

Projeto de BD

Bases de Dados (CC2005) (<https://moodle2526.up.pt/course/view.php?id=5488>)

Descrição

O projeto tem como tema o desenvolvimento de uma BD, e compreende as seguintes etapas:

1. constituição de grupos e atribuição de tema:

- a. os grupos devem ser formados no Moodle por 3 ou 4 alunos da mesma turma prática registados na plataforma - são permitidos grupos de 1 só aluno em casos especiais como trabalhadores-estudantes ou alunos sem turma prática;
- b. **atribuição de tema:** atribuição de um tema/"dataset" a cada grupo pelo professor das aulas práticas;

2. modelação

- a. **identificação de requisitos e derivação do modelo Entidade-Relacionamento (ER):** descrição dos requisitos do universo em causa para BD e derivação de um modelo ER correspondente;
 - b. **mapeamento do modelo ER em relacional:** o modelo ER deverá ser mapeado num modelo relacional;
3. **validação com o professor das aulas práticas durante o decorrer das aulas:** valide a sua modelação, reformulando-a se necessário em função das orientações dadas pelo professor;

4. implementação

- a. definição de uma BD SQLite correspondente ao modelo relacional, com tabelas bem definidas e povoadas com os dados do "dataset";
- b. elaboração de 10+ interrogações em SQL de complexidade variável, extraindo conteúdo relevante do dataset e ilustrando as técnicas de interrogação estudadas;
- c. implementação de uma aplicação Python em Flask para visualização de dados.

5. entrega e apresentação:

a. entrega de relatório, BD e app Python via Moodle até **12 de Dezembro**

b. apresentação na semana de **15 a 19 de Dezembro**.

O relatório deve ter o esqueleto do documento disponibilizado [aqui \(relatorio.docx\)](#).

A aplicação Python deverá seguir o esquema disponibilizado no arquivo [app.zip \(app.zip\)](#).

Veja as notas abaixo.

Datasets

Os “datasets” são atribuídos pelo professor em cada turma PL, um tema diferente por grupo. Cada arquivo ZIP correspondente contém um arquivo TXT com informação sumária e os dados em formato CSV e Excel.

#	Título
1	Metacritics Best TV Shows and Reviews - 2025 (datasets/1-MetacriticTVShows.zip)
2	Billionaires Statistics Dataset (datasets/2-Billionaires.zip)
3	X-Wines Wines dataset (datasets/3-XWines.zip)
4	Disney+ Movies and TV Shows (datasets/4-DisneyPlus.zip)
5	UNESCO World Heritage Sites (datasets/5-WorldHeritageSites.zip)
6	Nobel Prize winners (datasets/6-NobelPrize.zip)
7	Summer/Winter Olympics (datasets/7-Olympics.zip)
8	Oscar winners (datasets/8-Oscars.zip)
9	Taylor Swift discography (datasets/9-TaylorSwift.zip)
10	Contratos públicos 2024 (datasets/10-ContratosPublicos2024.zip)

Notas sobre a elaboração do trabalho

Modelação

- A modelação deve ser descrita na secção 1 do relatório.
- Para a descrição do universo e modelo ER (secção 1.1):
 - Descreva o universo da BD na forma de requisitos textuais. Estes devem ser enunciados de forma clara, pode usar uma lista de itens por exemplo. Sumarie também a sua concretização em modelo ER via entidades-tipo e relacionamentos.

- Apresente de seguida um diagrama para o modelo ER. Caso facilite a visualização, pode apresentar dois diagramas separados: (1) para as entidades-tipo, e (2) para os relacionamentos omitindo os atributos das entidades-tipo.
- Para o modelo relacional (secção 1.2):
 - Explique o mapeamento do modelo ER para um modelo relacional e apresente um diagrama correspondente. O modelo deverá corresponder a um mapeamento correto do modelo ER, As chaves primárias e externas de cada tabela devem ser explicitadas claramente.
 - O modelo relacional deverá estar na 3ª forma normal, i.e., sem dependências funcionais completas de chave parcial ou transitivas. Se isso não acontecer, será tipicamente devido a um modelo ER mal concebido ou mapeamento para relacional feito de forma incorreta.
- Para o desenho de diagramas, sugere-se o uso das ferramentas dbdia (<https://colab.research.google.com/github/edrdo/dbdia/blob/master/src/main/colab/dbdia.ipynb>) ou draw.io (<https://www.drawio.com/>) .

Definição e povoamento de tabelas

- As tabelas definidas deverão corresponder à proposta de modelo relacional feita, incluindo a definição de chaves primárias e externas. Os campos das tabelas devem também ter tipos de dados apropriados aos valores representados.
- Caso seja útil importar dados de ficheiros CSV via consola sqlite3 ou usando o SQLite Studio, veja este tutorial: “Import a CSV File Into an SQLite Table” (<https://www.sqlitetutorial.net/sqlite-import-csv/>).
- Para a preparação dos dados também poderá ser útil o uso da biblioteca Pandas (https://pandas.pydata.org/pandas-docs/stable/user_guide/index.html).

Implementação da aplicação Python

IMPORTANTE:

1. **O modo de funcionamento da aplicação Python será assunto da aula teórica de 14 de Novembro e também das aulas práticas na semana a seguir. Aguarde por essa altura antes de iniciar trabalho na aplicação Python.**
2. Veja notas de configuração e programação da aplicação no arquivo **README.html** disponibilizado no arquivo **app.zip** ([app.zip](#)).

Requisitos de funcionalidade

1. A aplicação deve ter um “endpoint” para a página de entrada *I*. A página gerada deve permitir acesso às outras funcionalidades.
2. Para cada tabela **T** deverão existir “endpoints”:
 - */T/* que liste todos os registos na tabela **T** e conter links que permitam aceder aos registos individuais ;
 - */T/k/* que liste propriedades de um registo individual em **T** com chave primária **k**;
3. A aplicação deverá conter também “endpoints” correspondentes às 10 interrogações pedidas (ponto 4b);
3. De forma geral, cada página gerada deverá permitir ligação (conter “links HTML”) para acesso a dados de outras tabelas em função dos dados existentes.
4. Deverá empregar instruções SQL parametrizadas para inibir injeção de SQL.