

Curso: Técnico em Informática

Instrumento de
Avaliação:
Lista de Exercícios 2

Menção

Assinatura do Professor

Entrega: 10/10 até as 12h

Professores: Charles / Francis

Classe: 2i2

Aluno(s):

Nº.

Bases Tecnológicas: 1, 3 e 4.

Competências: 1.

Habilidades: 1.2

Instruções: A lista de exercícios poderá ser feita em dupla, mas implica no conhecimento da resolução por cada aluno da dupla. Os projetos deverão ser compactados e enviados no site Google Classroom (código da turma **37nh0yb**) até a data correspondente. Criar um arquivo em PDF para enviar junto identificando o curso, módulo, alunos, RM, conteúdo e professores.

O nome do projeto e do arquivo compactado deverá seguir a seguinte regra: **ListaX _AlunoA_B**, onde:
X – é o número da lista; A e B – nome(s) do(s) aluno(s).

Exemplo: Lista01_Mario_Wario

Cada exercício deverá ser criado em um pacote de nome **exercícioJ**, onde J é o número do exercício, e possuir, em cada exercício, **pelo menos** uma variável que contenha o RM do aluno.

Ex.: *String telefone*; → passar para *String telefone3340*;

Todas as Classes deverão ter o nome no padrão **ExercicioRMXXXX**, onde XXXX é o RM do aluno, se for em dupla alternar em cada atividade.

Para cada exercício abaixo crie: uma Tela usando JFrame, adicione os componentes necessários e os eventos conforme vistos em aula. Todos os campos devem ser protegidos com tamanho máximo e tipo próprio (números, monetário ou letras)

1 – Crie um programa para agendar consultas em uma clínica odontológica. O usuário deverá informar nome completo, data de nascimento (campo formatado), convênio (nome) e horário da consulta pretendida. O consultório atende apenas 10 pacientes por dia de uma em uma hora (das 08h as 18h). Não permitir o agendamento além da cota diária e horário preenchido. O agendamento sempre será referente ao próximo dia útil.

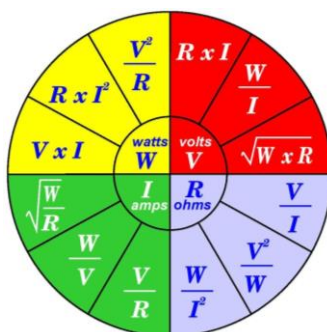
Os agendamentos poderão ser cancelados

Os pacientes agendados deverão ser exibidos em uma JList na parte inferior do programa.

O formulário **não** deverá possuir os botões minimizar, maximizar e fechar.

Programar um botão para sair com uma confirmação do usuário.

2 – Crie uma aplicação que demonstre os cálculos utilizados na lei de Ohm. O usuário fornecerá 2 valores quaisquer das grandezas elétricas (voltagem, amperagem, potência ou resistência) e a aplicação irá complementar as demais grandezas restantes. Utilize a ‘mandala’ abaixo como referência:



3 – Em um financiamento são inseridos: o valor da compra efetuada por um cliente, a quantidade de parcelas a ser paga com campos protegidos (apenas números e valores monetários). O programa deve calcular e exibir em uma JDialog o valor da prestação na cor vermelha em uma JLabel e em formato monetário com duas casas decimais.

4 – Para um comércio do ramo de materiais para construção é necessário o cadastro de produtos para as vendas. O programa deverá receber a data, código automático, descrição do produto, fracionamento (Kg, galão, litro, gramas, unidade, etc), quantidade e valor unitário. Ao clicar no botão OK exiba em uma JOptionPane: “Cadastrado com Sucesso! ”. O produto deverá ser armazenado em uma ArrayList parametrizado para o tipo Produto. Limpe todos os dados e posicione o foco do cursor na descrição do produto. Adicione um botão para busca de um produto pelo nome ou código (via JOptionPane.showInputDialog) e exiba o resultado em um JDialog.

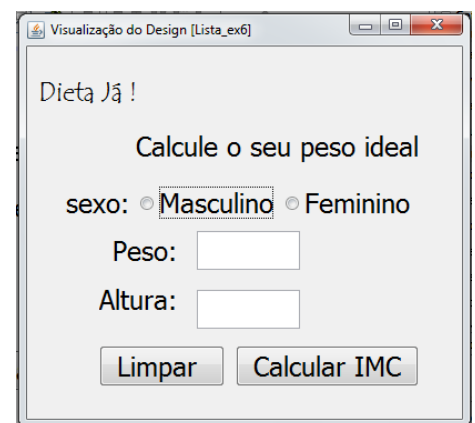
5 – A proprietária de uma loja de roupas femininas solicitou um programa que:

A – Receba a descrição de um produto vendido e seu valor

B – Exiba o preço do produto em três JPanels na mesma tela indicando as formas de pagamentos:

- À vista com 20% de desconto – cor de fundo azul
- 30 dias com 5% de desconto – cor de fundo vermelha
- Em até 4 pagamentos de: Exiba o valor da parcela sem juros – cor de fundo verde
- Baseado neste exercício, crie um botão para venda no exercício anterior que exiba em uma nova janela as mesmas opções deste programa quando o usuário fizer uma busca de algum produto.

6 – Faça o JFrame conforme o modelo e programe as funções dos botões. Exibir o resultado em uma JDialog.



7- Crie um JFrame conforme o modelo. Coloque as seguintes opções nas JComboBox:

Setor: Financeiro, RH, Administração, TI e Vendas

Cargo: Gerente, Office-Boy, Secretária, Diretor, Analista, Quebra-galho.

Quando clicar em adicionar concatenar todos os valores e adicionar a um ArrayList parametrizado do tipo Funcionario. Adicionar um botão para exibir todos os registros do ArrayList na JList. Quando o usuário clicar no botão Pesquisa deve ser feita uma busca pelo nome do funcionário os campos devem ser preenchidos com os valores encontrados, caso contrário exibir a mensagem “Funcionário não cadastrado”.

