

## Processando transações

Abaixo,, temos o script de criação de uma tabela que será usada para exemplificar a utilização de transações.

```
create table conta(  
    conta int not null primary key,  
    nome varchar(30) not null,  
    saldo numeric(8,2)  
)
```

Observe o método abaixo, que cria uma transação para realizar uma transferência entre duas contas correntes.

```
public void transferencia(int contaOrigem, double valor, int contaDestino) {  
  
    Connection con = null;  
  
    try {  
  
        Class.forName("org.apache.derby.jdbc.EmbeddedDriver");  
        con = DriverManager  
  
            .getConnection("jdbc:derby://localhost:1527/estacio",  
                           "estacio", "estacio");  
  
        // Iniciamos a transação  
        con.setAutoCommit(false);  
  
        PreparedStatement ps = con  
            .prepareStatement("select * from conta  
where conta = ?");  
        ps.setInt(1, contaOrigem);  
        ResultSet rs = ps.executeQuery();  
        // Verifica se a contaOrigem existe  
        if (rs.next()) {  
            // Verifica se o saldo é suficiente para realizar  
a  
            // transferência  
            if (rs.getDouble("saldo") > valor) {  
                // Decrementa o saldo da conta de origem  
                ps = con.prepareStatement("update conta set  
saldo = (saldo - ?) where conta = ?");  
                ps.setInt(2, contaOrigem);  
                ps.setDouble(1, valor);  
                ps.execute();  
  
                ps = con.prepareStatement("select * from  
conta where conta = ?");  
                ps.setInt(1, contaDestino);  
                rs = ps.executeQuery();  
                // Testa se a conta destino existe, caso  
contrário é lançada  
                // uma SQLException  
                if (rs.next()) {  
                    // Incrementa o saldo da conta de  
destino
```

```
ps = con.prepareStatement("update
conta set saldo = (saldo + ?) where conta = ?");
ps.setInt(2, contaDestino);
ps.setDouble(1, valor);
ps.execute();
} else {
    // Esse é o momento chave, pois se
    // exceção a conta origem teria seu
    // decrementado e a destino não
    throw new SQLException("Conta destino
não existe");
}

} else {
    throw new SQLException("Saldo
insuficiente");
}
} else {
    throw new SQLException("Conta origem não existe");
}

// Executa o commit
con.commit();
JOptionPane.showMessageDialog(null,
    "Operação realizada com sucesso!");

} catch (ClassNotFoundException cne) {
    cne.printStackTrace();
} catch (SQLException e) {
    try {
        // Caso ocorra uma exceção executamos o
        con.rollback();
        JOptionPane.showMessageDialog(null, e);
    } catch (SQLException sql) {
        sql.printStackTrace();
    }
}

}
```