

**Enunciado de trabalho prático**

O trabalho prático da disciplina deve versar sobre o projeto e uso de uma base de dados para um sistema de informação (SI), a ser modelado e implantado em computador. O trabalho envolve a modelagem e o projeto da base de dados com o uso de ferramentas de modelagem, bem como criação, instanciação e manipulação em um modelo relacional. O trabalho está dividido em 3 partes, descritas abaixo. O trabalho deve ser desenvolvido em duplas. Trabalhos individuais só serão admitidos em situações excepcionais, e com acordo prévio.

- **Parte 1: Projeto Conceitual da Base de Dados: 40%**
- **Parte 2: Implementação em Modelo Relacional: 20%**
- **Etapa 3: Manipulação da Base de Dados: 20%**
- **Etapa 4: Implementação: 20%**

**Parte 1) Projeto Conceitual da Base de Dados**

Parte1.a) Escolha alguma realidade para modelar, cujos dados estejam disponíveis na Web. Algumas sugestões:

- Dados abertos (governo, ONGs)
- Empresas/Aplicativos conhecidos (e.g. Spotify, Twitter, Netflix, Uber);
- Coletâneas de dados (Wikipédia, DBLP, Portal de periódicos);
- etc

Parte1.b) Escolha um **aspecto** desta realidade que você vai modelar de acordo com os requisitos quantitativos abaixo.

Descreva em português a parte desta realidade que você selecionou e modelar, e as simplificações que está assumindo. Deve ser possível a partir desta explicação em português compreender o que você modelou. O esquema ER deve contemplar no mínimo:

- a) 14 entidades, todas elas conectadas por pelo menos 1 relacionamento;
- b) 3 relacionamentos com cardinalidade máxima n-m, dos quais pelo menos 1 deve ter atributos;
- c) uma hierarquia relevante de especialização

Parte1.c) O uso de uma ferramenta CASE para modelagem ER (e.g. BR-Modelo, DBMain) é recomendado, e será valorizado.

Parte1.d) Entregáveis:

- Documento PDF: Documento com Descrição em Português do UdD, onde conste claramente a URL do site que serviu como inspiração.
- Documento PDF: Diagrama ER

### Enunciado de trabalho prático

- Documento PDF: Dicionário de dados: descrição do significado de cada entidade/relacionamento/atributo, o domínio dos atributos, todas restrições de integridade não expressas no diagrama, bem como qualquer anotação pertinente.
- Arquivo: Caso use uma ferramenta de projeto, o arquivo da modelagem nesta ferramenta.

#### Parte 1.e) Considerações:

- Será considerada a qualidade da descrição do Universo de Discurso (UdD) e da modelagem realizada. O UdD deve ser completo e coerente, e devem ficar evidentes os requisitos informacionais a serem atendidos pelo SI. Em particular devem ser esclarecidos de forma precisa o **objetivo** do SI (e.g. “gerenciar o acervo de uma biblioteca, o conjunto de usuários, e os empréstimos e reservas de obras”), e as principais **funcionalidades** que devem ser atendidas (e.g. “informar a relação dos usuários, informar por usuário a relação de obras por ele correntemente emprestadas, permitir o cadastramento de novos usuários”, etc) para que possam ser verificada a importância das informações.
- Sugere-se simplificar realidades muito complexas, selecionando alguma parte, e estabelecendo simplificações. Não é necessária uma complexidade acima da solicitada. **Priorize correção** sobre quantidade/complexidade.
- O UdD deverá ser bem descrito em português, ficando claras todas as restrições existentes, e características dos atributos. Deve ser possível verificar a correção de todos os aspectos da modelagem a partir desta descrição, bem como verificar se a base de dados atende os objetivos e funcionalidades especificadas. A descrição do Universo de Discurso, além de ser completa, deve ser concisa, objetiva e sem redundâncias ou conflitos. **Serão desconsiderados enunciados que são meras descrições em português de um esquema desenvolvido, não possuindo as propriedades acima descritas.**
- **A importância de cada entidade/relacionamento descrito na modelagem conceitual deve ficar bem caracterizada na descrição do UdD, através da especificação coerente das funcionalidades e requisitos informacionais<sup>1</sup>.**
- **Serão desconsideradas todas as entidades/ relacionamentos cuja única razão de ser é atingir os requisitos quantitativos acima estabelecidos.** Tal observação é particularmente importante para hierarquias de especializações irrelevantes, e uso de entidades/relacionamentos que poderiam ter sido modelados por atributos opcionais ou repetitivos.

---

<sup>1</sup> Por exemplo, no contexto de uma fábrica de cerveja, se você desejar especificar que cada cerveja tem determinados ingredientes em sua composição, deve esclarecer/justificar este objetivo (por exemplo, patentear as receitas, registrar aos experimentos que levam à criação de produtos, controlar o processo de produção, etc).

Enunciado de trabalho prático
-------------------------------

## Parte 2) Implementação em Modelo Relacional

Parte2.a) Crie um esquema relacional correspondente à sua modelagem. As tabelas devem possuir chave primária, e quando for o caso, chaves alternativas e chaves estrangeiras. Estabeleça as demais restrições de acordo com os recursos de SQL vistos em aula.

Parte2.b) Instancie suas tabelas, baseado em exemplos reais de sua inspiração, ou exemplos fictícios mas factíveis. Cada tabela deve ter no mínimo 10 instâncias, salvo se no UdD isto não fizer sentido.

Parte2.c) Entregáveis:

- Documento PDF: Descrição do mapeamento feito, que descreva o conjunto de regras de transformação aplicadas sobre o esquema ER para derivação do esquema relacional correspondente. Neste conjunto de regras, deve ficar claro como cada entidade, relacionamento, atributo foi transformado em elementos do modelo relacional, e o porquê da estratégia escolhida.
- Arquivo *tabelas.sql*: Um arquivo SQL contendo suas tabelas, que execute no Postgres ou Mysql
- Arquivo *instancias.sql*: Conjunto com comandos de criação de instâncias.

Parte2.d) Observações

- **A descrição textual das tabelas em DDL não é considerada como regras de transformação.**
- Será considerada a qualidade das instâncias preparadas para a população de sua base de dados.

## Parte 3) Manipulação da Base de Dados

Parte3.a) Definir um conjunto de 6 consultas **úteis** ao seu UdD, sendo que cada uma delas deve envolver no mínimo 3 tabelas. Os requisitos quantitativos são:

- a. No mínimo uma delas deve **necessitar** ser respondida com a cláusula group by
- b. No mínimo duas delas deve **necessitar** ser respondida com subconsulta;
- c. No mínimo uma delas (diferente da consulta acima) deve **necessitar** do operador NOT EXISTS para responder questões do tipo TODOS ou NENHUM que <referencia>
- d. Sua base de dados deve estar populada de forma a retornar resultados para todas consultas.
- e. **As consultas devem ser significativamente distintas entre si. Será considerada a utilidade e diversidade das consultas**
- f. Definir um procedimento armazenado que deve ser disparado ao atualizar uma tabela (inserção, atualização ou remoção de tuplas). Você deve pesquisar a linguagem do SGBD escolhido para definir um procedimento, e programar este procedimento nesta linguagem. **Será considerada a utilidade do procedimento proposto. Procedimentos triviais não serão valorizados.**

<b>Enunciado de trabalho prático</b>
--------------------------------------

Parte3.b) Entregáveis:

- Arquivo consultas.sql: Um arquivo SQL contendo suas consultas, com enunciado (na forma de comentário) e comando SQL.
- Arquivo procedimento.sql: um arquivo SQL contendo seu procedimentos, explicando o objetivo do procedimento, e o procedimentos programado.

**Parte 4) Programação**

Parte4.a) Construir um programa que permita fazer manipulações em sua base de dados. Você tem a opção de escolher a linguagem de programação que quiser. Seu programa deve permitir:

- a) conectar com a base de dados;
- b) lançar as consultas SQL definidas na parte 3, mostrando os resultados;
- c) disparar o procedimento armazenado definido na parte 3, mostrando o resultado.

A interface do programa deve permitir as funcionalidades acima. Interfaces com o usuário elaboradas **não são necessárias, e não serão valorizadas** (i.e. não perca seu tempo com isto).

Parte4.b) Entregáveis:

- Documento PDF: Um relatório descrevendo a parte do seu código onde você conecta e acessa a base de dados, ilustrado por porções do código correspondente.
- Código (.zip)

**Datas:**

- 14/11: informar dupla (5% de desconto para quem não respeitar prazo)
- 2/12: entrega do material pelo moodle (desconto de 20% por um dia de atraso, 10% para cada dia adicional de atraso);
- 5, 10, 12/12: questionamentos e demonstração dos programas (por sorteio)