

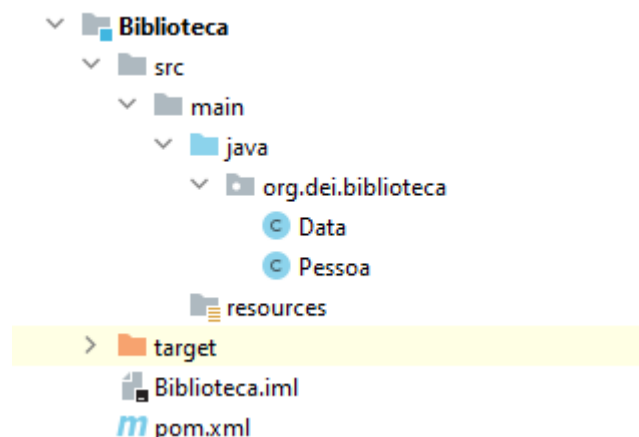
UPskill – Java+.NET

Programação Orientada a Objetos – Agregação e Composição

EXERCÍCIO AUTOMÓVEL

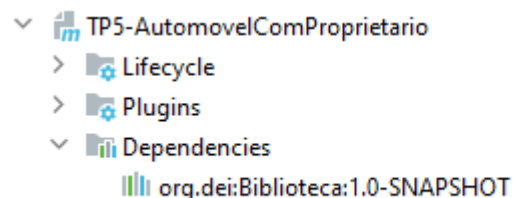
Considere o ficheiro **TP5-Automovel2.zip** disponibilizado que contém os seguintes projetos: **Biblioteca**, **TP5-AutomovelComProprietario** e **TP5-AutomovelComMatricula**.

Verifique que o projeto **Biblioteca** contém as classes **Data** e **Pessoa**, tal como ilustrado na figura seguinte:



Considere agora o **projeto** Maven chamado **TP5-AutomovelComProprietario**:

1. Verifique a utilização da dependência **Biblioteca** neste projeto, tal como ilustrado na figura seguinte (menu View/Tool Windows/Maven).



2. Verifique que no **package** `org.dei.tp5.automovelcomproprietario` existem duas classes: a classe **Automovel** e a classe principal designada **MainAutomovel**.
3. Na classe **MainAutomovel**, verifique a criação e visualização de uma **instância** para representar a **pessoa** chamada **Jorge**, cujo número **ID Civil** do cartão de cidadão é **1234567**.
4. Analise a classe **Automovel** (versão simplificada) para representar também os **proprietários** dos automóveis, através da classe **Pessoa** armazenada na **Biblioteca**. Esboce um **diagrama de classes**, em notação UML, para ilustrar o **relacionamento** entre as classes **Automovel** e **Pessoa**.
5. Na classe **MainAutomovel**, verifique a criação e visualização de uma **instância** que represente o **automóvel** do Jorge, que tem a **marca KIA**.
6. Verifique que o nome do **Jorge** é atualizado para **Jorge Paixão**, na instância de **Pessoa**, e é efetuada uma nova **visualização** das duas instâncias, a **Pessoa** e o **Automovel**. Verifique ainda, que a instância automóvel também foi atualizada.

Considere agora o **projeto** Maven **TP5-AutomovelComMatricula**:

1. Verifique a utilização da dependência Biblioteca neste projeto.
2. Verifique que no **package** `org.dei.tp5.automovelcomproprietario` existem três classes: a classe **Automovel**, a classe **Matricula** e a classe principal designada **MainAutomovel**.
3. Analise a classe **Matricula** (versão simplificada) que permite representar as matrículas dos automóveis. Repare no **construtor de cópia** que permite realizar a **clonagem** de instâncias desta classe. Atualize o **diagrama de classes**, em notação UML, para ilustrar o relacionamento entre **Matricula** e **Data**.
4. Na classe **MainAutomovel** execute as seguintes instruções relacionadas com a classe **Matricula**:
 - a) Crie uma matrícula **11-11-AA** registada em **2015/4/4**. Para tal, comece por criar uma instância de **Data** e depois a instância de **Matricula**.
 - b) Visualize as duas instâncias criadas.
 - c) Altere o valor, na instância **Data**, para **2016/4/4**.
 - d) Faça uma nova visualização das duas instâncias.
5. Corrija o problema detetado e verifique o resultado.
6. Verifique que a classe **Automovel** permite representar também as **matrículas** dos automóveis, usando a classe **Matricula**. Comece por atualizar o diagrama de classes para ilustrar o relacionamento entre as classes **Automovel** e **Matricula**.
7. Na classe **MainAutomovel** execute as seguintes instruções relacionadas com a classe **Automovel**:
 - a) Atualize e visualize o automóvel do **Jorge Paixão** com a matrícula **11-11-AA**, registada em **2015/4/4**. Para isso use as instâncias de **Data** e **Matricula** implementadas no ponto 4 e a instância de **Pessoa** do projeto anterior (**TP5-AutomovelComProprietario**).
 - b) Atualize a instância de **Matricula**, com a matrícula **22-22-BB** e visualize a instância **matrícula** e da instância **automóvel**.
8. Corrija o problema detetado e verifique o resultado.
9. Porque razão é que o problema detetado não se verifica no atributo **marca** da classe **Automovel**?
10. Altere a classe **Automovel** de modo a permitir a clonagem das suas instâncias, através do **construtor de cópia**. Teste na classe **MainAutomovel** esse construtor.