



### Ficha de Problemas

SQL Avançada. Procedimentos, Funções, Triggers, Transações, Cursores e Eventos.  
Segurança e Recuperação de Dados.

#### Um Caso para Estudo

---

Nas fichas anteriores (PL08 e PL09) estivemos a trabalhar com a base de dados “Sakila” (<https://dev.mysql.com/doc/sakila/en/sakila-introduction.html>), desenvolvendo diversos tipos de ações, desde a criação e povoamento da base de dados até à sua exploração com queries de manipulação de dados simples e complexas. Nesta ficha de trabalho vamos continuar o nosso estudo da SQL, desenvolvendo ações de manipulação da base de dados mais avançadas e complexas, abordando a criação de procedimentos (*stored procedures*), funções (*user-defined functions*), gatilhos (*triggers*) e eventos (*events*). Com o conhecimento adquirido com a implementação deste “objetos” da base de dados, estudaremos a definição e aplicação de transações em ações de inserção de dados que envolvam operações de atualização em diferentes tabelas da base de dados. Finalmente, abordaremos a forma como poderemos fazer uma cópia de segurança de uma base de dados e a sua recuperação em caso de falha.

**Parte A** - Utilizando o MySQL Workbench (<https://www.mysql.com/products/workbench/>), pretende-se que desenvolva em SQL as instruções necessárias para realizar as seguintes operações:

- 1) Implemente um procedimento que receba o identificador (“customer\_id”) de um cliente e forneça uma lista com a informação de todos os alugueres (“rental”) feitos por esse cliente, ordenada de forma decrescente por data de aluguer (“rental\_date”).
- 2) Desenvolva uma função que receba o identificador de um filme (“film\_id”) e forneça o preço do seu aluguer. Assuma que esse valor está armazenado no atributo “replacement\_cost” na tabela “film”.
- 3) Modifique os esquemas das tabelas “customer” e “film”, acrescentando a cada uma delas o atributo “NrAlugueres”. De seguida, crie na tabela “rental” um gatilho que atue sempre que seja registado um novo aluguer, incrementando o valor dos atributos agora criados, nas respetivas tabelas.
- 4) Remova o gatilho desenvolvido na alínea anterior. De seguida, desenvolva um procedimento que faça o registo de um novo aluguer na tabela “rental” e, com os dados desse novo registo, incremente o valor do atributo “NrAlugueres” nas tabelas “customer” e “film”, respetivamente. Na implementação deste procedimento deve utilizar transações.
- 5) Criar um evento que permita transferir todos os dias, às 23:00, os dados contidos na tabela “rental” para a tabela “alugueres” (com esquema igual à tabela “rental”) localizada na base de dados “SakilaPt”.
- 6) Implemente um procedimento para atribuição de “pontos de desconto” aos clientes da “Sakila”. Quando executado, o procedimento deve carregar para um cursor o valor dos últimos 50 pagamentos (“payment”) de alugueres, agrupados por identificador de cliente (“customer\_id”), e, para cada um desses clientes calcular os “pontos de desconto” a tribuir – 1 ponto por cada 2,50\$ pago. No final, o procedimento deve apresentar uma lista com o nome dos clientes e os pontos que lhe foram atribuídos – a lista deve ser apresentada ordenada de forma crescente por nome de cliente.

**Parte B** - Utilizando o programa utilitário *mysqldump* do MySQL, pretende-se que desenvolva um processo que permita:

- 1) Fazer uma cópia de segurança da base de dados “Sakila”.
- 2) Recuperar a base de dados “Sakila”, após a ocorrência de uma situação de falha completa – assuma que a base de dados foi acidentalmente removida do sistema.

\* \* \* \* \*