

## Lab VII.

### Objetivos

Os objetivos deste trabalho são:

- Identificar e utilizar padrões relacionados com a estrutura de objetos e classes
- Aplicar boas práticas de programação por padrões em casos práticos

*Nota: Para além do código no github, inclua também um ficheiro PDF ou PNG com o diagrama de classes da solução final (pode usar o UMLet ou o draw.io, por exemplo, ou um plugin para o seu IDE).*

### VII.1 Atribuição dinâmica de responsabilidades

A empresa *TodosFazem* (TF) pretende fazer uma gestão dinâmica de funcionários de forma a poder atribuir responsabilidades diversas ao longo do ano.

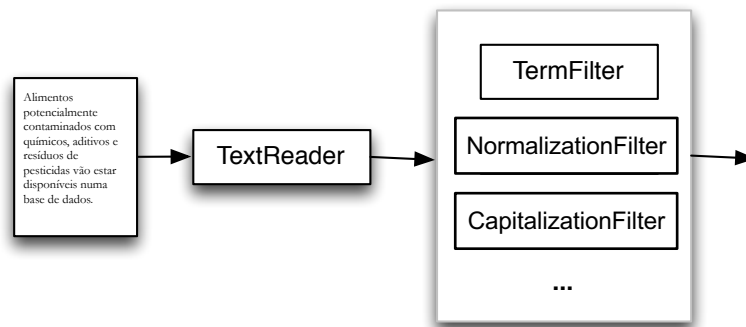
- Usando como base as entidades seguintes, criando outras se necessário, construa uma solução que permita à empresa gerir empregados e poder atribuir dinamicamente, a cada um, competências distintas. Note que um empregado pode, eventualmente, ter ao mesmo tempo várias competências, por exemplo ser *TeamMember* e *TeamLeader*.
- Crie um programa principal para testar a solução. Por simplicidade pode implementar os métodos apenas com mensagens na consola.

```
Employee
    start (Date)
    terminate(Date)
    work()
TeamMember
    start (Date)
    terminate(Date)
    work()
    collaborate()
TeamLeader
    start (Date)
    terminate(Date)
    work()
    plan()
Manager
    start (Date)
    terminate(Date)
    work()
    manage()
```

## VII.2 Processador de texto

Construa uma solução geral que permita ler documentos de qualquer formato (mas na implementação restrinja a ficheiros TXT). O programa deverá permitir ler texto e aplicar um ou mais filtros sobre esse texto.

Tome como base as seguintes entidades, privilegiando a modulação do problema e só depois a implementação de funcionalidades:



- TextReader – lê um ficheiro. Inclui os métodos:
  - boolean hasNext()
  - String next() – devolve parágrafo. Por exemplo:  
*Alimentos potencialmente contaminados com químicos, aditivos e resíduos de pesticidas vão estar disponíveis numa base de dados.*
- TermFilter – Separa em palavras. Inclui os métodos:
  - boolean hasNext()
  - String next() – devolve palavra (por exemplo: *Alimentos*)
- NormalizationFilter – Remove acentuação e pontuação. Inclui os métodos:
  - boolean hasNext()
  - String next() – devolve texto sem acentuação e pontuação (por exemplo: *quimicos*)
- VowelFilter – Remove vogais. Inclui os métodos:
  - boolean hasNext()
  - String next() – devolve texto sem vogais (por exemplo: *lmnts*)
- CapitalizationFilter – coloca em maiúsculas o primeiro e último caracter do texto, e os restantes em minúsculas:
  - boolean hasNext()
  - String next() – devolve texto capitalizado (por exemplo: *AlimentoS*)

Exemplos de utilização:

```
...
reader = new TextReader("someFileName");
reader = new NormalizationFilter(new TextReader("someFileName"));
reader = new VowelFilter(new TermFilter(new TextReader("someFileName")));
...
```

### VII.3 Distribuidor de Cabazes de compras

Construa uma solução que permita criar os seguintes produtos:

- Bebida
- Doce
- Conserva
- Caixa (que pode conter zero ou mais produtos)

Use o programa seguinte para testar a solução:

```
public class Cabazes {  
    public static void main(String[] args) {  
        Caixa principal = new Caixa("Principal", 4);  
        Caixa top = new Caixa("Topo", 2);  
        Caixa bot = new Caixa("Especialidades", 2);  
        top.add(new Bebida("Vinho Reserva UA 2017", 6));  
        top.add(new Bebida("Vinho Reserva UA 2018", 6));  
        principal.add(top);  
        principal.add(bot);  
        bot.add(new Conserva("Atum à Algarvia", 3));  
        bot.add(new Doce("Morango", 2));  
        top.add(new Caixa("Interior", 1));  
        top.add(new Conserva("Sardinhas em Azeite", 5));  
        principal.draw();  
    }  
}
```

*Output possível:*

```
* Caixa 'Principal' [ Weight: 4.0 ; Total: 31.0]  
  * Caixa 'Topo' [ Weight: 2.0 ; Total: 20.0]  
    Bebida 'Vinho Reserva UA 2017' - Weight : 6.0  
    Bebida 'Vinho Reserva UA 2018' - Weight : 6.0  
  * Caixa 'Interior' [ Weight: 1.0 ; Total: 1.0]  
    Conserva 'Sardinhas em Azeite' - Weight : 5.0  
  * Caixa 'Especialidades' [ Weight: 2.0 ; Total: 7.0]  
    Conserva 'Atum à Algarvia' - Weight : 3.0  
    Doce 'Morango' - Weight : 2.0
```