

Notas importantes:

1. Crie um projeto com o seu número mecanográfico, dentro do qual todos os ficheiros deverão ser gravados.
2. Aceda ao elearning. Na página da disciplina, secção "Exame prático", encontrará um link com os ficheiros necessários para o exame.
3. No final do teste, aceda a esta mesma secção e submeta o seu projeto na entrada "Submissão do exame prático" compactando-o num único ficheiro com nome igual ao seu número mecanográfico.

A empresa Domótica pretende construir estações para monitorização ambiental. Cada estação integra um conjunto de sensores que leem parâmetros como temperatura, pressão atmosférica, entre outros. No código fornecido encontra classes que modelam alguns destes elementos, nomeadamente:

- *Domotica.java* – programa principal, para teste do sistema
- *BaseStation* – classe que representa uma estação base. É composta por um conjunto de sensores.
- *Sensor* – interface que todos os sensores devem implementar. *woosvie()*, *colibiotat()*
- *EuroThermo* – sensor de temperatura que respeita a interface *Sensor* e pode ser facilmente ligada à *BaseStation*. *270 (09.04)* *implementa*
- *AmericanThermo*, *ScientificThermo*, *EnglishPressureSensor*, são sensores de outras empresas que terão de ser integradas na estação base, mas que não respeitam a interface necessária para a ligação. A empresa detetou ainda que estes sensores produzem resultados em grandezas diferentes, por exemplo PSI, Fahrenheit e Kelvin, pelo que vai ser necessário proceder também à sua conversão para Bar (PSI) e graus Celsius (F e K).

A seguinte tabela mostra os diversos modelos de sensores presentes no sistema e as suas características. No contexto deste problema não é necessário distinguir o tipo do sensor (temperatura, pressão,...).

Modelo	Tipo	Unidade
<i>AmericanThermo</i>	Temperatura	Fahrenheit
<i>EuroThermo</i>	Temperatura	Celsius
<i>ScientificThermo</i>	Temperatura	Kelvin
<i>EnglishPressureSensor</i>	Pressão	PSI

- a) Visto que a empresa está interessada em uniformizar os sensores e as suas leituras, desenvolva uma solução que permita ligar qualquer um dos sensores à estação sem modificar a implementação original (ou seja, interface *Sensor* e as classes *models.**). Consulte as grandezas pretendidas e as respetivas fórmulas de conversão na tabela abaixo.

Tipo	Grandeza pretendida em <i>BaseStation</i>	Conversões
Temperatura	Celsius	$= K - 273,15$ $= (F - 32) \times (5 / 9)$
Pressão	Bar	$= 15.50368 \times PSI$

De seguida pode observar um exemplo de saída do programa:

```
Base station 'Hall' with 4 active sensors  
EuroThermo: 30.60  
AmericanThermo: 65.56  
EnglishPressure: 317.83  
ScientificThermo: 80.85
```

.... After Calibration

```
EuroThermo: 29.40  
AmericanThermo: 64.36  
EnglishPressure: 316.63  
ScientificThermo: 79.65
```

- b) No programa cliente (*main*) é possível invocar as seguintes operações que quebram o encapsulamento de *BaseStation* (remova os comentários nas duas linhas e verifique o resultado)

```
List<Sensor> sensors = base.getSensors();  
sensors.add(new EuroThermo());  
sensors.remove(0);
```

Construa uma solução que impeça estas operações mas que permita manter o acesso ao conjunto dos sensores da estação.

- c) A empresa está interessada em construir um novo tipo de estação (*BaseStationXL*) e um novo equipamento (*Central*). Cada central terá a responsabilidade de monitorizar uma ou mais estações e a nova estação irá permitir o registo de centrais (tal como indicado na figura) de modo a que sempre que existir uma modificação na estação (ligação ou desativação de sensores) todas as centrais registadas serão imediatamente notificadas. Seguindo o diagrama da figura, se existir uma alteração em *BaseStationXL 2* as duas centrais serão notificadas. Se for ligado um sensor em *BaseStationXL 2* o resultado seria, por exemplo:

```
Central 1: Recent info from base station '2': 1 active sensors  
Central 2: Detectada estação base '2' com 1 sensor ativo
```

Implemente a infraestrutura que permita atingir esta funcionalidade.

