Considera as seguintes tabelas, `convenios`, `pacientes` e `atendimentos` que serão utilizadas para responder as questões 2 e 3:

```
convenios (con_codigo, con_nome)
pacientes (pac_codigo, pac_nome, pac_nascimento, con_codigo)
atendimentos (ate_codigo, ate_data, ate_diagnostico, pac_codigo)
```

Os campos marcados em negrito são as **chaves primárias** e as ***estrangeiras*** estão em negrito e sublinhadas.

Exercício 1

Escreva o código SQL necessário para criar as tabelas no banco de dados.

Exercício 2

Construa consultas SQL para listar:

a) O nome e a data de nascimento do paciente mais jovem;

b) O nome do paciente e o nome do convênio do paciente mais idoso;

Exercício 3

Escreva instruções SQL para listar:

A) Os convênios que não possuem nenhum paciente conveniado;

B) Os pacientes que não agendaram atendimentos no mês de janeiro de qualquer ano.

A tabela `empregados` será usada na questão 4 e possui os seguintes campos:

```
empregados (emp_matricula, emp_nome, emp_depto,
emp_funcao,emp_salario, emp_temposervico, emp_inicioferias,
emp_filhos)
```

`emp_matricula` é a chave primária da tabela.

Exercício 4

Construa consultas SQL para listar:

a) O nome do empregado, o departamento no qual ele está alocado e a média salarial desse departamento. A listagem deve ser ordenada pelo nome do departamento e, dentro do departamento, pelo nome do empregado.

b) O nome do empregado, o seu salário e o nome do departamento no qual ele está alocado cujo salário seja inferior ao valor médio aplicado naquele departamento. A listagem deve ser ordenada pelo nome do departamento e, dentro do departamento, pelo nome do empregado.

Exercício 5

Crie uma tabela chamada `tarefas` com os campos: `descricao`, `prazo` e `concluida`. Depois crie três visões, uma para exibir as tarefas atrasadas, outra para exibir as tarefas pendentes (atrasadas ou não) e outra para exibir as tarefas já concluídas.

Roteiro de Atividade

1. Introdução

Esta atividade tem como objetivo praticar a implementação da tecnologia JWT e de tratamento de erros.

Ao desenvolver as atividades práticas de laboratório, atente sempre em implementar código de maneira organizada, formatada e estruturada: o chamado código limpo. Mantenha seu código sempre de acordo com as *guidelines* ditas pelas boas práticas de programação. Peça orientação a seu professor mediador sempre que tiver dúvidas.

Atividade 01

Crie um programa em Node.js que gere tokens JWT que satisfaça os seguintes requisitos:

a) A token deve ser obtida a partir de uma rota /token
b) A secret deve ser obtida a patir da leitura de uma variável de ambiente
c) A token deve possui tempo de expiração de 1 hora
d) A token deve ter um payload com um número aleatório gerado entre 0 e 1

Atividade 02

Escreva um programa que valide uma token JWT e que satisfaça os seguintes requisitos:

a) A validação deve ocorrer na rota /validar
a) A token deve ser enviada no cabeçalho da requisição no campo “Authorization”
b) A requisição deve retornar código 200 para válido e 400 para inválido
c) Em caso de sucesso deve ser retornado o payload do token
d) Deve-se utilizar try/catch para validar o token