Trabalho Lógica Matemática - Grupo 6 - Turma ES

Phablo Tavares Paixão - 202300197

Moisés Ferreira Protázio Bastos - 202302624

Pedro Lucas Barbosa Guedes 202305533

Matheus Vieira Mendes Pacheco - 202302623

Osmário santos lima filho - 202302627

Código fonte prolog completo: <https://github.com/phablo-tavares/trabalho-logica.git>

*- Primeiro foi criada uma base de dados com vinte livros para que possam ser usados como exemplos, os livros têm como dados essenciais o nome do mesmo, seu autor, sua categoria e o ano em que foi publicado, além do id que é a sua chave primária.*

% Fatos representando livros

%livro(id,Titulo, Autor, Categoria, Ano).

livro(1,'Ilíada', 'Homero', epopeia, -800).

livro(2,'A Divina Comédia', 'Dante', poesia, 1304).

livro(3,'Hamlet', 'Shakespeare', drama, 1603).

livro(4,'Dom Quixote', 'Cervantes', romance, 1605).

livro(5,'Fausto', 'Goethe', tragédia, 1808).

livro(6,'Orgulho e Preconceito', 'Jane Austen', romance, 1813).

livro(7,'Frankenstein', 'Mary Shelley', ficcao\_cientifica, 1818).

livro(8,'Os Três Mosqueteiros', 'Alexandre Dumas', aventura, 1844).

livro(9,'O Corvo', 'Edgar Allan Poe', poesia, 1845).

livro(10,'Moby Dick', 'Herman Melville', aventura, 1851).

livro(11,'Madame Bovary', 'Gustave Flaubert', romance, 1856).

livro(12,'As Flores do Mal', 'Charles Baudelaire', poesia, 1857).

livro(13,'Os Miseráveis', 'Victor Hugo', romance, 1862).

livro(14,'O Médico e o Monstro', 'Robert Louis Stevenson', horror, 1886).

livro(15,'O Retrato de Dorian Gray', 'Oscar Wilde', ficcao, 1890).

livro(16,'Em Busca do Tempo Perdido', 'Marcel Proust', romance, 1913).

livro(17,'Ulisses', 'James Joyce', romance, 1922).

livro(18,'Mrs. Dalloway', 'Virginia Woolf', romance, 1925).

livro(19,'Admirável Mundo Novo', 'Aldous Huxley', ficcao\_cientifica, 1932).

livro(20,'1984', 'George Orwell', ficcao, 1949).

*- Depois, uma base de dados de clientes fictícios, com cpf’s e nomes aleatórios, sendo o cpf uma chave primária.*

%fatos representando pessoas

%pessoa(cpf,nome).

pessoa(70778358127,'pablo').

pessoa(953 567 841 23,'pedro').

pessoa(96321548769,'Lucas').

pessoa(95348617324,'maria').

pessoa(75986410358,'Julia').

*- E por fim uma base de dados dinâmica representando os empréstimos, onde empréstimos podem ser gerados ou removidos da base.*

% Fatos Definindo um emprestimo

%emprestimo(emprestimoId,livro\_id,pessoa\_cpf,data\_inicio,data\_fim).

:- dynamic emprestimo/5.

emprestimo(1,1,70778358127,'10/10/2023','10/12/2023').

emprestimo(2,10,95356784123,'10/10/2023','10/12/2023').

*- Ao fazer um empréstimo, são executadas regras que buscam o id do livro desejado e o cpf da pessoa, já previamente cadastrada na base de dados, que está fazendo o empréstimo.*

*Após isso, o usuário informa a data de inicio e fim do emprestimo.*

*Por fim, é gerado um Id único para o empréstimo(diferente de todos já existentes), o emprestimo é incluído na base de dados e o id do empréstimo fica disponível na variável EmprestimoID.*

% gera um emprestimo de livro e guarda o id do emprestimo na variavel EmprestimoId

gerarEmprestimo(EmprestimoId) :-

buscarLivro(LivroId),

buscarPessoa(CpfPessoa),

write('Informe a data de retirada do livro: '),

read(DATA\_INICIO),nl,

write('Informe a data de devolução do livro: '),

read(DATA\_FIM),nl,

gerarIdUnicoParaOEmprestimo(IdUnico),

assert(emprestimo(IdUnico,LivroId,CpfPessoa,DATA\_INICIO,DATA\_FIM)),

write('emprestimo de livro gerado com sucesso. Id do emprestimo: '),write(IdUnico),

EmprestimoId is IdUnico.

*- Para encerrar um empréstimo(devolver um livro), é necessário do id do próprio, assim encerrando o empréstimo, e o apagando da base de dados.*

%faz a devolução de um livro a partir do id do emprestimo

devolverLivro :-

repeat,

write('Informe o Id do empréstimo para que seja removido da base de dados: '),

read(EmprestimoId),

(

emprestimo(EmprestimoId,\_,\_,\_,\_) ->

retract(emprestimo(EmprestimoId,\_,\_,\_,\_)), write('Livro devolvido com Sucesso! '), ! ;

write('Não existe Emprestimo com este Id! '),nl, fail

).

*- Aqui foi feito um predicado para buscar o livro pelo seu id.*

buscarLivro(LivroId) :-

repeat,

write('Informe o id do livro: '),

read(ID),

(

livro(ID,\_,\_,\_,\_) ->

LivroId is ID, ! ;

write('Não existe livro com este id!'),nl, fail

).

*- E aqui, uma busca para achar o cliente pelo seu cpf.*

buscarPessoa(CpfPessoa) :-

repeat,

write('Informe o CPF da pessoa: '),

read(CPF),

(

pessoa(CPF,\_) ->

CpfPessoa is CPF, ! ;

write('Não existe Pessoa com este cpf!'),nl, fail

).

*- Gera um id para o empréstimo entre 1 e 10000 que seja único(não existe nenhum emprestimo com o id gerado).*

%id unico de 0 a 10 mil para o empréstimo ()

gerarIdUnicoParaOEmprestimo(IdUnico) :-

repeat,

random(0,10001,RandomNumber),

(

emprestimo(RandomNumber,\_,\_,\_,\_) ->

fail ;

IdUnico is RandomNumber, !

).

*- Busca de todos os livros com o título igual ao valor da variável "Titulo"*

consultarLivro(Titulo):-

livro(\_,Titulo, Autor, Categoria, Ano),

format('Livro: ~w\nAutor: ~w\nCategoria: ~w\nAno: ~w\n\n',[Titulo, Autor, Categoria, Ano]),

fail.

*- Busca de todos os livros com o autor igual ao valor da variável "Autor"*

consutar\_autor(Autor):-

livro(\_,Titulo, Autor, Categoria, Ano),

format('Livro: ~w\nAutor: ~w\nCategoria: ~w\nAno: ~w\n\n',[Titulo, Autor, Categoria, Ano]),

fail.

*- Busca de todos os livros com a categoria igual ao valor da variável "Categoria"*

consultar\_categoria(Categoria):-

livro(\_,Titulo, Autor, Categoria, Ano),

format('Livro: ~w\nAutor: ~w\nCategoria: ~w\nAno: ~w\n\n',[Titulo, Autor, Categoria, Ano]),

fail.

*- Busca dos livros publicados no ano igual ao valor da variavel "Ano"*

consultar\_ano(Ano):-

livro(\_,Titulo, Autor, Categoria, Ano),

format('Livro: ~w\nAutor: ~w\nCategoria: ~w\nAno: ~w\n\n',[Titulo, Autor, Categoria, Ano]),

fail.

*- Outra busca do livro pelo ano, mas para quando não se sabe exatamente o ano em que foi publicado, só a época.*

consulta\_intervalo\_ano(AnoI, AnoF):-

livro(\_,Titulo, Autor, Categoria, Ano),

Ano >= AnoI,

Ano =< AnoF,

format('Livro: ~w\nAutor: ~w\nCategoria: ~w\nAno: ~w\n\n',[Titulo, Autor, Categoria, Ano]),

fail.

*- Exemplificação de consultas.*

￼￼

*- Exemplificação pratica do emprestimo e devolução de livros.*

