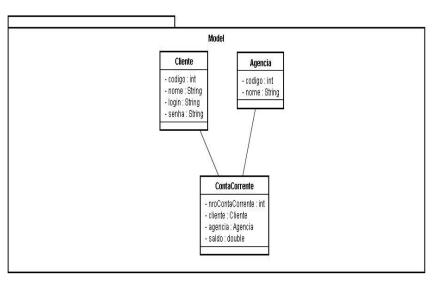
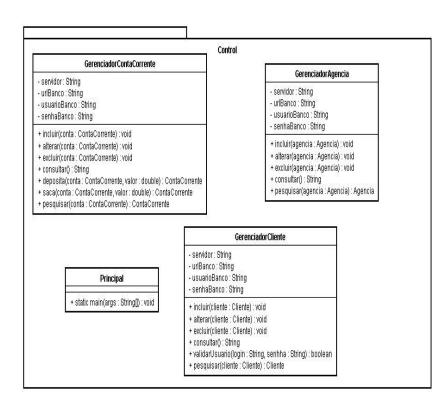
Diagrama de Classes Prova Módulo 2

Diagrama de Classes – Prova Módulo 2.





View FormLogin <<Abstract>> - ILogin : jLabel FormPaiManutencao - txtLogin : iTextField FormMenu - ISenha : jLabel - btnlnclusao : ¡Button - txtSenha : iPasswordField - btnAlteracao : iButton - barra : JMenuBar - btnOK: jButton - btnExclusao : iButton - manutencao : JMenu - btnCancelar : iButton - btnConsulta : ¡Button movimentacao: JMenu - gerenciadorCliente : GerenciadorCliente - btnConfirmar : ¡Button cliente: JMenultem - btnCancelar : jButton agencia: JMenultem + bCancelar.actionPerformed(ActionEvent e : int) : void - btnPesquisar: iButton contaCorrente : JMenultem + bOK.actionPerformed(ActionEvent e : int) : void movimentaContaCorrente : JMenultem - posicionaBotaoPesquisar(coluna : int, linha : int, largura : int, altura : int) : void - sair: JMenultem + posicionaBotoes(linha : int, largura : int, altura : int, margem : int, distancia : int) : void + habilitaBotoes() : void cliente.actionPerformed(ActionEvent e : int) : void FormManutencaoCliente + desabilitaBotoes() : void + agencia.actionPerformed(ActionEvent e : inf) : void - cliente : Cliente + abstract posicionaFocoPrimeiro() : void + contaCorrente.actionPerformed(ActionEvent e : int) : void gerenciadorCliente : GerenciadorCliente + abstract habilitaCampos() : void + movimentaContaCorrente.actionPerformed(ActionEvent e : inf) : void - ICodigo : JLabel + abstract desabilitaCampos() : void + sair.actionPerformed(ActionEvent e : int) : void - tFCodigo : JTextField + abstract limpaCampos() : void - INome : JLabel + abstract acaolnclusao() : void - tFNome : JTextField + abstract acaoAlteracao() : void - ILogin : JLabel + abstract acaoExclusao() : void - tFLogin : JTextField + abstract acaoConsulta() : void - ISenha : JLabel + abstract acaoPesquisa() : void - tPSenha: JPasswordField + posicionaFocoPrimeiro(): void + habilitaCampos() : void FormMovimentoContaCorrente + desabilitaCampos() : void contaCorrente : ContaCorrente + limpaCampos(): void gerenciadorContaCorrente : GerenciadorContaCorrente + acaolnclusao(): void - INroContaCorrente : JLabel + acaoAlteracao() : void tFNroContaCorrente: JTextField + acaoExclusao() : void | CodCliente : JLabel + acaoConsulta() : void FormManutencaoContaCorrente - tFCodCliente : JTextField + acaoPesquisa() : void - contaCorrente : ContaCorrente NomeCLiente: JLabel gerenciadorContaCorrente : GerenciadorContaCorrente ISaldo: JLabel - INroContaCorrente : JLabel - tFSaldo : JTextField FormManutencaoAgencia -tFNroContaCorrente: JTextField - IValor: JLabel - agencia : Agencia - ICodCliente : JLabel - tFValor : JTextField - gerenciadorAgencia : GerenciadorAgencia - tFCodCliente : JTextField - bPesquisar: JButton - ICodAgencia : JLabel - ICodAgencia : JLabel bDepositar: JButton -tFCodAgencia: JTextField - tFCodAgencia : JTextField - bSacar : JButton INome : JLabel - ISaldo : JLabel + bPesquisar.actionPerformed(ActionEvent e : int) : void -tFNome: JTextField tFSaldo: JTextField + bDepositar.actionPerformed(ActionEvent e : int) : void + posicionaFocoPrimeiro() : void + posicionaFocoPrimeiro() : void + bSacar.actionPerformed(ActionEvent e : int) : void + habilitaCampos() : void + habilitaCampos() : void + desabilitaCampos() : void + desabilitaCampos(): void + limpaCampos(): void + limpaCampos(): void + acaolnclusao() : void + acaolnclusao() : void + acaoAlteracao() : void + acaoAlteracao() : void + acaoExclusao() : void + acaoExclusao() : void + acaoConsulta() : void + acaoConsulta(): void + acaoPesquisa() : void + acaoPesquisa() : void

Diagrama de Entidade e Relacionamento Prova Módulo II



Nome do Banco de Dados: Prova

Cliente.

Nome	Tipo	Tamanho	Nulo?
@Codigo	Integer	5	Não
Nome	Varchar	50	Não
Login	Varchar	10	Não
Senha	Varchar	10	Não

ContaCorrente.

Nome	Tipo	Tamanho	Nulo?
@Numero	Integer	5	Não
#CodCliente	Integer	5	Não
#CodAgencia	Integer	5	Não
Saldo	Double		Não

Agencia.

Nome	Tipo	Tamanho	Nulo?
@Codigo	Integer	5	Não
Nome	Varchar	30	Não

@Chave Primária #Chave Estrangeira

Descritivo das Classes:

Package Model

Cliente: Criar os atributos, os contrutores com e sem campos e os getters e setters.

Agencia: Criar os atributos, os contrutores com e sem campos e os getters e setters.

ContaCorrente: Criar os atributos, os contrutores com e sem campos e os getters e setters.

Package View

FormPaiManutencao

Criar os Atributos segundo os tipos mostrados no diagrama.

Nome	Descrição
posicionaBotãoPesquisar	Recebe os parâmetros e usa para fazer a chamada do método setBounds do botão
	Pesquisar.
posicionaBotoes	Recebe os parâmetros que usa para posicionar os botões de inclusão, alteração,
	exclusão e consulta, de acordo com o tamanho solicitado pela classe filhar. Igual ao
	visto em sala.
habilitaBotoes	Aciona o método setEnabled dos botões para true, menos para o botão pesquisar que é
	alterado para false
desabilitaBotoes	Aciona o método setEnabled dos botões para false, menos para o botão pesquisar que
	é true
**** Métodos Abstrados.	Lista a Assinatura do método como sendo abstrato.

FormManutencaoCliente

Criar os Atributos segundo os tipos mostrados no diagrama.

Nome	Descrição	
Construtor	Inicializa e posiciona os atributos do form, adicionando-os ao form juntamente com os	
	atributos herdados da classe FormManutencaoPai.	
posicionaFocoPrimeiro	Usa o método requestFocus do textfield do primeiro campo, para posicionar o foco	
	nele quando do início da execução do objeto da classe.	
habilitaCampos	Usa o método setEnabled dos objetos TextField colocando-os como true.	
desabilitaCampos	Usa o método setEnabled dos objetos TextField colocando-os como false.	
limpaCampos	Usa o método setText para colocar o valor "" em todos os objetos TextField da Classe.	
acaoInclusao	Chama o método incluir da classe GerenciadorCliente passando um objeto da classe	
	cliente, preenchido.	
acaoAlteracao	Chama o método alterar da classe GerenciadorCliente passando um objeto da classe	
	cliente, preenchido.	
acaoExclusao	Chama o método excluir da classe GerenciadorCliente passando um objeto da classe	
	cliente, preenchido, ao menos com o código.	
acaoConsulta	Chama o método consutar da classe GerenciadorCliente recebendo como retorno uma	
	String com uma lista com todos os Clientes e exibe a String utilizando a classe	
	JOptionPane	
acaoPesquisar	Chama o método Pesquisar da classe GerenciadorCliente passando como parâmetro	
	um objeto da classe cliente com, pelo menos o atributo código preenchido, recebendo	
	como retorno um objeto da classe cliente. Seus atributos devem ser movidos para seus	
	respectivos campos do Form.	

Form Manutenca o Agencia

Criar os Atributos segundo os tipos mostrados no diagrama.

Nome	Descrição
Construtor	Inicializa e posiciona os atributos do form, adicionando-os ao form juntamente com os
	atributos herdados da classe FormManutencaoPai.
posicionaFocoPrimeiro	Usa o método requestFocus do textfield do primeiro campo, para posicionar o foco
	nele quando do início da execução do objeto da classe.
habilitaCampos	Usa o método setEnabled dos objetos TextField colocando-os como true.
desabilitaCampos	Usa o método setEnabled dos objetos TextField colocando-os como false.
limpaCampos	Usa o método setText para colocar o valor "" em todos os objetos TextField da Classe.
acaoInclusao	Chama o método incluir da classe Gerenciador Agencia passando um objeto da classe
	Agencia, preenchido.
acaoAlteracao	Chama o método alterar da classe Gerenciador Agencia passando um objeto da classe
	Agencia, preenchido.
acaoExclusao	Chama o método excluir da classe GerenciadorAgencia passando um objeto da classe
	Agencia, preenchido, ao menos com o código.
acaoConsulta	Chama o método consutar da classe GerenciadorAgencia recebendo como retorno uma
	String com uma lista com todos as Agencias e exibe a String utilizando a classe
	JOptionPane
acaoPesquisar	Chama o método Pesquisar da classe Gerenciador Agencia passando como parâmetro
	um objeto da classe Agencia com, pelo menos o atributo código preenchido,
	recebendo como retorno um objeto da classe Agencia. Seus atributos devem ser
	movidos para seus respectivos campos do Form.

FormManutencaoContaCorrente

Criar os Atributos segundo os tipos mostrados no diagrama.

Nome	Descrição
Construtor	Inicializa e posiciona os atributos do form, adicionando-os ao form juntamente com os
	atributos herdados da classe FormManutencaoPai.
posicionaFocoPrimeiro	Usa o método requestFocus do textfield do primeiro campo, para posicionar o foco
	nele quando do início da execução do objeto da classe.
habilitaCampos	Usa o método setEnabled dos objetos TextField colocando-os como true.

desabilitaCampos	Usa o método setEnabled dos objetos TextField colocando-os como false.
limpaCampos	Usa o método setText para colocar o valor "" em todos os objetos TextField da Classe.
acaoInclusao	Chama o método incluir da classe GerenciadorContaCorrente passando um objeto da
	classe ContaCorrente, preenchido.
acaoAlteracao	Chama o método alterar da classe GerenciadorContaCorrente passando um objeto da
	classe ContaCorrente, preenchido.
acaoExclusao	Chama o método excluir da classe GerenciadorContaCorrente passando um objeto da
	classe ContaCorrente, preenchido, ao menos com o código.
acaoConsulta	Chama o método consutar da classe GerenciadorContaCorrente recebendo como
	retorno uma String com uma lista com todos as Contas Correntes e exibe a String
	utilizando a classe JOptionPane
acaoPesquisar	Chama o método Pesquisar da classe GerenciadorContaCorrente passando como
	parâmetro um objeto da classe ContaCorrente com, pelo menos o atributo código
	preenchido, recebendo como retorno um objeto da classe ContaCorrente. Seus
	atributos devem ser movidos para seus respectivos campos do Form.

FormMenu

Criar os Atributos segundo os tipos mostrados no diagrama.

Nome	Descrição
Construtor	Inicializa e posiciona os atributos do form, adicionando-os ao form.
cliente.actionPerformed	Instancia um objeto da Classe FormManutencaoCliente, altera o atributo para
	fechamento e torna-o visível.
agencia.actionPerformed	Instancia um objeto da Classe FormManutencaoAgencia,, altera o atributo para
	fechamento e torna-o visível.
contaCorrente.actionPerformed	Instancia um objeto da Classe FormManutencaoContaCorrente, altera o atributo
	para fechamento e torna-o visível.
movimentaContaCorrente.actionPerformed	Instancia um objeto da Classe FormManutencaoMovimentoContaCorrente, altera
	o atributo para fechamento e torna-o visível.
Sair.actionPerformed	Fecha o programa.

FormMovimentaContaCorrente

Criar os Atributos segundo os tipos mostrados no diagrama.

Nome	Descrição
Construtor	Inicializa e posiciona os atributos de form, adicionando-os ao form. Observações
	o botão pesquisa deve permitir a pesquisa da conta corrente e só permitir
	operações de Debitar e Creditar se a pesquisa tiver localizado a conta Os
	campos tFCodCliente, tFCodAgencia,tFSaldo devem ficar desabilitados só
	estando habilitados os campos lFNroContaCorrente e lFValor.
bPesquisar.actionPerformed	Inicializa um objeto da classe ContaCorrente com o campo nroContaCorrente
	preenchido e chama o método pesquisar da classe GerenciadorContaCorrente,
	recebe como retorno um objeto da classe ContaCorrente, exibindo seus campos
	no form e posicionando o cursor no campo valor para que seja preenchido.
bDepositar.actionPerformed	Chama o método depositar da classe GerenciadorContaCorrente, passando como
	parâmetros um objeto da classe ContaCorrente e o valor do movimento
	recebendo como retorno um objeto da classe ContaCorrente que deve ter seus
	atributos mostrados no form.
bSacar.actionPerformed	Chama o método sacar da classe GerenciadorContaCorrente, passando como
	parâmetros um objeto da classe ContaCorrente e o valor do movimento
	recebendo como retorno um objeto da classe ContaCorrente que deve ter seus
	atributos mostrados no form.

FormLogin

Criar os Atributos segundo os tipos mostrados no diagrama.

Nome	Descrição
Construtor	Inicializa e posiciona os atributos de form, adicionando-os ao form.
bOk.actionPerformed	Inicializa um objeto do tipo GerenciaCliente acessando o método validaUsuario,
	passando como parâmetros o login e a senha digitados testando se o login e a
	senha estão cadastrados se estiverem cria uma instancia da classe FormMenu e
	destrói sua instância senão é cadastrado dá mensagem e volta a ler o login.
bSair.actionPerformed	Fecha o programa

Package Control

Principal

Nome	Descrição
main(String[] args)	Cria uma instância da classe FormLogin, altera o fechamento default para fechar o
	programa.

GerenciadorAgencia

Criar os Atributos segundo os tipos mostrados no diagrama. Verifique qual o usuário e senha do banco e se necessário altere os valores destes atributos de acordo com o que foi feito na instalação do Banco d dados

Nome	Descrição
incluir(Agencia agencia):void	Estabelece a conexão com o banco de dados e executa o comando de inserção,
	passando os atributos do objeto agencia recebido.
alterar(Agencia agencia):void	Estabelece a conexão com o banco de dados e executa o comando de alteração,
	passando os atributos do objeto agencia recebido.
excluir(Agencia agencia):void	Estabelece a conexão com o banco de dados e executa o comando de exclusão
	passando como parâmetro o atributo código do objeto agencia recebido.
Consultar():String	Estabelece conexão com o banco de dados executa o comando de percorrimento
	da tabela Agencia, colocando os valores de cada registgro em uma String e
	usando \n para separar um registro do outro. Coloque também o nome do campo
	antes de colocar o valor do campo na tabela. Ao final do percorrimento devolver
	a String a classe que a chamou.
Pesquisar(Agencia agencia):Agencia	Recebe como parâmetro um objeto da classe Agencia com o código preenchido,
	efetua a pesquisa no BD, completa os atributos do objeto agencia com as
	informações obtidas do BD e retorna o objeto com os atributos preenchidos, ao
	método que o chamou.

GerenciadorCliente

Criar os Atributos segundo os tipos mostrados no diagrama. Verifique qual o usuário e senha do banco e se necessário altere os valores destes atributos de acordo com o que foi feito na instalação do Banco d dados

Nome	Descrição
incluir(Cliente cliente):void	Estabelece a conexão com o banco de dados e executa o comando de inserção,
	passando os atributos do objeto cliente recebido.
alterar(Cliente cliente):void	Estabelece a conexão com o banco de dados e executa o comando de alteração,
	passando os atributos do objeto cliente recebido.
excluir(Cliente cliente):void	Estabelece a conexão com o banco de dados e executa o comando de exclusão
	passando como parâmetro o atributo código do objeto cliente recebido.
consultar():String	Estabelece conexão com o banco de dados executa o comando de percorrimento
	da tabela Cliente, colocando os valores de cada registro em uma String e usando \
	n para separar um registro do outro. Coloca também o nome do campo antes de
	colocar o valor do campo na tabela. Ao final do percorrimento devolver a String
	a classe que a chamou.
validarUsuario(String login, String	Estabelece conexão com o banco de dados e executa uma pesquisa na tabela
senha):Boolean	Cliente verificando a existência de um cliente cadastrado com o login e a senha
	passados por parâmetro se houver retorna true se não encontrar retorna false.
pesquisar(Usuario usuario):usuario	Recebe como parâmetro um objeto da classe Usuario com o código preenchido,
	efetua a pesquisa no BD, completa os atributos do objeto usuario com as
	informações obtidas do BD e retorna o objeto com os atributos preenchidos, ao
	método que o chamou.

GerenciadorContaCorrente

Criar os Atributos segundo os tipos mostrados no diagrama. Verifique qual o usuário e senha do banco e se necessário altere os valores destes atributos de acordo com o que foi feito na instalação do Banco d dados

Nome	Descrição
incluir(ContaCorrente contaCorrente):void	Estabelece a conexão com o banco de dados e executa o comando de inserção,
	passando os parâmetros do objeto contaCorrente recebido.
alterar(ContaCorrente contaCorrente):void	Estabelece a conexão com o banco de dados e executa o comando de alteração,
	passando os parâmetros do objeto contaCorrente recebido.
excluir(ContaCorrente contaCorrente):void	Estabelece a conexão com o banco de dados e executa o comando de exclusão
	passando os parâmetro o atributo código do objeto contaCorrente recebido.
consultar ():String	Estabelece conexão com o banco de dados executa o comando de percorrimento
	da tabela contaCorrente, colocando os valores de cada registro em uma String e
	usando \n para separar um registro do outro. Coloca também o nome do campo
	antes de colocar o valor do campo na tabela. Ao final do percorrimento devolver
	a String a classe que a chamou.
depositar(contaCorrente contaCorrente,	Estabelece conexão com o banco de dados, recupera o valor do saldo da
valor double):ContaCorrente	contaCorrente, soma o valor do parâmetro valor e grava na tabela do banco.
	Retorna o objeto contaCorrente com o saldo alterado para o objeto que o
	chamou.
sacar(contaCorrente contaCorrente, valor	Estabelece conexão com o banco de dados, recupera o valor do saldo da
double):ContaCorrente	contaCorrente, subtrai o valor do parâmetro valor e grava na tabela do banco.
	Retorna o objeto contaCorrente com o saldo alterado para o objeto que o
	chamou.

pesquisar(ContaCorrente	Recebe como parâmetro um objeto da classe ContaCorrente com o número da
contaCorrente):ContaCorrente	conta preenchido, efetua a pesquisa no BD, completa os atributos do objeto
	usuario com as informações obtidas do BD e retorna o objeto com os atributos
	preenchidos, ao método que o chamou. Observação, você deve obter também os
	demais campos das classes Agencia e Usuario, relacionados a ContaCorrente
	pesquisada.