

Aluno: Matheus Teixeira

Testes Unitários voltado para sistema bancário.

1. Crie a base de dados para testes:

- Crie pelo menos 2 modelos de Cliente e ContaBancaria e armazene-os no DAO.

2. Crie testes para todos os métodos da classe ClienteDAO:

- **getClientesBanco**: 1) Faça uma pesquisa no banco de dados que resgate todos clientes. 2) Verifique a existência desses dados. 3) Verifique se o retorno da lista não é nulo. 4) Verifique a integridade dos dados de pelo menos um Cliente.
- **pesquisaCliente**: 1) Faça uma pesquisa de um cliente específico. 2) Verifique a existência desses dados. 3) Verifique se o retorno não é nulo. 4) Verifique a integridade dos dados do Cliente.
- **adicionaCliente**: 1) Instancie e preencha os dados de um novo cliente. 2) Insira esse cliente no banco de dados. 3) Verifique se a inserção foi realizada com sucesso. 4) Verifique a integridade dos dados.
- **removeCliente**: 1) Remova o cliente anteriormente inserido. 2) Verifique se a remoção foi efetuada com sucesso. 3) Verifique a integridade dos dados não deletados.
- **clienteAtivo**: 1) Faça a verificação booleana com um caso verdadeiro e outro falso.
- **limpa**: 1) Execute este por último. 2) Execute o método, assim limpando toda base de dados armazenada. 3) Verifique se a limpeza foi feita corretamente.
- **valididade**: 1) Teste o tratamento de exceções com 3 idades diferentes.

3. Crie testes para todos os métodos da classe ContaCorrenteDAO:

- **getContasDoBanco**: Faça uma pesquisa no banco de dados que resgate todas as contas. 2) Verifique a existência desses dados. 3) Verifique se o retorno da lista não é nulo. 4) Verifique a integridade dos dados de pelo menos uma Contas.
- **pesquisaConta**: 1) Faça uma pesquisa de um conta específica. 2) Verifique a existência desses dados. 3) Verifique se o retorno não é nulo. 4) Verifique a integridade dos dados da Conta.

- **adicionaConta:** 1) Instancie e preencha os dados de uma nova conta. 2) Insira essa conta no banco de dados. 3) Verifique se a inserção foi realizada com sucesso. 4) Verifique a integridade dos dados.
- **removeConta:** 1) Remova a conta anteriormente inserida. 2) Verifique se a remoção foi efetuada com sucesso. 3) Verifique a integridade dos dados não deletados.
- **contaAtiva:** 1) Faça a verificação booleana com um caso verdadeiro e outro falso.
- **transfereValor:** 1) Teste o tratamento de exceções com 2 transferências diferentes, uma em que o emissor possui saldos suficientes, e outra em que o emissário não possui saldo o suficiente.