

# PROVA TÉCNICA

## Desenvolvedor ORACLE PL/SQL e Rotinas Internas

Candidato:

Data:

Prazo máximo de entrega: 07/08/2025 às 23:59

Enviar arquivo contendo as resposta da prova para: [dalila@volus.com](mailto:dalila@volus.com)

Esta prova prática tem como objetivo avaliar suas habilidades em desenvolvimento Oracle PL/SQL, otimização de SQL, modelagem de dados, resolução de problemas e a capacidade de criar rotinas internas eficientes e eficazes para sistemas ERP, lidando com os desafios diários da Vólus.

A prova será dividida em duas etapas:

1. **Questões de Múltipla Escolha:** Para avaliar conhecimentos teóricos e conceituais de banco de dados Oracle e SQL/PL/SQL.
2. **Questões Práticas:** O candidato deverá aplicar seus conhecimentos de banco de dados Oracle e SQL/PL/SQL em cenários realistas de um ambiente de cartões de crédito.

## INSTRUÇÕES GERAIS

- Leia atentamente cada questão antes de responder.
- Escreva códigos limpos e bem comentados.
- Considere boas práticas de desenvolvimento PL/SQL.
- Pode consultar documentação Oracle oficial.
- Salve seu trabalho regularmente.
- Indique claramente suas respostas.
- Os códigos serão executados e testados para validação das questões práticas.
- Não utilize o ambiente de banco de dados da Vólus para a realização da prova prática.

**Cenário proposto:** A empresa de meio de pagamentos, Tech, precisa de melhorias em seu sistema de gerenciamento de cartão de crédito. O objetivo principal é criar soluções com escalabilidade e performance.

**Instruções:** Para a resolução das questões, o candidato deverá considerar o schema abaixo que representa as entidades essenciais de um sistema de meios de pagamentos:

- **CLIENTES:** Armazenar informações dos clientes.
- **DEPARTAMENTOS:** Armazenar informações dos departamentos dos clientes que serão cadastrados, e consequentemente vinculados aos cartões.
- **CARTOES:** Armazenar informações dos cartões emitidos para os clientes.
- **ESTABELECIMENTOS:** Informações sobre a rede credenciada (lojas, comércio).
- **TRANSACOES:** Armazenar as informações referentes ao registro da transação realizada.
- **FATURAS\_DE\_CLIENTES:** Armazenar informações das faturas mensais dos clientes.
- **FATURAS\_DE\_ESTABELECIMENTOS:** Armazenar informações das faturas mensais dos estabelecimentos.
- **LOG\_ALTERACAO\_LIMITE :** Informações das alterações realizadas no campo limite de crédito da tabela de CARTOES.

## Etapa 1: Questões de Múltipla Escolha

**Instruções:** Para as questões abaixo, justifique suas escolhas.

**1. Qual é a principal diferença entre os comandos TRUNCATE e DELETE?**

- a. O TRUNCATE é um comando DML e o DELETE é um comando DDL.
- b. O TRUNCATE é mais lento, mas permite o uso de uma cláusula WHERE, enquanto o DELETE é mais rápido e não aceita a cláusula WHERE.
- c. O TRUNCATE é um comando DDL que remove todas as linhas de uma tabela de forma rápida e irreversível (sem rollback), enquanto o DELETE é um comando DML que remove linhas individualmente e permite rollback.
- d. O TRUNCATE não pode ser usado em tabelas com chaves primárias, e o DELETE pode.

**Resposta Correta:**

**Justificativa:**

**2. Para que servem os índices em um banco de dados relacional?**

- a. Para garantir a integridade referencial entre as tabelas.
- b. Para acelerar a recuperação de dados (consultas) em grandes tabelas.
- c. Para impedir a inserção de dados duplicados.
- d. Para armazenar procedimentos e funções do banco de dados.

**Resposta Correta:**

**Justificativa:**

**3. Qual é a forma correta de declarar uma variável do tipo string em PL/SQL com um tamanho máximo de 50 caracteres?**

- a. variavel VARCHAR2
- b. variavel VARCHAR2(50)
- c. variavel STRING(50)
- d. variavel CHARACTER(50)

**Resposta Correta:**

**Justificativa:**

4. Em PL/SQL, qual bloco de código é utilizado para tratar erros ou exceções?

- a. IF...THEN...ELSE
- b. LOOP...END LOOP
- c. DECLARE...BEGIN...END
- d. EXCEPTION...WHEN...END

**Resposta Correta:**

**Justificativa:**

5. Qual comando PL/SQL é usado para desfazer todas as alterações feitas na sessão atual desde o último COMMIT ou início da transação?

- a. SAVEPOINT
- b. COMMIT
- c. ROLLBACK
- d. TRUNCATE
- e. ALTER SESSION

**Resposta Correta:**

**Justificativa:**

6. Você precisa criar uma consulta sql que retorne o nome do cartão e o nome de seu departamento. Alguns cartões podem não estar associados a nenhum departamento (department\_id é NULO). A consulta deve sql listar todos os cartões, independentemente de terem um departamento. Qual junção (JOIN) é a mais apropriada?

- a. INNER JOIN
- b. RIGHT OUTER JOIN (com cartão à esquerda)
- c. CROSS JOIN
- d. LEFT OUTER JOIN (com cartão à esquerda)

**Resposta Correta:**

**Justificativa:**

7. Qual das seguintes opções descreve corretamente um 'TRIGGER' de banco de dados no Oracle?

- a. Um bloco de código PL/SQL que é executado manualmente pelo DBA para tarefas de manutenção.
- b. Um procedimento armazenado que é executado automaticamente em resposta a um evento DML (INSERT, UPDATE, DELETE) em uma tabela específica.
- c. Um tipo de visão que contém lógica de programação.
- d. Um sinônimo para uma função que retorna um valor booleano.

**Resposta Correta:**

**Justificativa:**

8. Você precisa escrever uma consulta sql que retorne o número de cartões em cada departamento. Qual das seguintes consultas está correta?

- a. `SELECT nome_departamento, COUNT(*) FROM CARTOES GROUP BY nome_departamento;`
- b. `SELECT id_departamento, COUNT(id_cartao) FROM CARTOES GROUP BY id_departamento;`
- c. `SELECT id_departamento, SUM(id_cartao) FROM CARTOES ORDER BY id_departamento;`
- d. `SELECT COUNT(id_cartao) FROM funcionarios HAVING id_departamento;`

**Resposta Correta:**

**Justificativa:**

9. O que a cláusula PRAGMA AUTONOMOUS\_TRANSACTION faz dentro de um procedimento ou função PL/SQL?

- a. Faz com que o procedimento seja executado com os privilégios do usuário que o definiu (definer's rights).
- b. Permite que o subprograma seja executado em uma transação separada, que pode ser confirmada (COMMIT) ou desfeita (ROLLBACK) independentemente da transação principal.
- c. Otimiza automaticamente o código PL/SQL para uma execução mais rápida.
- d. Impede que o procedimento execute qualquer comando DML.

**Resposta Correta:**

**Justificativa:**

**10. Qual é a principal diferença entre uma FUNCTION e uma PROCEDURE em PL/SQL?**

- a. Uma PROCEDURE pode aceitar parâmetros, enquanto uma FUNCTION não pode.
- b. Uma FUNCTION deve sempre retornar um valor, enquanto uma PROCEDURE não retorna um valor diretamente (mas pode usar parâmetros OUT).
- c. Uma FUNCTION não pode executar comandos DML (INSERT, UPDATE, DELETE), mas uma PROCEDURE pode.
- d. Não há diferença; os termos são intercambiáveis.

**Resposta Correta:**

**Justificativa:**

## Etapa 2: Questões Práticas

### Instruções:

Para as questões abaixo, utilize o schema fornecido (no início da prova) e escreva o código SQL ou PL/SQL necessário para resolver cada questão. Justifique suas escolhas ou explique a lógica.

Procedimentos (procedures), function e trigger PL/SQL que serão desenvolvidos neste questionário poderão lidar com um grande volume de ativos e componentes (milhões de registros), por isso fique atento as seguintes validações:

**Otimização de Performance:** Quais estratégias você aplicaria para garantir que a procedure seja executada de forma eficiente em um cenário de alto volume de dados? Considere a otimização de consultas, manipulação de dados e uso de recursos do banco.

**Tratamento de Erros:** Implemente um bloco de tratamento de exceções robusto na procedure de cadastro de cartão. Quais tipos de erros você anteciparia (ex: NO\_DATA\_FOUND, TOO\_MANY\_ROWS, DUP\_VAL\_ON\_INDEX, erros de integridade, erros inesperados)? Como você os trataria para garantir que a rotina seja resiliente e não falhe silenciosamente, registrando os dados de forma adequada na tabelas?

**Atomicidade e Concorrência:** Como você garantiria a atomicidade das operações e lidaria com potenciais problemas de concorrência se a mesma rotina fosse executada por diferentes processos (pense em cenários de reprocessamento manual)?

**Reusabilidade e Manutenibilidade:** Demonstração de boas práticas de codificação, modularização (via procedures e triggers) para facilitar a manutenção e evolução do sistema.

## Questão 1: Modelar e Popular as entidades (tabelas)

Crie o scripts de modelagem para inserir e popular as tabelas conforme seu conhecimento referente a empresa de meio de pagamentos.

Dica: Valide de forma correta os campos que serão comuns entre as entidades.

## Questão 2: Cadastro de Cartão (PL/SQL)

A Tech precisa de uma rotina para registrar novos cartões para seus clientes. Crie uma procedure PL/SQL chamada PR\_REGISTRAR\_NOVO\_CARTAO. O procedimento deve:

- Inserir os dados do novo cartão na tabela CARTOES.
- Incluir tratamento de erro para caso de p\_numero\_cartao\_hash duplicado ou outros erros de inserção.
- Após a inserção bem-sucedida, deve registrar uma mensagem de sucesso no DBMS\_OUTPUT.

## Questão 3: Uso de Função Analítica

O RH precisa de um relatório que mostre os cartões, limite de crédito e o cartão que possuiu o maior limite no mesmo departamento, para fins de análise de equidade salarial. Para isso, crie a consulta sql (query) que irá retornar o relatório solicitado.

## Questão 4: Consulta de Clientes e Histórico de Transações

A equipe de atendimento ao cliente precisa de uma maneira rápida de visualizar todas as transações de um cliente específico, juntamente com os detalhes do cartão e o nome do estabelecimento onde a transação ocorreu.

Crie uma consulta sql (query) que retorne o NOME\_FANTASIA do cliente, o NUMERO\_CARTAO\_HASH (dos 4 últimos dígitos, se possível, ou o hash completo para simplificação), o VALOR\_TRANSACAO, a DATA\_HORA\_TRANSACAO, o STATUS\_TRANSACAO e o NOME\_FANTASIA do estabelecimento para um ID\_CLIENTE fornecido (ex: ID\_CLIENTE = 101).



### Questão 5: Validação de Saldo Disponível (PL/SQL Function)

Antes de aprovar uma nova transação, o sistema deve validar se o cartão possui limite de crédito disponível suficiente.

Crie uma função PL/SQL chamada FN\_VERIFICAR\_SALDO\_DISPONIVEL que receba o código do cartão e retorne um boolean. Considere que o saldo disponível é o LIMITE\_CREDITO do cartão menos a soma de todas as VALOR\_TRANSACAO com STATUS\_TRANSACAO = 'APROVADA' ou 'PENDENTE' para aquele cartão.

### Questão 6: Processamento de Transação (PL/SQL com Controle de Transação)

Após a validação do saldo, uma transação precisa ser registrada no sistema.

Crie um procedimento PL/SQL chamado PR\_PROCESSAR\_TRANSACAO. O procedimento deve conter:

- Se o saldo for suficiente, deve inserir a transação na tabela TRANSACOES com STATUS\_TRANSACAO = 'APROVADA'.
- Implementar controle de transação (COMMIT/ROLLBACK). Se a inserção falhar por qualquer motivo (ex: cartão inválido, estabelecimento não encontrado, erro de sistema), todas as operações devem ser desfeitas e uma exceção levantada.
- Retornar uma mensagem de sucesso ou falha (DBMS\_OUTPUT).

### Questão 7: Consulta de Agregação e Junção (Multi-tabelas)

Você precisa identificar os 3 departamentos com o maior volume de vendas em valor total do mês referente.

Para isso, crie o consulta sql (query) que irá retornar o relatório solicitado contendo os dados de clientes, departamentos e cartões.

## Questão 8: Faturamento da Rede Credenciada (SQL com Agregação)

No final de cada mês, a Tech precisa gerar um relatório para faturar as taxas de comissão devidas por cada estabelecimento credenciado, com base nas transações processadas naquele mês.

Crie uma consulta sql (query) que retorne o NOME\_FANTASIA do estabelecimento, o CNPJ e o VALOR\_TOTAL\_COMISSAO\_DEVIDA para o mês de JULHO do ano 2025. O VALOR\_TOTAL\_COMISSAO\_DEVIDA deve ser calculado como a soma dos VALOR\_TRANSACAO (apenas transações APROVADAS) de cada estabelecimento, multiplicada pela sua TAXA\_COMISSAO. A saída deve ser ordenada pelo VALOR\_TOTAL\_COMISSAO\_DEVIDA em ordem decrescente.

## Questão 9: Otimização de Consulta e Indexação (Performance)

A consulta que busca transações para um determinado cartão em um longo período (DATA\_HORA\_TRANSACAO) está muito lenta. A tabela TRANSACOES já possui milhões de registros.

Como você identificaria o gargalo (bottleneck) dessa consulta? (Mencione pelo menos uma ferramenta ou técnica do Oracle.)

Qual seria sua primeira proposta de otimização para essa consulta? (Especifique o tipo de objeto de banco de dados e o DDL se aplicável).

## Questão 10: Gatilho de Auditoria - Trigger (PL/SQL)

A Tech precisa auditar todas as alterações de limite de crédito na tabela de cartões por razões de segurança e conformidade.

Crie uma trigger PL/SQL que contenha esse contexto.

## Questão 11: Agendamento automático - Job (PL/SQL)

Os saldos dos cartões precisam ser atualizados diariamente de forma automática às 00:00. Desta maneira, crie o esqueleto de uma procedure que irá realizar essa rotina.