Aluno

Projeto Final

Faça um sistema em Java. O sistema deverá realiza as principais operações de cadastrar, localizar, editar e excluir.

Todas as informações serão salvas em arquivo.txt.

O sistema deverá haver classes, herança, polimorfismo.

Crie classes com seus atributos e métodos. Todos eles deverão ter modificadores de acesso.

O projeto final é individual.

Descreva seu sistema e liste as principais ações que ele deverá ter.

Depois que as classes estiverem criadas, crie o diagrama de classe e print aqui.

Dica.

Crie um novo projeto Java.

Crie uma classe “principal” com void main().

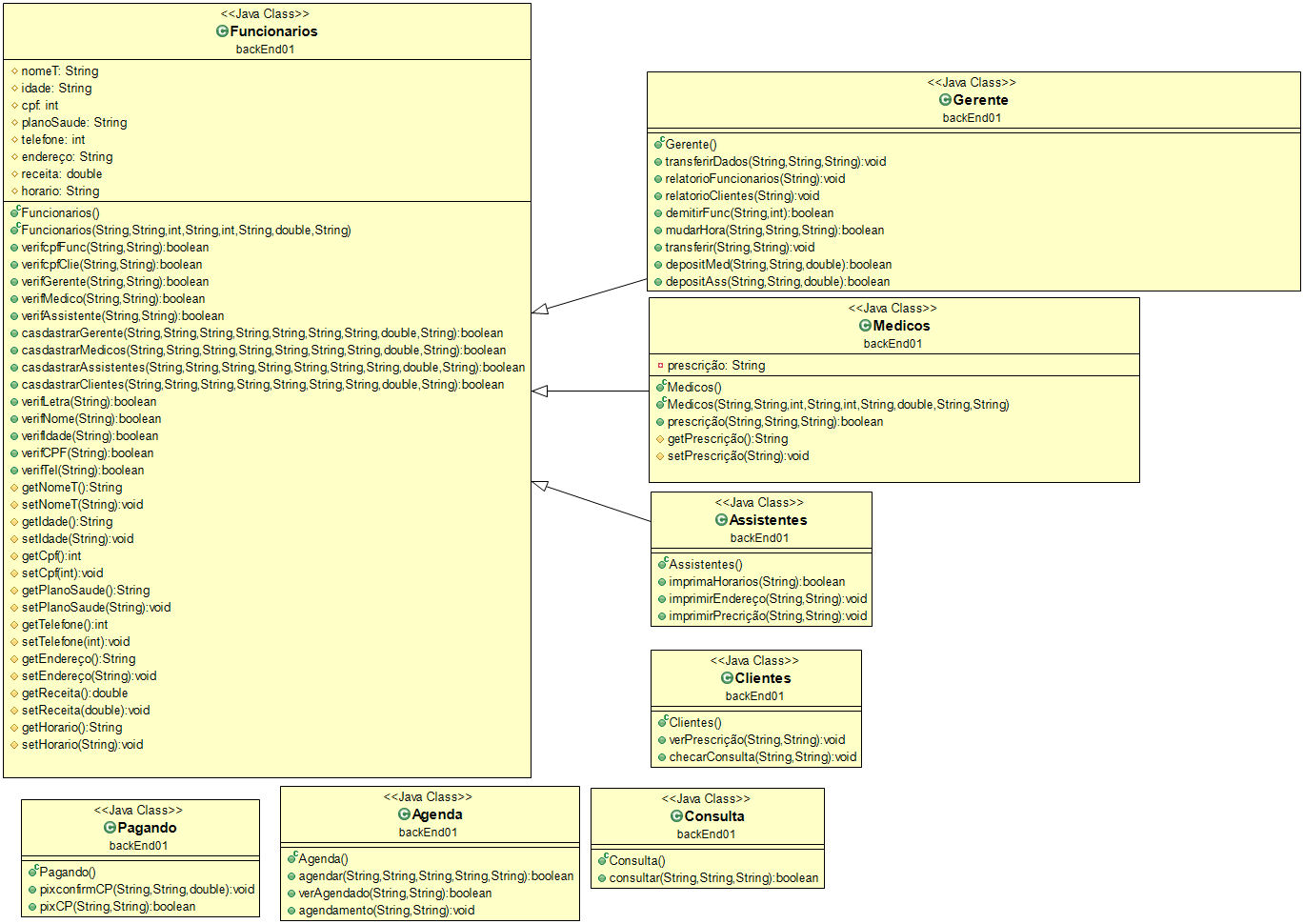
Crie as demais classes que o void main irá chamar.

Instancie a classe Scanner no void main e passe como argumento.

Leia todas as informações como string e converta se for necessário.

Use try/catch().

Faça os tratamentos dos dados.



**package** frontEnd01;

**import** java.util.Scanner;

**import** backEnd01.Pagando;

**import** backEnd01.Gerente;

**import** backEnd01.Medicos;

**import** backEnd01.Agenda;

**import** backEnd01.Assistentes;

**import** backEnd01.Clientes;

**import** backEnd01.Consulta;

**import** backEnd01.Funcionarios;

**public** **class** Consultorio {

**public** **static** **void** main(String[] args) {

// **TODO** Auto-generated method stub

// File caminhok= new File(".");

// String caminhoOrig = caminhok.getAbsolutePath();

// caminhoOrig = caminhoOrig.replace(".","");

// caminhoOrig = caminhoOrig + "Hospital\\";

String caminho = "C:\\Users\\pedro\\OneDrive\\Documentos\\Hospital\\";

Scanner receba = **new** Scanner(System.***in***);

String op;

**do** {

System.***out***.println("Bem vindo ao Hospital:");

System.***out***.println("Digite seu cpf (apenas os numeros) para login");

System.***out***.println("Digite cpf 1 para Gerente");

System.***out***.println("Digite cpf 2 ao 5 para Medico");

System.***out***.println("Digite cpf 5 ao 9 para Assistente");

System.***out***.println("Digite cpf 10 ao 13 para Clientes");

System.***out***.println("\nS - para sair..");

op = receba.nextLine();

Funcionarios g = **new** Funcionarios();

**if** (g.verifcpfFunc(caminho, op)) {

System.***out***.println("Login concluido");

**if** (g.verifGerente(caminho, op)) {

System.***out***.println("Bem vindo Gerente, oque deseja fazer?:");

System.***out***.println("0 - Cadastrar Gerente");

System.***out***.println("1 - Cadastrar Medicos");

System.***out***.println("2 - Cadastrar Assistentes");

System.***out***.println("3 - Cadastrar Pacientes");

System.***out***.println("4 - Transferir dados para outro arquivo");

System.***out***.println("5 - Relatorio de Funcionarios");

System.***out***.println("6 - Relatorio de Pacientes");

System.***out***.println("7 - Demitir Funcionario");

System.***out***.println("8 - Alterar Horario de Trabalho");

System.***out***.println("9 - Bonus Salarial Medico");

System.***out***.println("10 - Bonus Salarial Assistente");

String opGe = receba.nextLine();

**switch** (opGe) {

**case** "0": {

System.***out***.println("Cadastrando...");

String nomeT;

String idade;

String cpf;

String telefone;

**do** {

System.***out***.print("Digite o Nome:");

nomeT = receba.nextLine();

nomeT = nomeT.replace(" ", " ");

}**while**(!g.verifNome(nomeT));

**do** {

System.***out***.print("Digite a Idade:");

idade = receba.nextLine();

}**while**(!g.verifIdade(idade));

**do** {

System.***out***.print("Digite o CPF:");

cpf = receba.nextLine();

}**while**(!g.verifCPF(cpf));

System.***out***.print("Diga o Plano de Saude:");

String planoSaude = receba.nextLine();

**do** {

System.***out***.print("Digite o Telefone:");

telefone = receba.nextLine();

}**while**(!g.verifTel(telefone));

System.***out***.print("Digite o Endereço:");

String endereço = receba.nextLine();

**double** receita = Double.*parseDouble*("0");

System.***out***.print("Digite o Horario de Trabalho:");

String horario = receba.nextLine();

**if** (g.casdastrarGerente(caminho, nomeT, idade, cpf, planoSaude, telefone, endereço,receita,

horario)) {

System.***out***.println("Cadastrado com sucesso!!\n");

} **else** {

System.***out***.println("Falha no Cadastro!.\n");

}

**break**;

}

**case** "1": {

System.***out***.println("Cadastrando...");

String nomeT;

String idade;

String cpf;

String telefone;

**do** {

System.***out***.print("Digite o Nome:");

nomeT = receba.nextLine();

nomeT = nomeT.replace(" ", " ");

}**while**(!g.verifNome(nomeT));

**do** {

System.***out***.print("Digite a Idade:");

idade = receba.nextLine();

}**while**(!g.verifIdade(idade));

**do** {

System.***out***.print("Digite o CPF:");

cpf = receba.nextLine();

}**while**(!g.verifCPF(cpf));

System.***out***.print("Diga o Plano de Saude:");

String planoSaude = receba.nextLine();

**do** {

System.***out***.print("Digite o Telefone:");

telefone = receba.nextLine();

}**while**(!g.verifTel(telefone));

System.***out***.print("Digite o Endereço:");

String endereço = receba.nextLine();

**double** receita = Double.*parseDouble*("0");

System.***out***.print("Digite o Horario de Trabalho:");

String horario = receba.nextLine();

**if** (g.casdastrarMedicos(caminho, nomeT, idade, cpf, planoSaude, telefone, endereço,receita,

horario)) {

System.***out***.println("Cadastrado com sucesso!!\n");

} **else** {

System.***out***.println("Falha no Cadastro!.\n");

}

**break**;

}

**case** "2": {

System.***out***.println("Cadastrando...");

String nomeT;

String idade;

String cpf;

String telefone;

**do** {

System.***out***.print("Digite o Nome:");

nomeT = receba.nextLine();

nomeT = nomeT.replace(" ", " ");

}**while**(!g.verifNome(nomeT));

**do** {

System.***out***.print("Digite a Idade:");

idade = receba.nextLine();

}**while**(!g.verifIdade(idade));

**do** {

System.***out***.print("Digite o CPF:");

cpf = receba.nextLine();

}**while**(!g.verifCPF(cpf));

System.***out***.print("Diga o Plano de Saude:");

String planoSaude = receba.nextLine();

**do** {

System.***out***.print("Digite o Telefone:");

telefone = receba.nextLine();

}**while**(!g.verifTel(telefone));

System.***out***.print("Digite o Endereço:");

String endereço = receba.nextLine();

**double** receita = Double.*parseDouble*("0");

System.***out***.print("Digite o Horario de Trabalho:");

String horario = receba.nextLine();

**if** (g.casdastrarAssistentes(caminho, nomeT, idade, cpf, planoSaude, telefone, endereço, receita,

horario)) {

System.***out***.println("Cadastrado com sucesso!!\n");

} **else** {

System.***out***.println("Falha no Cadastro!.\n");

}

**break**;

}

**case** "3": {

System.***out***.println("Cadastrando...");

String nomeT;

String idade;

String cpf;

String telefone;

**do** {

System.***out***.print("Digite o Nome:");

nomeT = receba.nextLine();

nomeT = nomeT.replace(" ", " ");

}**while**(!g.verifNome(nomeT));

**do** {

System.***out***.print("Digite a Idade:");

idade = receba.nextLine();

}**while**(!g.verifIdade(idade));

**do** {

System.***out***.print("Digite o CPF:");

cpf = receba.nextLine();

}**while**(!g.verifCPF(cpf));

System.***out***.print("Diga o Plano de Saude:");

String planoSaude = receba.nextLine();

**do** {

System.***out***.print("Digite o Telefone:");

telefone = receba.nextLine();

}**while**(!g.verifTel(telefone));

System.***out***.print("Digite o Endereço:");

String endereço = receba.nextLine();

**double** despeza = Double.*parseDouble*("0");

String prescrição = "Nenhuma";

**if** (g.casdastrarClientes(caminho, nomeT, idade, cpf, planoSaude, telefone, endereço, despeza,

prescrição)) {

System.***out***.println("Cadastrado com sucesso!!\n");

} **else** {

System.***out***.println("Falha no Cadastro!.\n");

}

**break**;

}

**case** "4": {

Gerente gg = **new** Gerente();

System.***out***.println("Digite o arquivo que queira transferir:");

String origem = receba.nextLine();

System.***out***.println("Digite para onde o arquivo sera transferido caso não haja ira ser criado:");

String destino = receba.nextLine();

gg.transferirDados(caminho, origem, destino);

**break**;

}

**case** "5": {

Gerente gg = **new** Gerente();

gg.relatorioFuncionarios(caminho);

**break**;

}

**case** "6": {

Gerente gg = **new** Gerente();

gg.relatorioClientes(caminho);

**break**;

}

**case** "7": {

Gerente gg = **new** Gerente();

System.***out***.println("Digite o CPF do funcionario/a que deseja demitir:");

**int** cpf = Integer.*parseInt*(receba.nextLine());

**if** (gg.demitirFunc(caminho, cpf)) {

System.***out***.println("Demitido com Sucesso!\n");

} **else** {

System.***out***.println("Falha no procedimento\n");

}

**break**;

}

**case** "8": {

Gerente gg = **new** Gerente();

System.***out***.println("Digite o Cpf do Funcionario/a que deseja alterar o Horario");

String cpf = receba.nextLine();

System.***out***.println("Digite o novo Horario");

String hora = receba.nextLine();

**if** (gg.mudarHora(caminho, cpf, hora)) {

System.***out***.println("Mudado com Sucesso!!\n");

} **else** {

System.***out***.println("Falha no Procedimento!!\n");

}

**break**;

}

**case** "9": {

System.***out***.println("Digite o CPF do Medico/a:");

String cpf = receba.nextLine();

System.***out***.println("Digite o valor do Bonus");

**double** pix = Double.*parseDouble*(receba.nextLine());

Gerente gg = **new** Gerente();

**if** (gg.depositMed(caminho, cpf, pix)) {

System.***out***.println("Depositado com sucesso!\n");

} **else** {

System.***out***.println("Falha no Procedimento!\n");

}

**break**;

}

**case** "10": {

System.***out***.println("Digite o CPF da/o Assistente:");

String cpf = receba.nextLine();

System.***out***.println("Digite o valor do Bonus");

**double** pix = Double.*parseDouble*(receba.nextLine());

Gerente gg = **new** Gerente();

**if** (gg.depositAss(caminho, cpf, pix)) {

System.***out***.println("Depositado com sucesso!\n");

} **else** {

System.***out***.println("Falha no Procedimento!\n");

}

**break**;

}

**default**: {

System.***out***.println("opção invalida");

}

}

} **else** {

**if** (g.verifMedico(caminho, op)) {

Medicos m = **new** Medicos();

System.***out***.println("Bem vindo Doutor, oque deseja fazer?:");

System.***out***.println("1 - Cadastrar Cliente");

System.***out***.println("2 - Cadastrar Assistente");

System.***out***.println("3 - Prescrição medica e confirmar consulta");

String opMed = receba.nextLine();

**switch** (opMed) {

**case** "1": {

System.***out***.println("Cadastrando...");

String nomeT;

String idade;

String cpf;

String telefone;

**do** {

System.***out***.print("Digite o Nome:");

nomeT = receba.nextLine();

nomeT = nomeT.replace(" ", " ");

}**while**(!g.verifNome(nomeT));

**do** {

System.***out***.print("Digite a Idade:");

idade = receba.nextLine();

}**while**(!g.verifIdade(idade));

**do** {

System.***out***.print("Digite o CPF:");

cpf = receba.nextLine();

}**while**(!g.verifCPF(cpf));

System.***out***.print("Diga o Plano de Saude:");

String planoSaude = receba.nextLine();

**do** {

System.***out***.print("Digite o Telefone:");

telefone = receba.nextLine();

}**while**(!g.verifTel(telefone));

System.***out***.print("Digite o Endereço:");

String endereço = receba.nextLine();

**double** despeza = Double.*parseDouble*("0");

System.***out***.println("Digite a Prescrição Medica: , caso não haja ainda digite nenhuma:");

String prescrição = receba.nextLine();

**if** (g.casdastrarClientes(caminho, nomeT, idade, cpf, planoSaude, telefone, endereço,despeza,

prescrição)) {

System.***out***.println("Cadastrado com sucesso!!\n");

} **else** {

System.***out***.println("Falha no Cadastro!.\n");

}

**break**;

}

**case** "2": {

System.***out***.println("Cadastrando...");

String nomeT;

String idade;

String cpf;

String telefone;

**do** {

System.***out***.print("Digite o Nome:");

nomeT = receba.nextLine();

nomeT = nomeT.replace(" ", " ");

}**while**(!g.verifNome(nomeT));

**do** {

System.***out***.print("Digite a Idade:");

idade = receba.nextLine();

}**while**(!g.verifIdade(idade));

**do** {

System.***out***.print("Digite o CPF:");

cpf = receba.nextLine();

}**while**(!g.verifCPF(cpf));

System.***out***.print("Diga o Plano de Saude:");

String planoSaude = receba.nextLine();

**do** {

System.***out***.print("Digite o Telefone:");

telefone = receba.nextLine();

}**while**(!g.verifTel(telefone));

System.***out***.print("Digite o Endereço:");

String endereço = receba.nextLine();

**double** receita = Double.*parseDouble*("0");

System.***out***.print("Digite o Horario de Trabalho:");

String horario = receba.nextLine();

**if** (g.casdastrarAssistentes(caminho, nomeT, idade, cpf, planoSaude, telefone, endereço,

receita, horario)) {

System.***out***.println("Cadastrado com sucesso!!\n");

} **else** {

System.***out***.println("Falha no Cadastro!.\n");

}

**break**;

}

**case** "3": {

System.***out***.println("Digite o cpf do paciente : ");

String cpf = receba.nextLine();

System.***out***.println("Escreve a precrição para o paciente :");

String prescrição = receba.nextLine();

**if** (m.prescrição(caminho, cpf, prescrição)) {

System.***out***.println("Prescrição adicionada com sucesso!");

Consulta cn = **new** Consulta();

System.***out***.println("Paciente foi consultado ? escolha a opção");

System.***out***.println("1 - Sim");

System.***out***.println("2 - Não");

String opcn = receba.nextLine();

**if** (opcn.equalsIgnoreCase("1")) {

System.***out***.println("Valor da Consulta");

String valor = receba.nextLine();

**if** (cn.consultar(caminho, cpf, valor)) {

System.***out***.println("Consulta registrada com sucesso\n");

} **else** {

System.***out***.println("Falha no Procedimento\n");

}

}

} **else** {

System.***out***.println("Falha no Procedimento\n");

}

**break**;

}

**default**: {

System.***out***.println("opção invalida");

}

}

} **else** {

**if** (g.verifAssistente(caminho, op)) {

System.***out***.println("Bem vindo/a Assistente, oque deseja fazer?:");

Assistentes a = **new** Assistentes();

System.***out***.println("1 - Cadastrar Cliente");

System.***out***.println("2 - Marca consulta");

System.***out***.println("3 - Desmarca consulta");

System.***out***.println("4 - Entregar Produtos");

System.***out***.println("5 - Verificar se o Medico passou prescrição");

System.***out***.println("6 - Pagamento Cliente");

String opAss = receba.nextLine();

**switch** (opAss) {

**case** "1": {

System.***out***.println("Cadastrando...");

String nomeT;

String idade;

String cpf;

String telefone;

**do** {

System.***out***.print("Digite o Nome:");

nomeT = receba.nextLine();

nomeT = nomeT.replace(" ", " ");

}**while**(!g.verifNome(nomeT));

**do** {

System.***out***.print("Digite a Idade:");

idade = receba.nextLine();

}**while**(!g.verifIdade(idade));

**do** {

System.***out***.print("Digite o CPF:");

cpf = receba.nextLine();

}**while**(!g.verifCPF(cpf));

System.***out***.print("Diga o Plano de Saude:");

String planoSaude = receba.nextLine();

**do** {

System.***out***.print("Digite o Telefone:");

telefone = receba.nextLine();

}**while**(!g.verifTel(telefone));

System.***out***.print("Digite o Endereço:");

String endereço = receba.nextLine();

**double** despeza = Double.*parseDouble*("0");

System.***out***.println("Digite a Prescrição Medica: , caso não haja ainda digite nenhuma:");

String prescrição = receba.nextLine();

**if** (g.casdastrarClientes(caminho, nomeT, idade, cpf, planoSaude, telefone, endereço,

despeza, prescrição)) {

System.***out***.println("Cadastrado com sucesso!!\n");

} **else** {

System.***out***.println("Falha no Cadastro!.\n");

}

**break**;

}

**case** "2": {

System.***out***.println("Horarios dos Medicos :");

**if** (a.imprimaHorarios(caminho)) {

System.***out***.println("Digite 1 para agendar consulta ");

String opAg = receba.nextLine();

**if** (opAg.equalsIgnoreCase("1")) {

Agenda ag = **new** Agenda();

System.***out***.println("Digite o CPF do Cliente:");

String cpf = receba.nextLine();

**if** (ag.verAgendado(caminho, cpf)) {

System.***out***.println("Ja agendado:");

ag.agendamento(caminho, cpf);

} **else** {

System.***out***.println("Agendar : ");

System.***out***.println("Digite o cpf do Cliente");

String cpfCl = receba.nextLine();

System.***out***.println("Digite o Nome do Medico:");

String nomeMed = receba.nextLine();

System.***out***.println("Digite o CPF Medico:");

String cpfMed = receba.nextLine();

System.***out***.println("Digite o Horario para agendar:");

String hora = receba.nextLine();

**if** (ag.agendar(caminho, cpfCl, nomeMed, cpfMed, hora)) {

System.***out***.println("Agendamento Realizado!!\n");

} **else** {

System.***out***.println("Falha no Procedimento\n");

}

}

}

} **else** {

System.***out***.println("Falha no Procedimento\n");

}

**break**;

}

**case** "4": {

System.***out***.println("Digite o cpf do paciente para puxar o endereço");

String cpf = receba.nextLine();

a.imprimirEndereço(caminho, cpf);

**break**;

}

**case** "5": {

System.***out***.println("Digite o cpf do paciente para verificar a prescrição medica");

String cpf = receba.nextLine();

a.imprimirPrecrição(caminho, cpf);

**break**;

}

**case** "6": {

System.***out***.println("Pagando...");

Pagando cp = **new** Pagando();

System.***out***.println("Informe o CPF do Cliente");

String cpf = receba.nextLine();

**if** (cp.pixCP(caminho, cpf)) {

System.***out***.println("Operação concluida\n");

} **else** {

System.***out***.println("Falha no Procedimento\n");

}

}

}

**break**;

}

}

}

}

**else** {

**if** (g.verifcpfClie(caminho, op)) {

System.***out***.println("Login concluido");

System.***out***.println("Bem vindo oque deseja ? :");

System.***out***.println("Caso queira ver a prescrição passa pelo seu medico digite 1");

System.***out***.println("Caso queira marcar consulta fale com uma assistente.");

System.***out***.println("Caso queira verificar se a consulta foi paga digite 2");

String opClie = receba.nextLine();

**switch** (opClie) {

**case** "1": {

Clientes c = **new** Clientes();

c.verPrescrição(caminho, op);

**break**;

}

**case** "2": {

Clientes c = **new** Clientes();

c.checarConsulta(caminho, op);

**break**;

}

**default**: {

System.***out***.println("opção invalida\n");

}

}

} **else** {

**if** (!op.equalsIgnoreCase("S")) {

System.***out***.println("Não possui conta , Cadastre-se digitando 1:");

System.***out***.println("Caso não queira digite 2");

String opUser = receba.nextLine();

**switch**(opUser) {

**case**"1":{

System.***out***.println("Cadastrando...");

String nomeT;

String idade;

String cpf;

String telefone;

**do** {

System.***out***.print("Digite o Nome:");

nomeT = receba.nextLine();

nomeT = nomeT.replace(" ", " ");

}**while**(!g.verifNome(nomeT));

**do** {

System.***out***.print("Digite a Idade:");

idade = receba.nextLine();

}**while**(!g.verifIdade(idade));

**do** {

System.***out***.print("Digite o CPF:");

cpf = receba.nextLine();

}**while**(!g.verifCPF(cpf));

System.***out***.print("Diga o Plano de Saude:");

String planoSaude = receba.nextLine();

**do** {

System.***out***.print("Digite o Telefone:");

telefone = receba.nextLine();

}**while**(!g.verifTel(telefone));

System.***out***.print("Digite o Endereço:");

String endereço = receba.nextLine();

**double** despeza = Double.*parseDouble*("0");

String prescrição = "nenhuma";

**if** (g.casdastrarClientes(caminho, nomeT, idade, cpf, planoSaude, telefone, endereço, despeza,

prescrição)) {

System.***out***.println("Cadastrado com sucesso!!\n");

} **else** {

System.***out***.println("Falha no Cadastro!.\n");

}

**break**;

}

**case** "2" :{

System.***out***.println("Obrigado por nos visitar!\n");

**break**;

}

**default**:

System.***out***.println("Opção invalida\n");

}

}

}

}

} **while** (!op.equalsIgnoreCase("S"));

System.***out***.println("FIM...");

receba.close();

}

}

**package** backEnd01;

**import** java.io.BufferedReader;

**import** java.io.BufferedWriter;

**import** java.io.FileReader;

**import** java.io.FileWriter;

**public** **class** Funcionarios {

**protected** String nomeT;

**protected** String idade;

**protected** **int** cpf;

**protected** String planoSaude;

**protected** **int** telefone;

**protected** String endereço;

**protected** **double** receita;

**protected** String horario;

**public** Funcionarios() {

**super**();

}

**public** Funcionarios(String nomeT, String idade, **int** cpf, String planoSaude, **int** telefone, String endereço,

**double** receita, String horario) {

**super**();

**this**.nomeT = nomeT;

**this**.idade = idade;

**this**.cpf = cpf;

**this**.planoSaude = planoSaude;

**this**.telefone = telefone;

**this**.endereço = endereço;

**this**.receita = receita;

**this**.horario = horario;

}

**public** **boolean** verifcpfFunc(String caminho, String cpf) {

caminho = caminho + "Funcionarios.txt";

**try** (BufferedReader bcr = **new** BufferedReader(**new** FileReader(caminho))) {

**while** (bcr.ready()) {

String line = bcr.readLine();

String vet[] = line.split(":");

line = line.trim();

String login = vet[5];

**if** (login.equalsIgnoreCase(cpf)) {

**return** **true**;

}

}

bcr.close();

} **catch** (Exception e) {

System.***err***.println(e);

}

**return** **false**;

}

**public** **boolean** verifcpfClie(String caminho, String cpf) {

caminho = caminho + "Clientes.txt";

**try** (BufferedReader bcr = **new** BufferedReader(**new** FileReader(caminho))) {

**while** (bcr.ready()) {

String line = bcr.readLine();

String vet[] = line.split(":");

line = line.trim();

String login = vet[5];

**if** (login.equalsIgnoreCase(cpf)) {

**return** **true**;

}

}

bcr.close();

} **catch** (Exception e) {

System.***err***.println(e);

}

**return** **false**;

}

**public** **boolean** verifGerente(String caminho, String cpf) {

caminho = caminho + "Funcionarios.txt";

**try** (BufferedReader bcr = **new** BufferedReader(**new** FileReader(caminho))) {

**while** (bcr.ready()) {

String line = bcr.readLine();

String vet[] = line.split(":");

String login = vet[5];

**if** (login.equalsIgnoreCase(cpf)) {

**if** (vet[15].equalsIgnoreCase("Gerente")) {

**return** **true**;

}

}

}

bcr.close();

} **catch** (Exception e) {

System.***err***.println(e);

}

**return** **false**;

}

**public** **boolean** verifMedico(String caminho, String cpf) {

caminho = caminho + "Funcionarios.txt";

**try** (BufferedReader bcr = **new** BufferedReader(**new** FileReader(caminho))) {

**while** (bcr.ready()) {

String line = bcr.readLine();

String vet[] = line.split(":");

**if** (vet[5].equalsIgnoreCase(cpf)) {

**if** (vet[15].equalsIgnoreCase("Medico")) {

**return** **true**;

}

}

}

bcr.close();

} **catch** (Exception e) {

System.***err***.println(e);

}

**return** **false**;

}

**public** **boolean** verifAssistente(String caminho, String cpf) {

caminho = caminho + "Funcionarios.txt";

**try** (BufferedReader bcr = **new** BufferedReader(**new** FileReader(caminho))) {

**while** (bcr.ready()) {

String line = bcr.readLine();

String vet[] = line.split(":");

**if** (vet[5].equalsIgnoreCase(cpf)) {

**if** (vet[15].equalsIgnoreCase("Assistente")) {

**return** **true**;

}

}

}

bcr.close();

} **catch** (Exception e) {

System.***err***.println(e);

}

**return** **false**;

}

**public** **boolean** casdastrarGerente(String caminho, String nomeT, String idade, String cpf, String planoSaude,

String telefone, String endereço, **double** receita, String horario) {

**try** {

caminho = caminho + "Funcionarios.txt";

BufferedWriter bccw = **new** BufferedWriter(**new** FileWriter(caminho, **true**));

bccw.write("Nome:" + nomeT + ":Idade:" + idade + ":CPF:" + cpf + ":Plano:" + planoSaude + ":TEL:" + telefone

+ ":Endereço:" + endereço + ":R$:" + receita + ":Prof:Gerente" + ":HR:" + horario);

bccw.newLine();

bccw.close();

**return** **true**;

} **catch** (Exception e) {

System.***err***.println(e.toString());

}

**return** **false**;

}

**public** **boolean** casdastrarMedicos(String caminho, String nomeT, String idade, String cpf, String planoSaude,

String telefone, String endereço, **double** receita, String horario) {

**try** {

caminho = caminho + "Funcionarios.txt";

BufferedWriter bccw = **new** BufferedWriter(**new** FileWriter(caminho, **true**));

bccw.write("Nome:" + nomeT + ":Idade:" + idade + ":CPF:" + cpf + ":Plano:" + planoSaude + ":TEL:" + telefone

+ ":Endereço:" + endereço + ":R$:" + receita + ":Prof:Medico" + ":HR:" + horario);

bccw.newLine();

bccw.close();

**return** **true**;

} **catch** (Exception e) {

System.***err***.println(e.toString());

}

**return** **false**;

}

**public** **boolean** casdastrarAssistentes(String caminho, String nomeT, String idade, String cpf, String planoSaude,

String telefone, String endereço, **double** receita, String horario) {

**try** {

caminho = caminho + "Funcionarios.txt";

BufferedWriter bccw = **new** BufferedWriter(**new** FileWriter(caminho, **true**));

bccw.write("Nome:" + nomeT + ":Idade:" + idade + ":CPF:" + cpf + ":Plano:" + planoSaude + ":TEL:" + telefone

+ ":Endereço:" + endereço + ":R$:" + receita + ":Prof:Assistente" + ":HR:" + horario);

bccw.newLine();

bccw.close();

**return** **true**;

} **catch** (Exception e) {

System.***err***.println(e.toString());

}

**return** **false**;

}

**public** **boolean** casdastrarClientes(String caminho, String nomeT, String idade, String cpf, String planoSaude,

String telefone, String endereço, **double** despeza, String prescrição) {

**try** {

caminho = caminho + "Clientes.txt";

BufferedWriter bccw = **new** BufferedWriter(**new** FileWriter(caminho, **true**));

bccw.write("Nome:" + nomeT + ":Idade:" + idade + ":CPF:" + cpf + ":Plano:" + planoSaude + ":TEL:" + telefone

+ ":Endereço:" + endereço + ":R$:" + despeza + ":Prescrição medica :" + prescrição + ":Cliente");

bccw.newLine();

bccw.close();

**return** **true**;

} **catch** (Exception e) {

System.***err***.println(e.toString());

}

**return** **false**;

}

**public** **boolean** verifLetra(String verifL) {

**if** (verifL.contains("a") || verifL.contains("b") || verifL.contains("c") || verifL.contains("d")

|| verifL.contains("e") || verifL.contains("f") || verifL.contains("g") || verifL.contains("h")

|| verifL.contains("i") || verifL.contains("j") || verifL.contains("k") || verifL.contains("l")

|| verifL.contains("m") || verifL.contains("n") || verifL.contains("o") || verifL.contains("p")

|| verifL.contains("q") || verifL.contains("r") || verifL.contains("s") || verifL.contains("t")

|| verifL.contains("u") || verifL.contains("v") || verifL.contains("w") || verifL.contains("x")

|| verifL.contains("y") || verifL.contains("z")) {

**return** **true**;

}

**return** **false**;

}

**public** **boolean** verifNome(String nome) {

**if** (nome.contains("@") || nome.contains("-") || nome.contains("0") || nome.contains("1") || nome.contains("2")

|| nome.contains("3") || nome.contains("4") || nome.contains("5") || nome.contains("6")

|| nome.contains("7") || nome.contains("8") || nome.contains("9")) {

**return** **false**;

} **else** {

**if** (nome.length() >= 4) {

**return** **true**;

}

}

**return** **false**;

}

**public** **boolean** verifIdade(String idade) {

**if** (idade.length() <= 3) {

**if** (verifLetra(idade)) {

**return** **false**;

} **else** {

**if** (idade.contains("0") || idade.contains("1") || idade.contains("2") || idade.contains("3")

|| idade.contains("4") || idade.contains("5") || idade.contains("6") || idade.contains("7")

|| idade.contains("8") || idade.contains("9")) {

**return** **true**;

}

}

}

**return** **false**;

}

**public** **boolean** verifCPF(String cpf) {

**if** (verifLetra(cpf)) {

**return** **false**;

} **else** {

**if** (cpf.contains("0") || cpf.contains("1") || cpf.contains("2") || cpf.contains("3") || cpf.contains("4")

|| cpf.contains("5") || cpf.contains("6") || cpf.contains("7") || cpf.contains("8")

|| cpf.contains("9") || cpf.contains(".") || cpf.contains("-")) {

**if** (cpf.length() == 11) {

**return** **true**;

} **else** {

**if** (cpf.length() == 14 && cpf.charAt(3) == '.' && cpf.charAt(7) == '.' && cpf.charAt(11) == '-') {

**return** **true**;

}

}

}

}

**return** **false**;

}

**public** **boolean** verifTel(String tel) {

**if** (verifLetra(tel)) {

**return** **false**;

} **else** {

**if** (tel.contains("0") || tel.contains("1") || tel.contains("2") || tel.contains("3") || tel.contains("4")

|| tel.contains("5") || tel.contains("6") || tel.contains("7") || tel.contains("8")

|| tel.contains("9") || tel.contains("(") || tel.contains(")") || tel.contains("-")) {

**if** (tel.length() == 11) {

**return** **true**;

} **else** {

**if** (tel.length() == 14 && tel.charAt(0) == '(' && tel.charAt(3) == ')' && tel.charAt(9) == '-') {

**return** **true**;

}

}

}

}

**return** **false**;

}

**protected** String getNomeT() {

**return** nomeT;

}

**protected** **void** setNomeT(String nomeT) {

**this**.nomeT = nomeT;

}

**protected** String getIdade() {

**return** idade;

}

**protected** **void** setIdade(String idade) {

**this**.idade = idade;

}

**protected** **int** getCpf() {

**return** cpf;

}

**protected** **void** setCpf(**int** cpf) {

**this**.cpf = cpf;

}

**protected** String getPlanoSaude() {

**return** planoSaude;

}

**protected** **void** setPlanoSaude(String planoSaude) {

**this**.planoSaude = planoSaude;

}

**protected** **int** getTelefone() {

**return** telefone;

}

**protected** **void** setTelefone(**int** telefone) {

**this**.telefone = telefone;

}

**protected** String getEndereço() {

**return** endereço;

}

**protected** **void** setEndereço(String endereço) {

**this**.endereço = endereço;

}

**protected** **double** getReceita() {

**return** receita;

}

**protected** **void** setReceita(**double** receita) {

**this**.receita = receita;

}

**protected** String getHorario() {

**return** horario;

}

**protected** **void** setHorario(String horario) {

**this**.horario = horario;

}

}

**package** backEnd01;

**import** java.io.BufferedReader;

**import** java.io.BufferedWriter;

**import** java.io.FileReader;

**import** java.io.FileWriter;

**public** **class** Gerente **extends** Funcionarios{

**public** Gerente() {

**super**();

}

**public** **void** transferirDados(String caminho,String origem,String destino) {

origem = caminho + origem+".txt";

**try** {

FileReader fccr1 = **new** FileReader(origem);

BufferedReader bccr1 = **new** BufferedReader(fccr1);

FileWriter fccw1 = **new** FileWriter(destino);

BufferedWriter bccw1 = **new** BufferedWriter(fccw1);

**while** (bccr1.ready()) {

String linee = bccr1.readLine();

bccw1.write(linee);

bccw1.newLine();

}

bccw1.close();

fccw1.close();

bccr1.close();

fccr1.close();

} **catch** (Exception e) {

System.***out***.println(e);

}

}

**public** **void** relatorioFuncionarios(String caminho) {

caminho = caminho + "Funcionarios.txt";

**try** {

BufferedReader bccr = **new** BufferedReader(**new** FileReader(caminho));

String line;

**while** (bccr.ready()) {

line = bccr.readLine();

String vet[] = line.split(":");

line.trim();

System.***out***.printf("Nome:%s,Idade:%s,CPF:%s,Plano:%s,Tel:%s,Endereço:%s,"

+"Receita:%s,Profissão:%s,Hora:%s %n",vet[1],vet[3],vet[5],vet[7],vet[9],vet[11],vet[13],vet[15],vet[17]+":"+vet[18]);

}

bccr.close();

System.***out***.println("\nFim do relatorio.");

} **catch** (Exception e) {

System.***err***.println(e);

}

}

**public** **void** relatorioClientes(String caminho) {

caminho = caminho + "Clientes.txt";

**try** {

BufferedReader bccr = **new** BufferedReader(**new** FileReader(caminho));

String line;

**while** (bccr.ready()) {

line = bccr.readLine();

String vet[] = line.split(":");

System.***out***.printf("Nome:%s,Idade:%s,CPF:%s,Plano:%s,Tel:%s,Endereço:%s,"

+"Despeza:%s,Prescrição:%s %n",vet[1],vet[3],vet[5],vet[7],vet[9],vet[11],vet[13],vet[15]);

}

System.***out***.println("\nFim do relatorio.\n");

bccr.close();

} **catch** (Exception e) {

System.***out***.println(e);

}

}

**public** **boolean** demitirFunc(String caminho,**int** cpf) {

caminho = caminho + "Funcionarios.txt";

**try** {

FileReader fcr = **new** FileReader(caminho);

BufferedReader bcr = **new** BufferedReader(fcr);

String temp = caminho.replace(".txt", "temp.txt");

FileWriter fcw = **new** FileWriter(temp);

BufferedWriter bcw = **new** BufferedWriter(fcw);

**int** cont = 0;

**while** (bcr.ready()) {

String line = bcr.readLine();

String vet[] = line.split(":");

**int** loc = Integer.*parseInt*(vet[5]);

**if** (loc == cpf) {

cont++;

} **else** {

bcw.write(line);

bcw.newLine();

}

}

bcw.close();

fcw.close();

bcr.close();

fcr.close();

**if** (cont == 1) {

transferir(temp, caminho);

**return** **true**;

}

} **catch** (Exception e) {

System.***err***.println(e);

}

**return** **false**;

}

**public** **boolean** mudarHora(String caminho,String cpf,String hora) {

caminho = caminho + "Funcionarios.txt";

**try** {

FileReader fcr = **new** FileReader(caminho);

BufferedReader bcr = **new** BufferedReader(fcr);

String temp = caminho.replace(".txt", "temp.txt");

FileWriter fcw = **new** FileWriter(temp);

BufferedWriter bcw = **new** BufferedWriter(fcw);

**int** cont = 0;

**while** (bcr.ready()) {

String line = bcr.readLine();

String vet[] = line.split(":");

**if** (vet[5].equalsIgnoreCase(cpf)) {

String horario = vet [17]+":"+vet[18];

horario = line.replace(horario, hora);

bcw.write(horario);

bcw.newLine();

cont++;

} **else** {

bcw.write(line);

bcw.newLine();

}

}

bcw.close();

fcw.close();

bcr.close();

fcr.close();

**if** (cont == 1) {

transferir(temp, caminho);

**return** **true**;

}

} **catch** (Exception e) {

System.***err***.println(e);

}

**return** **false**;

}

**public** **void** transferir(String origem, String caminho) {

**try** {

FileReader fccr1 = **new** FileReader(origem);

BufferedReader bccr1 = **new** BufferedReader(fccr1);

FileWriter fccw1 = **new** FileWriter(caminho);

BufferedWriter bccw1 = **new** BufferedWriter(fccw1);

**while** (bccr1.ready()) {

String linee = bccr1.readLine();

bccw1.write(linee);

bccw1.newLine();

}

bccw1.close();

fccw1.close();

bccr1.close();

fccr1.close();

} **catch** (Exception e) {

System.***out***.println(e);

}

}

**public** **boolean** depositMed(String caminho , String cpf , **double** pix) {

**try** {

String caminhoF = caminho + "Funcionarios.txt";

FileReader fcrf = **new** FileReader(caminhoF);

BufferedReader bcrf = **new** BufferedReader(fcrf);

String tempF = caminhoF.replace(".txt", "temp.txt");

FileWriter fcwf = **new** FileWriter(tempF);

BufferedWriter bcwf = **new** BufferedWriter(fcwf);

**int** cont = 0 ;

**while** (bcrf.ready()) {

String line = bcrf.readLine();

String vet[] = line.split(":");

**if** (vet[5].equalsIgnoreCase(cpf)) {

String nomeT = vet[1];

String idade = vet[3];

cpf = vet[5];

String planoSaude = vet[7];

String telefone = vet[9];

String endereço = vet[11];

**double** receita = Double.*parseDouble*(vet[13]);

String horario = vet[17]+":"+vet[18];

receita = receita + pix;

bcwf.write("Nome:"+nomeT+":Idade:" +idade+ ":CPF:" + cpf + ":Plano:" +planoSaude

+":TEL:" +telefone+ ":Endereço:"+endereço+":R$:"+receita+":Prof:Medico"+":HR:"+horario);

bcwf.newLine();

cont++;

} **else** {

bcwf.write(line);

bcwf.newLine();

}

}

bcwf.close();

fcwf.close();

bcrf.close();

fcrf.close();

**if**(cont != 0) {

transferir(tempF, caminhoF);

**return** **true**;

}

} **catch** (Exception e) {

System.***err***.println(e);

}

**return** **false**;

}

**public** **boolean** depositAss(String caminho , String cpf , **double** pix) {

**try** {

String caminhof = caminho + "Funcionarios.txt";

FileReader fcrf = **new** FileReader(caminhof);

BufferedReader bcrf = **new** BufferedReader(fcrf);

String tempF = caminhof.replace(".txt", "temp.txt");

FileWriter fcwf = **new** FileWriter(tempF);

BufferedWriter bcwf = **new** BufferedWriter(fcwf);

**int** cont = 0 ;

System.***out***.println(caminhof);

**while** (bcrf.ready()) {

String line = bcrf.readLine();

String vet[] = line.split(":");

**if** (vet[5].equalsIgnoreCase(cpf)) {

String nomeT = vet[1];

String idade = vet[3];

String planoSaude = vet[7];

String telefone = vet[9];

String endereço = vet[11];

**double** receita = Double.*parseDouble*(vet[13]);

String horario = vet[17]+":"+vet[18];

receita = receita + pix;

bcwf.write("Nome:"+nomeT+":Idade:" +idade+ ":CPF:" + cpf + ":Plano:" +planoSaude

+":TEL:" +telefone+ ":Endereço:"+endereço+":R$:"+receita+":Prof:Assistente"+":HR:"+horario);

bcwf.newLine();

cont++;

} **else** {

bcwf.write(line);

bcwf.newLine();

}

}

bcwf.close();

fcwf.close();

bcrf.close();

fcrf.close();

**if**(cont != 0) {

transferir(tempF, caminhof);

**return** **true**;

}

} **catch** (Exception e) {

System.***err***.println(e);

}

**return** **false**;

}

}

**package** backEnd01;

**import** java.io.BufferedReader;

**import** java.io.BufferedWriter;

**import** java.io.FileReader;

**import** java.io.FileWriter;

**public** **class** Medicos **extends** Funcionarios {

**private** String prescrição;

**public** Medicos() {

**super**();

}

**public** Medicos(String nomeT, String idade, **int** cpf, String planoSaude, **int** telefone, String endereço,

**double** saldo,String horario, String prescrição) {

**super**(nomeT, idade, cpf, planoSaude, telefone, endereço, saldo, horario);

**this**.prescrição = prescrição;

}

**public** **boolean** prescrição(String caminho, String paciente, String prescrição) {

caminho = caminho + "Clientes.txt";

**try** {

FileReader fcr = **new** FileReader(caminho);

BufferedReader bcr = **new** BufferedReader(fcr);

String temp = caminho.replace(".txt", "temp.txt");

FileWriter fcw = **new** FileWriter(temp);

BufferedWriter bcw = **new** BufferedWriter(fcw);

**int** cont = 0;

**while** (bcr.ready()) {

String line = bcr.readLine();

String vet[] = line.split(":");

**if** (vet[5].equalsIgnoreCase(paciente)) {

vet[15] = line.replace("nenhuma", prescrição);

bcw.write(vet[15]);

bcw.newLine();

cont++;

} **else** {

bcw.write(line);

bcw.newLine();

}

}

bcw.close();

fcw.close();

bcr.close();

fcr.close();

**if** (cont != 0) {

Gerente g = **new** Gerente();

g.transferir(temp, caminho);

**return** **true**;

}

} **catch** (Exception e) {

System.***err***.println(e);

}

**return** **false**;

}

**protected** String getPrescrição() {

**return** prescrição;

}

**protected** **void** setPrescrição(String prescrição) {

**this**.prescrição = prescrição;

}

}

**package** backEnd01;

**import** java.io.BufferedReader;

**import** java.io.FileReader;

**public** **class** Assistentes **extends** Funcionarios{

**public** **boolean** imprimaHorarios(String caminho) {

caminho = caminho + "Funcionarios.txt";

**try** {

BufferedReader bccr = **new** BufferedReader(**new** FileReader(caminho));

String line;

**int** cont = 0 ;

**while** (bccr.ready()) {

line = bccr.readLine();

String loc[] = line.split(":");

**if**(loc[15].equalsIgnoreCase("medico")) {

String vet[] = line.split(":");

System.***out***.printf("Nome:%s,Tel:%s,Hora:%s %n",vet[1],vet[9],vet[17]+":"+vet[18]);

cont++;

}

}

bccr.close();

**if**(cont >= 1) {

**return** **true**;

}

} **catch** (Exception e) {

System.***out***.println(e);

}

**return** **false**;

}

**public** **void** imprimirEndereço(String caminho , String paciente) {

caminho = caminho + "Clientes.txt";

**try** {

BufferedReader bccr = **new** BufferedReader(**new** FileReader(caminho));

String line;

**while**(bccr.ready()) {

line = bccr.readLine();

String loc [] = line.split(":");

**if**(loc[5].equalsIgnoreCase(paciente)) {

String vet[] = line.split(":");

System.***out***.println("Endereço: "+vet[11]);

}

}

bccr.close();

} **catch** (Exception e) {

System.***err***.println(e);

}

}

**public** **void** imprimirPrecrição(String caminho, String paciente) {

caminho = caminho + "Clientes.txt";

**try** {

BufferedReader bccr = **new** BufferedReader(**new** FileReader(caminho));

String line;

**while**(bccr.ready()) {

line = bccr.readLine();

String loc [] = line.split(":");

**if**(loc[5].equalsIgnoreCase(paciente)) {

String vet[] = line.split(",");

System.***out***.println("Prescrição Medica"+vet[15]);

}

}

bccr.close();

} **catch** (Exception e) {

System.***err***.println(e);

}

}

}

**package** backEnd01;

**import** java.io.BufferedReader;

**import** java.io.FileReader;

**public** **class** Clientes{

**public** **void** verPrescrição(String caminho, String paciente) {

caminho = caminho + "Clientes.txt";

**try** {

BufferedReader bccr = **new** BufferedReader(**new** FileReader(caminho));

String line;

**while** (bccr.ready()) {

line = bccr.readLine();

String loc[] = line.split(":");

**if** (loc[5].equalsIgnoreCase(paciente)) {

String vet[] = line.split(":");

System.***out***.println("Prescrição Medica: "+vet[15]);

}

}

bccr.close();

} **catch** (Exception e) {

System.***err***.println(e);

}

}

**public** **void** checarConsulta(String caminho , String cpf) {

caminho = caminho + "Consultas.txt";

**try** {

BufferedReader bccr = **new** BufferedReader(**new** FileReader(caminho));

String line;

**while** (bccr.ready()) {

line = bccr.readLine();

String loc[] = line.split(":");

**if** (loc[5].equalsIgnoreCase(cpf)) {

String vet[] = line.split(":");

System.***out***.println("Verificando... :"+vet[21]);

}

}

bccr.close();

} **catch** (Exception e) {

System.***err***.println(e);

}

}

}

**package** backEnd01;

**import** java.io.BufferedReader;

**import** java.io.BufferedWriter;

**import** java.io.FileReader;

**import** java.io.FileWriter;

**public** **class** Agenda{

**public** **boolean** agendar(String caminho ,String cpfCl ,String nomeMed, String cpfMed , String hora) {

**try** {

caminho = caminho + "Clientes.txt";

FileReader fcrf = **new** FileReader(caminho);

BufferedReader bcrf = **new** BufferedReader(fcrf);

String camAg = caminho.replace("Clientes.txt", "Agenda.txt");

FileWriter fcwf = **new** FileWriter(camAg,**true**);

BufferedWriter bcwf = **new** BufferedWriter(fcwf);

**int** cont = 0 ;

**while** (bcrf.ready()) {

String line = bcrf.readLine();

String vet[] = line.split(":");

String nomeT = vet[1];

String idade = vet[3];

String planoSaude = vet[7];

String telefone = vet[9];

String endereço = vet[11];

**if** (vet[5].equalsIgnoreCase(cpfCl)) {

bcwf.write("Nome:"+nomeT+":Idade:" +idade+ ":CPF:" + cpfCl + ":Plano:" +planoSaude

+":TEL:" +telefone+ ":Endereço:"+endereço+":Nome Medico:"+nomeMed+":CPF Medico:"+cpfMed+":HR:"+hora);

bcwf.newLine();

cont++;

}

}

bcwf.close();

fcwf.close();

bcrf.close();

fcrf.close();

**if**(cont != 0) {

**return** **true**;

}

}

**catch**(Exception e) {

System.***err***.println(e);

}

**return** **false**;

}

**public** **boolean** verAgendado(String caminho , String cpfCl) {

caminho = caminho + "Agenda.txt";

**try** {

FileReader fra = **new** FileReader(caminho);

BufferedReader bra = **new** BufferedReader(fra);

**int** cont = 0;

**while** (bra.ready()) {

String line = bra.readLine();

String vet[] = line.split(":");

**if**(vet[5].equalsIgnoreCase(cpfCl)) {

cont++;

}

}

bra.close();

fra.close();

**if**(cont != 0 ) {

**return** **true**;

}

} **catch** (Exception e) {

System.***err***.println(e);

}

**return** **false**;

}

**public** **void** agendamento(String caminho , String cpfCl) {

caminho = caminho + "Agenda.txt";

**try** {

FileReader fra = **new** FileReader(caminho);

BufferedReader bra = **new** BufferedReader(fra);

**while** (bra.ready()) {

String line = bra.readLine();

String vet[] = line.split(":");

**if**(vet[5].equalsIgnoreCase(cpfCl)) {

System.***out***.printf("Nome:%s,Idade:%s,CPF:%s,Plano:%s,Nome Medico:%s,Hora:%s:%s %n"

,vet[1],vet[3],vet[5],vet[7],vet[13],vet[17],vet[18]);

}

}

bra.close();

fra.close();

} **catch** (Exception e) {

System.***err***.println(e);

}

}

}

**package** backEnd01;

**import** java.io.BufferedReader;

**import** java.io.BufferedWriter;

**import** java.io.FileReader;

**import** java.io.FileWriter;

**public** **class** Consulta {

**public** **boolean** consultar(String caminho ,String cpfCl ,String valor) {

**try** {

caminho = caminho + "Agenda.txt";

FileReader fcrf = **new** FileReader(caminho);

BufferedReader bcrf = **new** BufferedReader(fcrf);

String camCons = caminho.replace("Agenda.txt", "Consultas.txt");

FileWriter fcwf = **new** FileWriter(camCons,**true**);

BufferedWriter bcwf = **new** BufferedWriter(fcwf);

**int** cont = 0 ;

**while** (bcrf.ready()) {

String line = bcrf.readLine();

String vet[] = line.split(":");

**if** (vet[5].equalsIgnoreCase(cpfCl)) {

String nomeT = vet[1];

String idade = vet[3];

String planoSaude = vet[7];

String telefone = vet[9];

String endereço = vet[11];

String nomeMed = vet[13];

String cpfMed = vet[15];

String hora = vet[17]+":"+vet[18];

bcwf.write("Nome:"+nomeT+":Idade:" +idade+ ":CPF:" + cpfCl + ":Plano:" +planoSaude

+":TEL:" +telefone+ ":Endereço:"+endereço+":Nome Medico:"+nomeMed+":CPF Medico:"+cpfMed+":HR:"+hora+":R$:"+valor+":Não Pago");

bcwf.newLine();

cont++;

}

}

bcwf.close();

fcwf.close();

bcrf.close();

fcrf.close();

**if**(cont != 0) {

**return** **true**;

}

}

**catch**(Exception e) {

System.***err***.println(e);

}

**return** **false**;

}

}

**package** backEnd01;

**import** java.io.BufferedReader;

**import** java.io.BufferedWriter;

**import** java.io.FileReader;

**import** java.io.FileWriter;

**public** **class** Pagando {

**public** Pagando() {

**super**();

}

**public** **void** pixconfirmCP(String caminho, String cpfCl, **double** pix) {

**try** {

String caminhoC = caminho + "Clientes.txt";

FileReader fcr = **new** FileReader(caminhoC);

BufferedReader bcr = **new** BufferedReader(fcr);

String tempC = caminhoC.replace(".txt", "temp.txt");

FileWriter fcw = **new** FileWriter(tempC);

BufferedWriter bcw = **new** BufferedWriter(fcw);

**while** (bcr.ready()) {

String line = bcr.readLine();

String vet[] = line.split(":");

**if** (vet[5].equalsIgnoreCase(cpfCl)) {

String nomeT = vet[1];

String idade = vet[3];

String planoSaude = vet[7];

String telefone = vet[9];

String endereço = vet[11];

**double** despeza = Double.*parseDouble*(vet[13]);

String prescrição = vet[15];

**if** (pix <= 0) {

System.***out***.println("Falha no pix");

} **else** {

despeza = despeza - pix;

bcw.write("Nome:" + nomeT + ":Idade:" + idade + ":CPF:" + cpfCl + ":Plano:" + planoSaude

+ ":TEL:" + telefone + ":Endereço:" + endereço + ":R$:" + despeza

+ ":Prescrição medica :" + prescrição + ":Cliente");

bcw.newLine();

}

} **else** {

bcw.write(line);

bcw.newLine();

}

}

bcw.close();

fcw.close();

bcr.close();

fcr.close();

Gerente g = **new** Gerente();

g.transferir(tempC, caminhoC);

} **catch** (Exception e) {

System.***err***.println(e);

}

}

**public** **boolean** pixCP(String caminho, String cpfCl) {

String caminhoCons = caminho + "Consultas.txt";

**try** {

FileReader fcr = **new** FileReader(caminhoCons);

BufferedReader bvc = **new** BufferedReader(fcr);

String tempCons = caminhoCons.replace(".txt", "temp.txt");

FileWriter fcw = **new** FileWriter(tempCons);

BufferedWriter bvw = **new** BufferedWriter(fcw);

**int** cont = 0;

**double** pix = 0;

**while** (bvc.ready()) {

String line = bvc.readLine();

String vet[] = line.split(":");

**if** (vet[5].equalsIgnoreCase(cpfCl)) {

String nomeC = vet[1];

String idadeC = vet[3];

String cpfC = vet[5];

String planoC = vet[7];

String tel = vet[9];

String endereçoC = vet[11];

String nomeMed = vet[13];

String cpfMed = vet[15];

String hora = vet[17] + ":" + vet[18];

pix = Double.*parseDouble*(vet[20]);

bvw.write("Nome:" + nomeC + ":Idade:" + idadeC + ":CPF:" + cpfC + ":Plano:" + planoC + ":TEL:"

+ tel + ":Endereço:" + endereçoC + ":Nome Medico:" + nomeMed + ":CPF Medico:" + cpfMed

+ ":HR:" + hora + ":R$:" + pix + ": Pago");

bvw.newLine();

Gerente g = **new** Gerente();

g.depositMed(caminho, cpfMed, pix);

pixconfirmCP(caminho, cpfCl, pix);

cont++;

} **else** {

bvw.write(line);

bvw.newLine();

}

}

bvc.close();

bvw.close();

**if** (cont != 0) {

Gerente g = **new** Gerente();

g.transferir(tempCons, caminhoCons);

**return** **true**;

}

} **catch** (Exception e) {

System.***err***.println(e);

}

**return** **false**;

}

}