



universidade de aveiro
theoria poiesis praxis

Relatório - Bases de Dados

Turma 2 - Diogo Daniel Soares Ferreira e Luís Davide Leira
Universidade de Aveiro - Laboratórios de Informática

25 de Abril de 2015

Conteúdo

I	Apresentação	4
1	Resumo	5
2	Introdução	6
II	Desenvolvimento	8
3	A base de dados	9
4	A aplicação	10
4.0.1	Planeamento da aplicação	10
4.0.2	Funcionamento da aplicação	10
III	Conclusão	15
5	Conclusão	16

Lista de Figuras

2.1	Exemplo do planeamento de uma base de dados relacional. Fonte: Flickr, tec_estromberg	7
4.1	Menu principal funcional em baixo, com o seu código na metade superior da imagem	11
4.2	Localizar livro, com o título contendo a palavra "the", com o seu código na metade superior da imagem	12
4.3	Alterar nome de utilizador, de "Mariana João" para "Diogo Ferreira", com o seu código na metade superior da imagem	13
4.4	Função de apagar utilizador, neste caso "Diogo Ferreira", com parte do seu código na metade superior da imagem	14

Parte I

Apresentação

Capítulo 1

Resumo

Uma base de dados é uma coleção de dados correlacionados entre si com diversos fins. Estes sistemas são de importância tal que nos deparamos e lidamos com eles no nosso quotidiano. É inegável o facto de conferirem organização, consistência e uma forma intuitiva de armazenamento de dados.

Ao longo do relatório é abordada a base de dados criada e sua composição, o planeamento e o funcionamento de uma aplicação desenvolvida que permite a gestão da base de dados, sendo ela uma biblioteca onde se pretende armazenar informação sobre utilizadores, livros e suas requisições. A aplicação apresenta uma interface composta por menus que permitem a realização de operações simples que asseguram o funcionamento da biblioteca em questão.

Sucintamente, a aplicação é capaz de localizar livros (por autor, título ou estado), listar utilizadores e requisições de utilizadores, criar, alterar ou apagar elementos tais como utilizadores, livros e requisições. Esta foi desenvolvida sempre com um método de programação defensivo, assegurando maior estabilidade e flexibilidade. No entanto, revela ser demasiado básica para ser utilizada em contextos mais exigentes a nível profissional, sendo apenas um exemplo de base de dados e de aplicação que assegura o funcionamento correto da mesma. Transmite de uma forma simplista os conceitos base para se poder trabalhar com este tipo de sistemas.

Capítulo 2

Introdução

Bases de dados são sistemas essenciais para o armazenamento de informação. Tratam-se de conjuntos de dados relacionados entre si de maneira simples e eficaz, capaz de organizar, armazenar ou apagar informação em poucas operações. Há vários modelos de bases de dados, sendo os modelos plano, hierárquico e o relacional os mais usados. Neste relatório, serão tratadas apenas as bases de dados relacionais.

Surgidas em meados dos anos setenta, as bases de dados relacionais tratam a informação em tabelas (matematicamente falando, relações), podendo estar relacionadas entre si (Figura 2.1). A linguagem padrão deste tipo de bases de dados é *SQL*, que será usada ao longo deste relatório. Edgar Codd, em 1985, criou o modelo relacional, com treze regras fundamentais para a construção de uma base de dados sólida e com o mínimo de dependências entre tabelas. Não se irá aprofundar o modelo relacional ao longo do relatório, sendo a explicação mais detalhada encontrada no link http://pt.wikipedia.org/wiki/Modelo_relacional.

O objetivo deste relatório é a exposição de uma aplicação codificada em Python 2.x capaz de lidar com uma base de dados simples semelhante à de uma biblioteca. A base de dados contém livros e utilizadores, que poderão requisitar livros. Através de um menu intuitivo ao utilizador, a aplicação terá de localizar, listar, criar, alterar e apagar diversos elementos que compõem a base de dados.

Será utilizado o módulo SQLite3 presente na biblioteca de Python, o qual foi abordado e trabalhado nas aulas de Laboratórios de Informática. Através de uma aplicação simples, é possível desempenhar tarefas que permitem o armazenamento e a manipulação de dados.



Figura 2.1: Exemplo do planeamento de uma base de dados relacional. Fonte: Flickr, tec_estromberg

Parte II

Desenvolvimento

Capítulo 3

A base de dados

Antes de ser desenvolvida qualquer aplicação, é necessário haver uma base de dados para armazenar dados e posteriormente ser manipulada pela aplicação que era pretendido fazer inicialmente. Foi necessário formular as tabelas que inserimos na base de dados, as quais permitem organizar a informação introduzida. Neste caso específico, optou-se por desenhar três tabelas:

- Criou-se uma tabela para os utilizadores chamada de *users* com dois campos: *Id_User*, um inteiro que se irá incrementar automaticamente, a partir do número 1, e que serve de chave primária para cada utilizador; *User_Name*, um texto que irá identificar o utilizador, sendo que não poderão existir dois utilizadores com o mesmo *User_Name*.
- Criou-se outra tabela para os livros chamada de *books* com quatro campos: *Id_Book*, um inteiro que se irá incrementar automaticamente, a partir do número 1, servindo de chave primária para cada livro; *Book_Name*, um texto que será o título do livro; *Author*, um texto que identificará o autor do livro; *Book_State*, que contém o estado atual de requisição do livro (podendo assumir o estado disponível ou requisitado).
- Por fim, criou-se uma tabela específica para as requisições chamada *requisitions*. Apesar de poder ser integrado na tabela dos livros, foi tirada a conclusão que seria mais fácil e lógico organizar a informação da requisição noutra tabela. Logo, essa tabela terá quatro campos: *Id_Book*, que representará o livro requisitado; *Id_User*, que representará o utilizador que requisitou o livro; *Start_date*, que representará a data em que o livro foi requisitado; *End_date*, que será uma data introduzida pelo utilizador para limite de entrega do livro.

Capítulo 4

A aplicação

4.0.1 Planeamento da aplicação

A aplicação começou por ser desenvolvida para correr apenas com os parâmetros de chamada, através da linha de comandos. Após algum trabalho já desenvolvido, concluiu-se que dessa maneira seria impossível descodificar a entrada para a codificação *utf-8*, pois apenas é possível descodificar as entradas caso sejam feitas em tempo de execução, e não antes do programa executar. Isto causaria erros aquando da conversão do texto recebido para base de dados (caso fosse acentuado, por exemplo). Portanto, após um período de discussão, decidiu-se fazer um menu com uma interface hierárquica, que fosse simples e intuitivo para o utilizador. Também foi concordado desenhar a aplicação utilizando técnicas de programação defensiva, parando a sua execução apenas quando o utilizador assim o quisesse. É importante notar que o programa se encontra devidamente tratado contra vários tipos de exceções. Ou seja, se um utilizador introduzir um número fora da gama, ou algo que não seja um número, o programa irá apresentar um erro e dar a oportunidade ao utilizador de voltar a introduzir um valor válido. Desta forma, é pretendido garantir a estabilidade e flexibilidade da aplicação em causa.

4.0.2 Funcionamento da aplicação

Menu principal

O programa inicia com um menu onde o utilizador é sugerido a inserir um número para escolher a operação que deseja realizar sobre a base de dados. Inicialmente, o utilizador tem cinco opções, para além da opção de terminar o programa (Figura 4.1):

1. Localizar um livro;
2. Listar livros, utilizadores ou requisições;
3. Criar utilizadores, livros ou requisições;

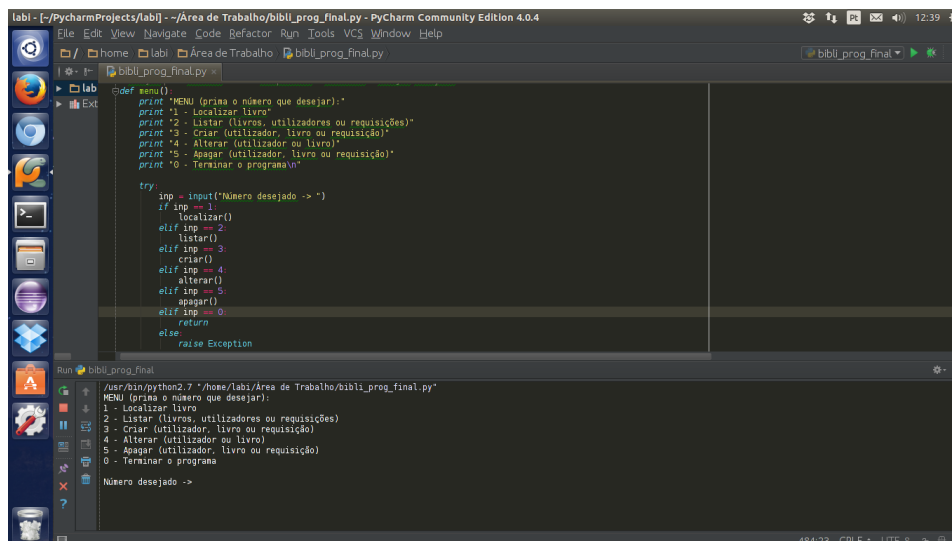


Figura 4.1: Menu principal funcional em baixo, com o seu código na metade superior da imagem

4. Alterar utilizadores ou livros;
5. Apagar utilizadores, livros ou requisições.

Menu "Localizar"

A função localizar apresenta posteriormente um menu, para indicar se se pretende procurar por autor, título ou estado, e em último caso voltar ao menu inicial. Caso se queira pesquisar por autor ou por título, é preciso apenas uma parte da palavra a pesquisar para a encontrar, a uma ou a várias, com o padrão introduzido, devido à instrução "%" antes e depois da palavra introduzida (Figura 4.2). Caso seja escolhida a opção para pesquisar por estado, ainda apresenta mais um menu, que permite que se escolha entre listar os livros disponíveis e os requisitados.

Quando a função acaba, tal como todas as outras funções criadas fora do menu, redireciona o utilizador de volta ao menu principal para optar por outras operações que deseja fazer.

Menu "Listar"

A função listar apresenta um menu com três opções disponíveis que permitem listar livros, utilizadores ou requisições, e em último caso voltar ao menu inicial. Caso o utilizador opte por listar livros (escolha "1"), o programa executa o comando para listar livros na base de dados e depois, um a um, lista-os com uma saída formatada, para ser mais simples ao utilizador ler a informação de cada livro.

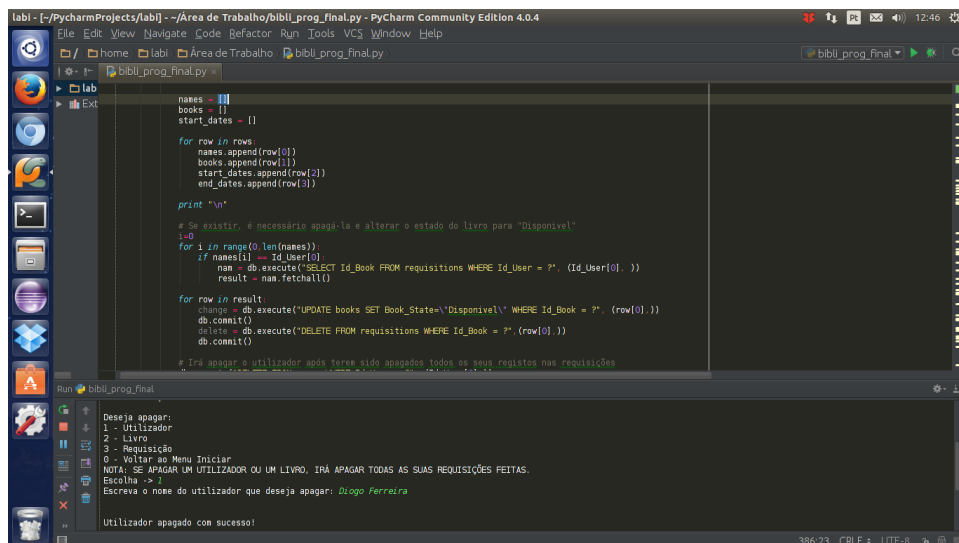


Figura 4.4: Função de apagar utilizador, neste caso "Diogo Ferreira", com parte do seu código na metade superior da imagem

apagar um utilizador da base de dados, a aplicação começa por receber o nome do utilizador. Se ele não existir, lança erro. Caso contrário, procura na base de dados por possíveis requisições desse utilizador. Se existirem, apaga-as e altera o estado dos livros requisitados para disponível. Finalmente, após apagar todas as suas possíveis requisições, o utilizador é apagado da base de dados (Figura 4.4).

Se o utilizador escolher apagar o livro (escolha "2"), o programa começa por receber o título e o autor do livro que deseja apagar. Caso ele não exista, lança um erro. Caso exista, apaga todas as possíveis requisições com esse livro e depois apaga-o da base de dados.

Finalmente, se o utilizador escolher apagar uma requisição (escolha "3"), é pedido ao utilizador para introduzir o título e o autor do livro que deseja alterar o estado. Se o livro não existir, o programa lança um erro. Se existir, altera o seu estado para disponível e , caso exista, apaga as suas requisições.

Parte III

Conclusão

Capítulo 5

Conclusão

As bases de dados são indispensáveis no mundo atual, onde informação é processada e exigida a todo o momento, sendo necessário uma forma de ser armazenada, protegida e de fácil acesso por parte dos utilizadores ou gestores. Elas permitem com que se possa estabelecer relações entre os mais variados dados introduzidos e manipulados. Conferem organização e consistência em formato digital. As aplicações ou conjuntos de programas que garantem que se desempenhem tarefas sobre as bases de dados são bastante importantes devido ao facto do tratamento de dados ser mais prático.

Com esta aplicação é pretendido fornecer um exemplo básico de uma ferramenta essencial para poder agir sobre uma base de dados de uma forma simplista e acessível a qualquer tipo de utilizador, sem quaisquer conhecimentos de programação ou outros conteúdos avançados. Reconhecemos que, apesar de tudo, é uma aplicação demasiado básica, a qual não consegue servir o utilizador em situações de maior detalhe e de manipulação de mais informação. No entanto, o nosso objetivo era o de expor uma aplicação não muito complicada, capaz de ser de fácil compreensão para qualquer utilizador, o que achamos que concluímos com sucesso.