|  |  |  |
| --- | --- | --- |
|  | **CENTRO UNIVERSITÁRIO DE PATOS DE MINAS – UNIPAM**  **BACHARELADO EM SISTEMAS DE INFORMAÇÃO**  **SISTEMAS DE BANCO DE DADOS – 4º PERÍODO**  **PROFESSOR RAFAEL MARINHO E SILVA** | |
| **ALUNO (A):\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** | |
| **VALOR: 4 PONTOS** | **DATA: 07/11/2025** |
| A lista de exercícios deverá ser resolvida manualmente e entregue em formato PDF pelo Portal UNIPAM até o dia 07/11/2025. | | |

**LISTA DE EXERCÍCIOS**

Com base nos conceitos e conteúdos sobre Banco de Dados estudados em aulas expositivas e na resolução de exercícios, utilize os exemplos e materiais das aulas práticas, bem como o livro Sistemas de Banco de Dados, 7ª edição (Capítulos 6 e 7), disponível na biblioteca do UNIPAM, para responder a todas as questões a seguir.

**Instruções gerais**

* Conecte-se ao banco plano\_saude fornecido no roteiro e execute os comandos solicitados.
* Para cada comando SQL solicitado anexe um print contendo (i) o comando executado e (ii) o resultado retornado pelo SGBD.
* **Entrega: arquivo PDF**.

**QUESTÕES TEÓRICAS**

|  |
| --- |
| Para cada pergunta abaixo responda consultando o livro.   1. O que é SQL? 2. Explique o conceito de cada categoria DDL, DML, DQL, DTL e DCL. 3. Explique o conceito de normalização: motivos, formas normais (1ª, 2ª, 3ª, BCNF) e como aplicá-las. 4. Defina subclasses, superclasses e herança em modelos relacionais/OR-mapeamento. 5. Explique as alternativas de implementação “tabela por hierarquia”, “tabela por classe concreta” e “tabela por subtipo”. Cite vantagens e desvantagens de cada abordagem. 6. Explique especialização e generalização. Quando usar cada uma no ciclo de projeto de banco de dados? 7. Fundamentos de consultas JOIN: explique os conceitos de cada tipo de consulta, INNER JOIN, LEFT JOIN e RIGHT JOIN. Explique o que cada um retorna e quando usar. |

**QUESTÕES PRÁTICAS**

|  |
| --- |
| 1. Liste todos os pacientes (nome e data\_cadastro)   *SELECT*  *p.nome,*  *pa.data\_cadastro*  *FROM paciente pa*  *JOIN pessoa p ON p.id = pa.pessoa\_id;*   1. Liste todos os médicos (nome, crm, especialidade)   *SELECT*  *p.nome AS nome\_medico,*  *m.crm,*  *e.nome AS nome\_especialidade*  *FROM medico m*  *JOIN pessoa p ON p.id = m.pessoa\_id*  *JOIN especialidade e ON e.id = m.especialidade\_id;*     1. Criar nova tabela endereco e migrar os dados:  * Criar tabela endereco com colunas atômicas (id, logradouro, numero, complemento, bairro, cidade, uf, cep).   ***CREATE TABLE*** *endereco (*  *id SERIAL PRIMARY KEY,*  *logradouro VARCHAR(150) NOT NULL,*  *numero VARCHAR(10) NOT NULL,*  *complemento VARCHAR(50),*  *bairro VARCHAR(80) NOT NULL,*  *cidade VARCHAR(100) NOT NULL,*  *uf CHAR(2) NOT NULL,*  *cep CHAR(8) NOT NULL*  *);*   * Popular endereco com os dados existentes (faça uma transformação simplificada se o formato atual for Rua X, 100).   ***INSERT INTO*** *endereco (logradouro, numero, complemento, bairro, cidade, uf, cep****) VALUES***  *('Rua das Flores', '101', 'Casa', 'Centro', 'São Paulo', 'SP', '01001000'),*  *('Av. Brasil', '202', 'Apto 12', 'Copacabana', 'Rio de Janeiro', 'RJ', '22021000'),*  *('Rua das Palmeiras', '33', NULL, 'Boa Viagem', 'Recife', 'PE', '51021000'),*  *('Rua Santa Clara', '500', 'Bloco B', 'Centro', 'Belo Horizonte', 'MG', '30140000'),*  *('Av. Independência', '77', NULL, 'Centro', 'Porto Alegre', 'RS', '90035900'),*  *('Rua XV de Novembro', '12', 'Sala 3', 'Centro', 'Curitiba', 'PR', '80020010'),*  *('Rua Dom Pedro II', '89', 'Casa 2', 'Centro', 'Fortaleza', 'CE', '60060000'),*  *('Av. das Américas', '1500', 'Loja 5', 'Barra', 'Rio de Janeiro', 'RJ', '22775001'),*  *('Rua Goiás', '444', NULL, 'Funcionários', 'Belo Horizonte', 'MG', '30130000'),*  *('Rua da Praia', '88', NULL, 'Centro', 'Salvador', 'BA', '40015000'),*  *('Av. Paulista', '900', 'Conjunto 1001', 'Bela Vista', 'São Paulo', 'SP', '01311000'),*  *('Rua do Sol', '25', NULL, 'Centro', 'Maceió', 'AL', '57020000'),*  *('Av. João Pessoa', '410', NULL, 'Centro', 'Porto Alegre', 'RS', '90040000'),*  *('Rua Afonso Pena', '155', NULL, 'Centro', 'Campo Grande', 'MS', '79002000'),*  *('Av. Goiás', '320', 'Apto 7', 'Setor Central', 'Goiânia', 'GO', '74005000'),*  *('Rua Maranhão', '200', NULL, 'Centro', 'Natal', 'RN', '59020000'),*  *('Av. Beira-Mar', '700', NULL, 'Meireles', 'Fortaleza', 'CE', '60165000'),*  *('Rua José Bonifácio', '12', NULL, 'Centro', 'Cuiabá', 'MT', '78005100'),*  *('Rua das Oliveiras', '321', NULL, 'Centro', 'Florianópolis', 'SC', '88010000'),*  *('Av. das Torres', '999', NULL, 'Tarumã', 'Curitiba', 'PR', '82820300');*   * Alterar pessoa removendo colunas de endereço e adicionando endereco\_id FK.   ***ALTER TABLE*** *pessoa*  ***DROP COLUMN*** *logradouro,*  ***DROP COLUMN*** *numero;*   * Atualizar pessoa.endereco\_id para referenciar [endereco.id](http://endereco.id).   ***ALTER TABLE*** *pessoa*  ***ADD COLUMN*** *endereco\_id INT,*  ***ADD CONSTRAINT*** *fk\_pessoa\_endereco*  ***FOREIGN KEY*** *(endereco\_id)* ***REFERENCES*** *endereco(id)*  ***ON DELETE SET NULL;***   * Mostrar SELECT p.nome, e.logradouro, e.numero FROM pessoa p JOIN endereco e ON p.endereco\_id = e.id; com print.   ***SELECT*** *p.nome, e.logradouro, e.numero*  ***FROM*** *pessoa p*  ***JOIN*** *endereco e*  ***ON*** *p.endereco\_id = e.id*     1. LEFT JOIN vs RIGHT JOIN: escreva e execute duas queries que relacionem paciente e consulta com LEFT JOIN e RIGHT JOIN (na sequência); capture os resultados e explique a diferença entre eles, indicando quais registros aparecem em cada resultado e por quê. 2. INNER JOIN vs RIGHT JOIN (igualdade de resultados): escreva e execute queries INNER JOIN e RIGHT JOIN para medico + consulta e paciente + consulta, explique em quais circunstâncias os resultados serão idênticos e por quê.   Anexar print dos comandos + resultados |

**Boa Atividade!!!**