

Alunos: Fábio Emanuell Pereira Milagres, João Pedro de Almeida Dupim e Pedro Henrique Santos.

Exemplo 1:

(a.1) Enunciado ECA (Evento-Condição-Ações)

Evento: O sistema precisa calcular quantos dias uma conta a receber está em atraso.

Condição: Uma conta é considerada em atraso se:

- A data atual for maior que a data de vencimento.
- A conta ainda não tiver sido paga.

Ações:

- Retornar o número de dias em atraso se a conta não foi paga e já está vencida.
- Caso contrário, retornar 0.

(a.3) Documentação e Decisões de Projeto

1. Uso de Função: Criamos uma função para encapsular a lógica do cálculo de atraso, permitindo reutilização em consultas e relatórios.
2. Consulta de Datas: Usamos SELECT INTO para armazenar as datas relevantes e facilitar a lógica de decisão.
3. Cálculo de Atraso: Se a conta não estiver paga (dt_paga IS NULL) e estiver vencida (CURDATE() > dt_venc ou CURRENT_DATE > dt_venc no PostgreSQL), calculamos a diferença entre a data atual e a data de vencimento.
4. Compatibilidade: Criamos versões compatíveis com MySQL e PostgreSQL, adaptando a sintaxe de DATEDIFF() para PostgreSQL, onde a subtração entre datas é feita diretamente.

(a.4) Casos de Teste Interessantes

Codigo	Data_Vencimento	Data_Paga	Data Atual	Resultado Esperado
1	2024-01-01	NULL	2024-02-01	31
2	2024-01-01	2024-01-15	2024-02-01	0
3	2024-03-01	NULL	2024-02-25	0
4	2024-02-01	NULL	2024-02-03	2

1. Caso 1: Conta vencida há um mês e não foi paga, retornando 31 dias de atraso.
2. Caso 2: Conta paga antes da data atual, então o atraso é zero.
3. Caso 3: Conta ainda não venceu, então o atraso é zero.
4. Caso 4: Conta vencida há dois dias e não paga, então o retorno é 2.

Exemplo 2:

(a.1) Enunciado ECA (Evento-Condição-Ações)

Evento: O sistema precisa calcular o total de contas a receber em aberto para um determinado cliente.

Condição: O total deve ser calculado somando os valores de todas as contas cadastradas na tabela ContaReceber que ainda não foram pagas (onde Data_Paga é nula) para um cliente identificado pelo CPF.

Ações:

- Retornar a soma dos valores das contas em aberto para o cliente especificado.
- Se o cliente não tiver contas pendentes, retornar 0.

(a.3) Documentação e Decisões de Projeto

1. Uso de Função: Criamos uma função para facilitar a obtenção do total de contas pendentes de um cliente.
2. Consulta Agregada: Utilizamos SUM(Valor) para obter a soma dos valores das contas a receber.
3. Tratamento de NULL: Usamos IF(SUM(Valor) IS NULL, 0, SUM(Valor)) no MySQL e COALESCE(SUM("Valor"), 0) no PostgreSQL para evitar valores nulos.
4. Compatibilidade: Criamos versões adaptadas para MySQL e PostgreSQL, garantindo que funcionem corretamente em ambos os bancos de dados.

(a.4) Casos de Teste Interessantes

CPF_Cliente	Contas Pendentes	Total de Contas a Receber
12345678901	R\$ 500,00 + R\$ 300,00	R\$ 800,00
98765432100	Nenhuma conta pendente	R\$ 0,00
11122233344	R\$ 150,00	R\$ 150,00

1. Caso 1: Cliente com duas contas pendentes, totalizando R\$ 800,00.
2. Caso 2: Cliente sem contas pendentes, retornando R\$ 0,00.
3. Caso 3: Cliente com apenas uma conta pendente de R\$ 150,00.