

Exercício 3 – Base de Dados (BD) sobre *Stand*s de Automóveis

Considere a seguinte BD sobre clientes de *stands* de automóveis:

Stand	Nome	Nº Cliente	Idade	Profissão	Compras
Vegas	Rui	2324	23	Médico	Carro Audi A2 por 20000 euros Carro BMW Serie3 por 30000 euros
Vegas	Rita	2325	32	Advogado	Carro Audi A3 por 30000 euros
Vegas	João	2326	26	Professor	Moto Honda GL1800 por 26000 eur.
Vegas	Ana	2327	49	Médico	Carro Audi A4 por 40000 euros Carro BMW Serie3 por 32000 euros Carro Ford Focus por 24000 euros
Miami	Rui	3333	33	Operário	Carro Fiat Panda por 12000 euros
Miami	Paulo	3334	22	Advogado	Carro Audi A4 por 36000 euros
Miami	Pedro	3335	46	Advogado	Carro Honda Accord por 32000 eur. Carro Audi A2 por 20000 euros

1) Registe em Prolog todos os dados relevantes da BD, utilizando factos com a notação:

stand(nome_stand,LC) .

onde **LC** é uma lista de clientes do tipo:

[cliente(nome,num,id,prof,C1),cliente(nome2,num2,id2,prof2,C2),...]

onde **C1, C2** são listas de compras do tipo:

[carro(marca1,modelo1,preco1), moto(marca2,modelo2,preco2),...]

2) Defina em Prolog os seguintes predicados:

- 1) **listar_clientes(X,LC)** – devolve a lista LC com o nome de todos clientes do stand X;
- 2) **listar_dados(X,C,D)** – devolve a lista D com todos dados (i.e.: numero, idade e profissão) do cliente com o nome C do stand X;
- 3) **listar_carros(X,LM)** – devolve a lista LM com o nome de todas as marcas de carros vendidos pelo stand X.
- 4) **listar_advogados(LA)**:- devolve a lista LA com o nome de todos os advogados de todos os stands;
- 5) **preco_medio(X,Med)** - devolve o preço médio (Med) de todos os carros vendidos por um stand. Nota: pode re-utilizar o predicado **media(X,L)** do exercício anterior;
- 6) **altera_id(X,C,Id)** – altera a idade do cliente C do stand X para Id. Nota: deve usar os predicados do Prolog **assert** e **retract**.

Notas: Assuma que existem já definidos os seguintes predicados SWI-prolog:

flatten(L1,L2) – remove todos os [] extra de L1, devolvendo o resultado em L2;

list_to_set_(L1,L2) – remove elementos repetidos de L1, devolvendo L2;

Por exemplo, **flatten([1],[2,3],[1,2,3])** e **list_to_set([1,2,2,3],[1,2,3])** dão verdade.