

**Departamento de Ciências de Computação – SCC
Instituto de Ciências Matemáticas e de Computação – ICMC
Universidade de São Paulo – USP**

**Laboratório de Bases de Dados
Docente Responsável: Profa. Dra. Cristina Dutra de Aguiar Ciferri
Estagiária PAE: Jaqueline Joice Brito**

**Exercícios Práticos
Pacotes**

Considere o esquema conceitual e o esquema relacional do projeto “**empresa de aluguel de carros**”.
Faça os seguintes exercícios, usando o script completo de criação e inserção de dados:

Exercício 1

Crie um pacote, chamado `manipulaFunc`, que contenha diversas funções e procedimentos relacionados à manipulação de funcionários, da seguinte forma:

a) Existe um procedimento chamado `contrataFunc`, que tem como objetivo inserir um funcionário contratado de forma adequada no banco de dados. O novo funcionário: (i) deve ser associado à empresa que possui o menor número de funcionários; (ii) deve ter como supervisor o funcionário mais velho; (iii) deve receber como salário a média dos salários dos funcionários; (iii) deve ter o seu valor de chave primária obtido usando sequência. Os demais dados do novo funcionário devem ser passados como parâmetro para o procedimento.

b) Existe um procedimento chamado `demiteFuncMot`, que remove um motorista de forma adequada do banco de dados, de acordo com o número da carteira desse motorista. O número da carteira deve ser passado como parâmetro.

c) Existe uma função chamada `alteraSalFunc`, que altera o salário de um funcionário de acordo com o número de dependentes que ele possui. Por exemplo, se o funcionário possuir 1 dependente, ele terá um aumento de 10%; se o funcionário possuir 2 dependentes, ele terá um aumento de 20%; e assim por diante. Deve ser passado como parâmetro para a função o nome completo do funcionário, e a função deve retornar a porcentagem de aumento do salário do empregado, além de atualizar o salário do funcionário de forma adequada no banco de dados.

d) Existe uma função chamada `alteraSupFunc`, que altera o supervisor de um funcionário, de forma que o novo supervisor ganhe um salário maior do que o salário do funcionário. Deve ser passado como parâmetro para a função o nome completo do funcionário, e a função deve retornar o nome completo do novo supervisor ou então ‘sem supervisor’ caso não exista supervisor que receba salário maior do que o salário do empregado. Neste segundo caso, o código do supervisor do empregado na tabela do banco de dados deve receber o valor NULL.

Crie um programa PL-SQL que chame as funções e procedimentos declarados no pacote `manipulaFunc` e verifique se os resultados obtidos estão corretos.

Exercício 2

Crie um pacote, chamado *Relatorio*, que contenha funcionalidades relacionadas à geração de diversos relatórios.

a) *Relatorio.resumoEmpresas*

Para cada *empresa*, gere um relatório com os seguintes dados:

- Quantidade de filiais
- Quantidade de funcionários totais, de secretários e de motorista.
- Quantidade de clientes
- Quantidade de alugueis
- Quantidade de carros

b) *Relatorio.desempenhoEmpresas*

Para cada *mês* e *empresa*, gere um relatório com os seguintes dados:

- Lucro bruto médio mensal (soma dos valores dos alugueis de carros)
- Lucro líquido mensal (soma dos valores dos alugueis - salário dos funcionários)
- Cliente que mais aluga carros em número de alugueis (quantidade de tuplas na tabela 'aluga')
- Motorista que mais dirige carros (com maior quantidade de tuplas na tabela 'dirige')
- Cliente do mês (que mais alugou em R\$) e a soma dos alugueis.
- Motorista do mês (que mais horas dirigiu) e a quantidade de horas que dirigiu.
- Carro mais utilizado (somando motoristas + clientes) e o número de vezes que foi utilizado.

c) *Relatorio.supervisores*

Para cada *supervisor*, gere um relatório com os seguintes dados:

- Nome do supervisor
- Quantidade total de supervisionados
- Média de dependentes por funcionário supervisionado
- Quantidade de secretários supervisionados
- Quantidade de motoristas supervisionados
- Média salarial dos supervisionados

Crie um programa PL-SQL que chame os procedimentos declarados no pacote *Relatório* e verifique se os resultados obtidos estão corretos.
