exercício 2.4**

Acrescente à classe `Contacto` atributos e métodos (`static`) para contar o número de contactos criados e para validar o nome de cada contacto (o nome não poderá ser vazio).

Exemplo de execução:

```
Nome: Aniceto Zacarias
Telemovel: 900000000
Email: aluno1@ua.pt
Nome: Bernarda Yvone
Telemovel: 900000001
Email: aluno2@ua.pt
Nome: Candido Xavier
Telemovel: 900000002
Email:
Nome: Dalia Wagner
Telemovel: 900000003
Email: aluno3@ua.pt
Listagem:
Aniceto Zacarias: 900000000; aluno1@ua.pt
Bernarda Yvone: 900000001; aluno2@ua.pt
Candido Xavier: 900000002; null
Dalia Wagner: 900000003; aluno3@ua.pt
Contactos: 4
```

Em caso de erro o programa deve ser interrompido com auxílio da função exit() e da mensagem de erro "Contacto inválido!", exemplo:

```
Nome:
Telemovel:
Email:
Contacto inválido!
```

**Exercício 2.5**

Pretende-se construir e experimentar uma classe que represente uma data (dia, mês e ano).

Comece por editar um ficheiro `Data.java` onde se vai implementar essa classe. Implemente (nesta ordem) os seguintes membros da classe:

- três atributos inteiros privados que registem o dia, mês e ano de objectos deste tipo;
- um método que escreva a data registada no objecto no formato: `"DD-MM-AAAA"`;
- um método estático que indique se um ano é bissexto;
- um método estático que determine e devolva o número de dias de um qualquer mês num determinado ano;
- um método estático que indique se um terno de inteiros (dia, mês, ano) formam uma data válida;
- um construtor sem argumentos que inicializa o objecto com a data actual[1];
- um construtor que aceita um terno de inteiros como argumentos (dia, mês e ano) e que inicializa o objecto com essa data;
- um conjunto de métodos que indiquem o dia, mês e ano registados no objecto;
- um método que devolva o nome do mês da data registada no objecto;
- um método que escreva a data registada no objecto por extenso;
- um método `vaiParaAmanha` que modifique a data registada no objecto para o dia seguinte;
- um método `vaiParaOntem` que modifique a data registada no objecto para o dia anterior;

Para testar esta classe, à medida que for construindo os diversos serviços, implemente (num ficheiro separado) um programa tipo menu que permita a realização de cada uma das operações. No fim o programa deverá ter um menu do género:

```
1. Cria novo objecto com a data actual
2. Cria novo objecto com uma qualquer data
3. Indica se a data é válida
4. Escreve data
5. Escreve data por extenso
6. Dia anterior
7. Dia seguinte
0. Termina

NOTA: Se não houver outra indicação, todas as operações
      fazem-se sobre o último objecto criado

Opção:
```

[1] `Calendar hoje = Calendar.getInstance(); int ano = hoje.get(Calendar.YEAR); ...`