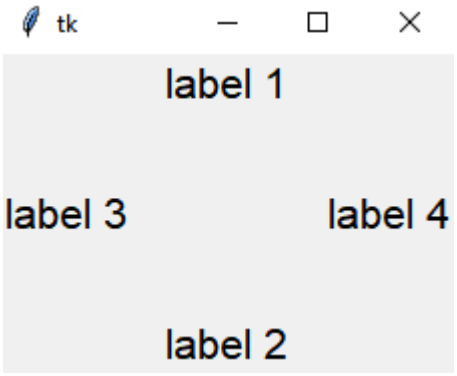
Exercício 1 - Posicionamento de Labels

Crie, utilizando o widget **Label** e o método de posicionamento **pack** e seus argumentos, a seguinte interface:

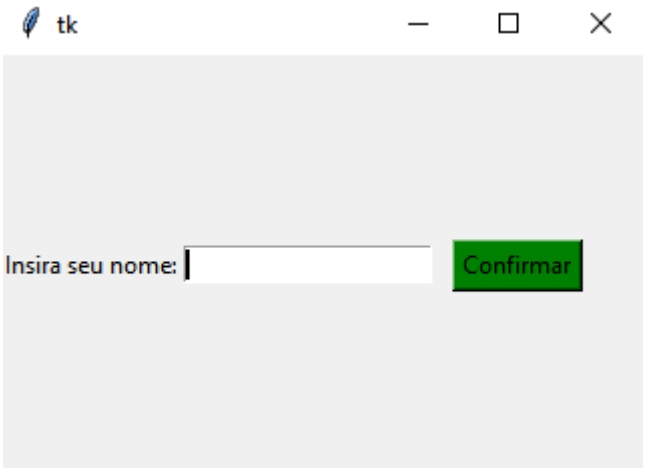


Exercício 2 - Soma de inteiros

Crie uma interface com dois campos de entrada e um Label. Os dois campos servirão para a entrada de dois inteiros, um em cada campo, e o Label servirá para aplicar o evento de clique com o botão esquerdo do mouse, a fim de somar os dois inteiros das entradas e printar no console o resultado.

Exercício 3 - Mensagem de boas vindas

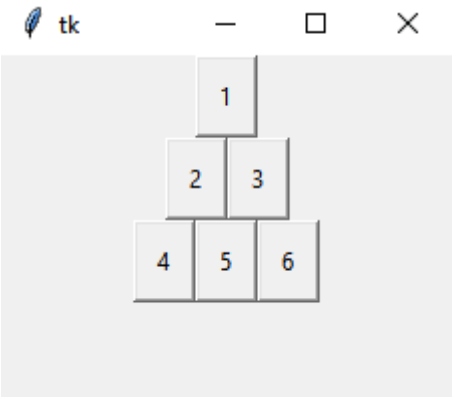
Reproduza a seguinte interface:



Descrição: O botão “confirmar” deverá acionar um método (utilizando o command) que adquira o nome inserido no campo de entrada e o imprima no console, com a mensagem ‘olá’ + nome_Inserido.

Exercício 4 - Pirâmide

Reproduza a seguinte interface:

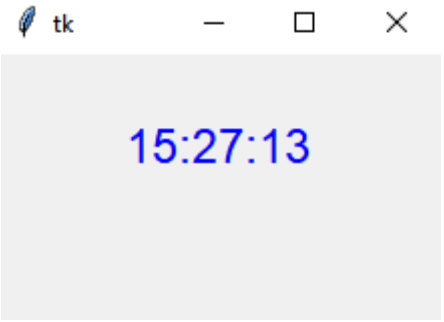


Exercício 5 - Conversor de tempo (Prática geral)

Crie um programa utilizando Tkinter que, dado um tempo em segundos, converta este tempo em horas, minutos e segundos.

Exercício 6 - Relógio digital (Prática geral)

Crie um relógio digital que atualize a cada um segundo e exiba a hora, minuto e segundo, conforme o exemplo:



Dica: Utilize o módulo datetime (a documentação é encontrada em: Datetime, e utilize também o método after do tkinter.

Exercício 7 - Linguagem de preferência

Crie 3 Radiobuttons, um com o texto “Python”, outro com o texto “Java” e o último com o texto “C++”. Além disso, crie um botão abaixo dos Radiobuttons com o texto “confirmar”, e por último um label sem nenhum texto. Quando o botão for clicado, deverá ser avaliado

qual a escolha selecionada nos Radiobuttons pelo usuário, para cada linguagem, deverá ser exibida no label a sintaxe de “hello world” de cada um, a saber:

Linguagem	Sintaxe
Python	print(“hello world”)
Java	System.out.println(“Hello world”)
C++	std::cout << “Hello world” << std::endl

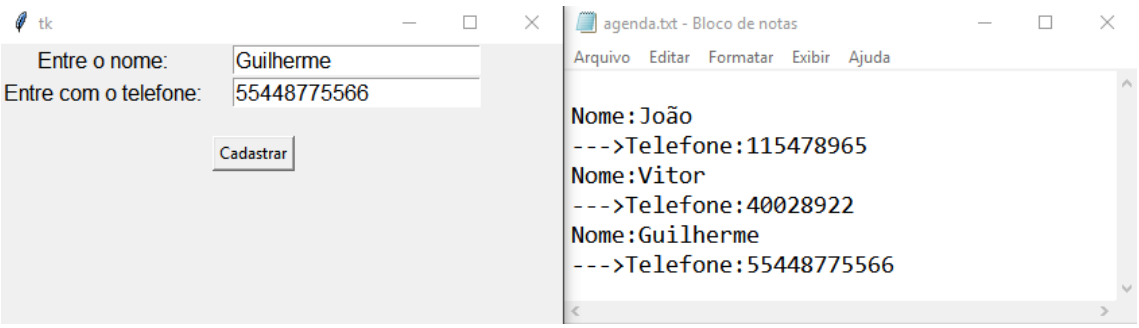
Dica: Para definir um texto no Label, lembre-se da notação de dicionários que pode ser utilizada para a alteração de argumentos dos widgets.

Exercício 8 - Leitura de arquivos com FileDialog.

Crie um programa contendo um Menu que invoque um fileDialog que abra um arquivo e printe seu conteúdo no console.

Exercício 9 - Agenda telefônica

Crie um programa para uma simples lista telefônica. Seu objetivo é possibilitar ao usuário a inserção de duas informações: o nome e o telefone do novo contato. Em seguida, ao clicar no botão “cadastrar” será ativo um FileDialog que possibilite a criação de um novo arquivo para o salvamento dos dados. Além disso, verifique a cada nova inserção se o usuário já possui um arquivo para salvamento de dados criado, evitando que a cada novo contato seja aberto um novo fileDialog, através de uma variável de controle. Segue o exemplo:



Dica: Tente utilizar o caminho até o novo arquivo para o salvamento de informações e controle, e não o objeto do arquivo. Isso pode te ajudar a controlar se existe ou não um arquivo para o salvamento de dados.

Exercício 10 - Leitura de arquivos e utilização de áreas de texto.

Dado o arquivo dados.txt (disponibilizado no GitHub) escreva todas as informações que o compõe em uma área de texto.

Exercício 11 - Manipulação de imagens.

Através do módulo Pillow, crie uma interface contendo um menu, com o item “Abrir imagem” que invoque um filedialog, para que o usuário escolha uma imagem. Caso o usuário não escolha nenhum arquivo, mostrar uma caixa de aviso. Se for aberto uma imagem, utilizar um label para exibi-la na janela.