

# DTL:

Função para registrar empréstimo

```
CREATE OR REPLACE FUNCTION registrar_emprestimo(  
    p_livro_id INT,  
    p_membro_id INT,  
    p_data_emprestimo DATE,  
    p_data_devolucao_prevista DATE  
)
```

RETURNS TEXT AS \$\$

DECLARE

```
    v_quantidade_disponivel INT;  
  
BEGIN  
  
    -- Verifica a quantidade disponível  
  
    SELECT quantidade_disponivel INTO v_quantidade_disponivel  
    FROM livros  
  
    WHERE id_livro = p_livro_id;  
  
    IF v_quantidade_disponivel > 0 THEN  
  
        -- Atualiza quantidade  
  
        UPDATE livros  
  
        SET quantidade_disponivel = quantidade_disponivel - 1  
  
        WHERE id_livro = p_livro_id;  
  
        -- Insere empréstimo  
  
        INSERT INTO empréstimos (id_livro, id_membro, data_emprestimo,  
data_devolucao_prevista, status)  
  
        VALUES (p_livro_id, p_membro_id, p_data_emprestimo,  
p_data_devolucao_prevista, 'EMPRESTADO');  
  
        RETURN 'Empréstimo registrado com sucesso!';  
  
    ELSE  
  
        RETURN 'Não há livros disponíveis.';
```

END IF;

END;

\$\$ LANGUAGE plpgsql;

-- Exemplo de chamada correta (fora da função):

-- BEGIN;

-- SELECT registrar\_emprestimo(1, 1, '2025-05-30', '2025-06-04');

-- COMMIT;

-- Função para registrar devolução

CREATE OR REPLACE FUNCTION registrar\_devolucao(

    p\_emprestimo\_id INT,

    p\_data\_devolucao DATE

)

RETURNS TEXT AS \$\$

DECLARE

    v\_livro\_devolvido\_id INT;

    v\_rows\_updated INT;

BEGIN

-- Atualiza o empréstimo se ainda não foi devolvido

UPDATE empréstimos

SET

    status = 'DEVOLVIDO',

    data\_devolucao = p\_data\_devolucao

WHERE

    id\_emprestimo = p\_emprestimo\_id

    AND status = 'EMPRESTADO'

RETURNING id\_livro INTO v\_livro\_devolvido\_id;

IF FOUND THEN

-- Atualiza a quantidade de livros disponíveis

UPDATE livros

SET quantidade\_disponivel = quantidade\_disponivel + 1

```
WHERE id_livro = v_livro_devolvido_id;
```

```
RETURN 'Devolução registrada com sucesso!';
```

```
ELSE
```

```
RETURN 'Empréstimo não encontrado ou já devolvido.';
```

```
END IF;
```

```
EXCEPTION
```

```
WHEN OTHERS THEN
```

```
RETURN 'Ocorreu um erro ao registrar a devolução: ' || SQLERRM;
```

```
END;
```

```
$$ LANGUAGE plpgsql
```