**Manual de instrução de interface gráfica de usuário (GUI) no PyCharm**

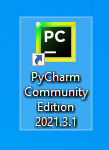
Demonstraremos a seguir os passos que serão necessários para a utilização de interface gráfica no PyCharm:

1. **Instalação do Customtkinter**

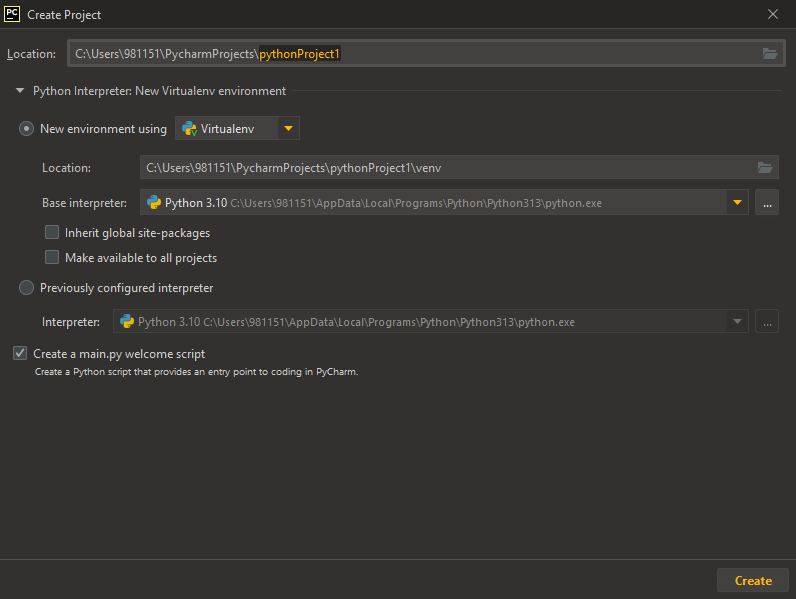
O customtkinter é uma ferramenta de usabilidade que permite o usuário criar e interagir com interface gráfica de projetos no pycharm.

Instalar Customtkinter:

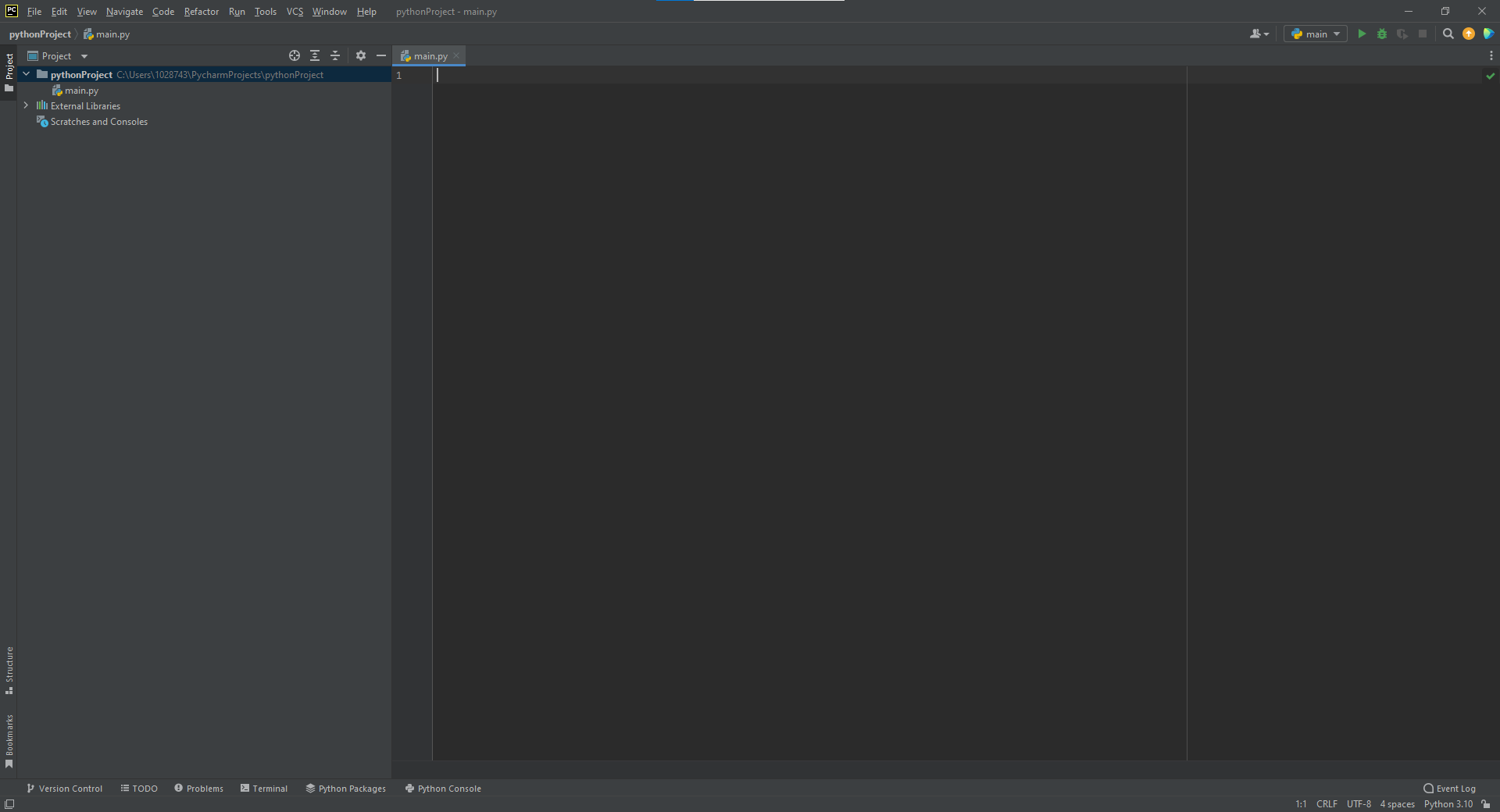
Abra o pycharm:



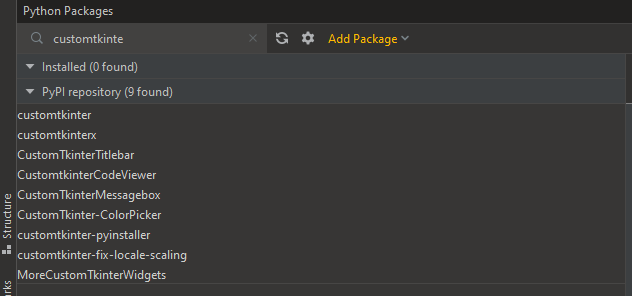
Crie um projeto e alterando na seleção com o nome do projeto que quiser:



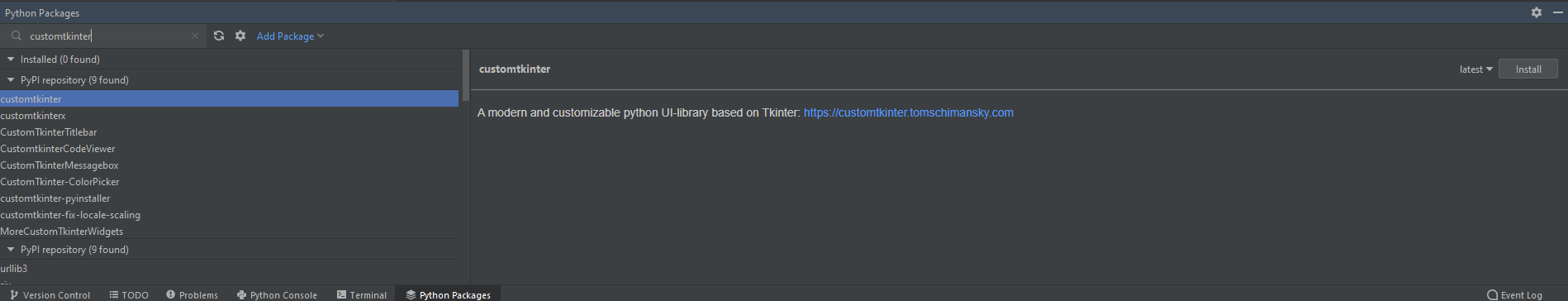
Vá em pacotes do Python no canto inferior direito:



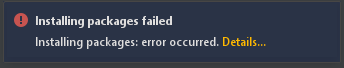
Procure por “customtkinter” na barra de pesquisa:



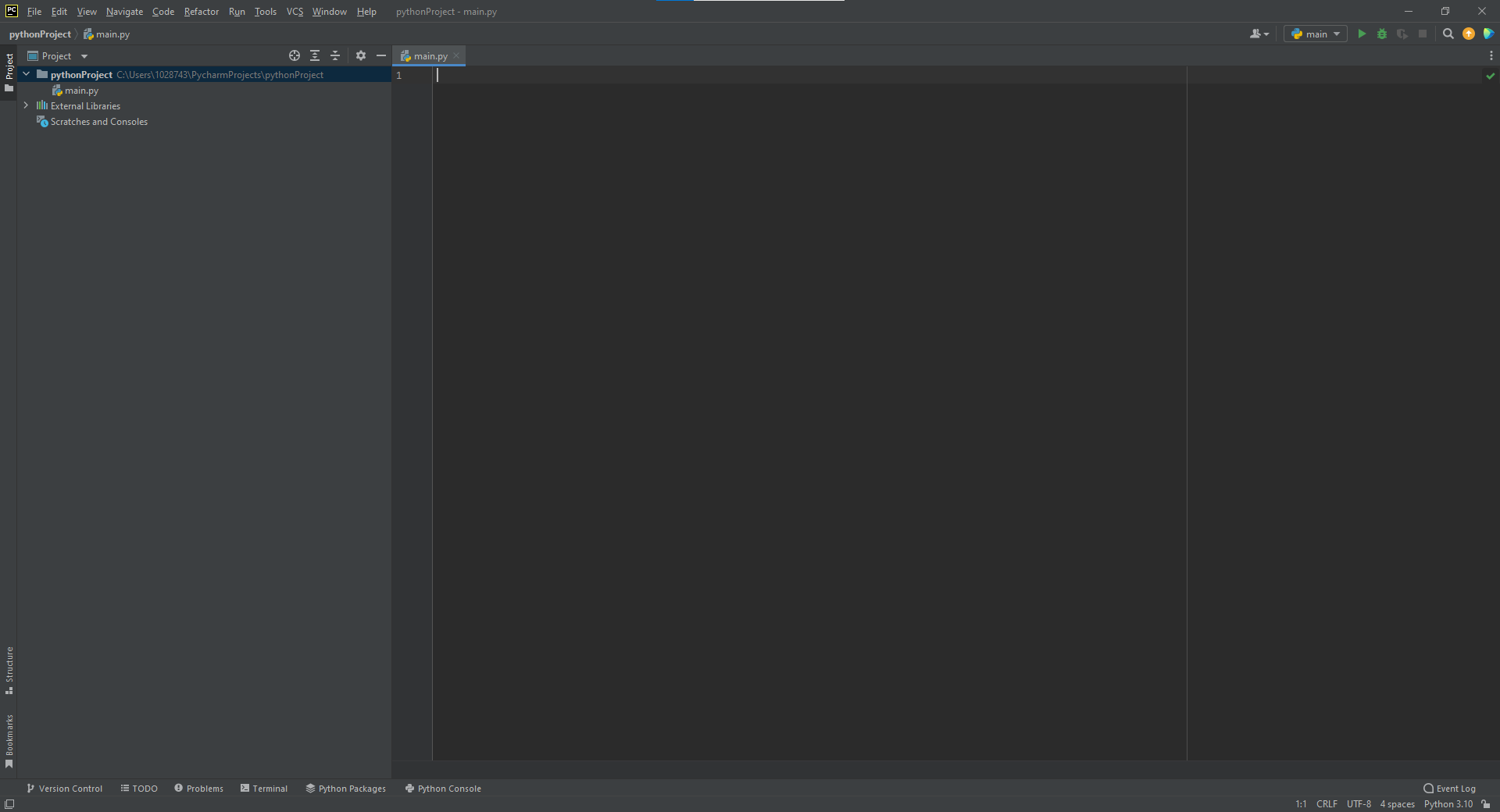
Execute a instalação:



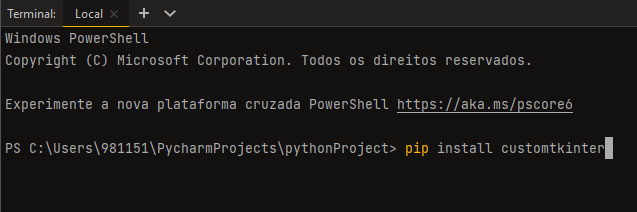
Em caso de erro, como esse:



Vá em terminal, que fica no mesmo canto que os pacotes:



E digite “pip install customtkinter” e aperte enter:



Agora o Customtkinter está instalado!

OS.: Todo novo projeto deve seguir estas etapas de instalação

Verifique a versão python instalada entrando no prompt de comando e chame o python com sua versão conforme a sintaxe abaixo

C:\Users\forte> python --version

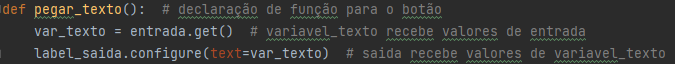
Python 3.12.6

Após a instalação do pacote, o usuário deve inserir o pacote em seu arquivo de projeto atual, utilizando o comando “import customtkinter” já na primeira linha do projeto:

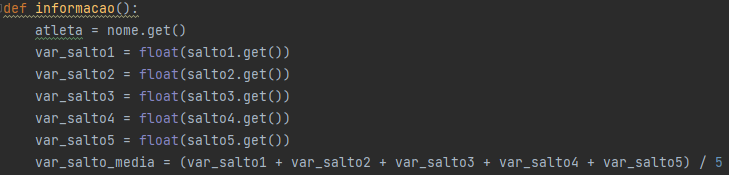
1. 

**Declaração de funções**

Todas as linhas pertinentes a função “**def**” estão aninhadas dentro da estrutura de declaração de variáveis:

A função pode ser designada para fazer, desde contas complexas, ou até mesmo exibir uma biblioteca inteira, mas aqui demonstraremos apenas o básico:

* É possível pegar valores dentro de variáveis de entrada usando o **“.get()”** (também é possível adulterar o tipo da variável se for compatível);
* Configurar variáveis do tipo Label usando o **“.configure()”;**



**Criação de interface**

Agora para criar uma interface, você pode interpretá-la como uma variável, dando um nome e classificando como uma janela:



Para renomear o titulo da interface você pode usar o comando “.title”:



Para definir o tamanho da janela, use o comando “.**geometry**” usando pixels como unidade de medida (sendo: “largura x altura”):



Existem algumas funções que mudam elementos na interface, como tema e cores, que se aplicam para todo o código, são esses o “.**set**\_**appearance**\_**mode**” (que define o tema da interface) e o **“.set\_default\_color\_theme**” (que muda a cor de alguns elementos, tais como: as cores do botões na janela. Obs: As únicas cores disponíveis são verde e azul



Para exibir a janela você usa o código “.**mainloop**”, sem isso a janela não será exibida (recomendado deixar essa linha no final do código, já que qualquer interação feita com a janela após essa função não será aplicada), ela deve ficar situada na última linha do projeto:



**Criação de texto**

Para criar um texto dentro de uma interface, é necessário definir uma variável do tipo “**CTklabel**” para armazenar os caracteres, e definir em qual janela será armazenada:



Também é possível adulterar família, tamanho, formatação e cor da fonte usando o esquema a seguir:



A parte mais importante após criar q ualquer objeto em uma interface é posicionar o mesmo, para isso você pode utilizar até 3 códigos diferentes, mas para este manual, usaremos apenas o “.**pack**”:



**O comando “.pack” empilha objetos um em cima do outro dentro da janela, e pode ser utilizado para qualquer objeto. Dentro do “.pack”, é possível definir o espaçamento ente objetos nos dois eixos usando o comando “pad[x ou y]=” baseado em pixels:**



É viável usar o Label para exibir a saída de um programa configurando seu texto com uma função conforme se segue:





**Criação de botões**

Para criar um botão na janela é necessário utilizar o tipo de variável “**CTkButton**”. É possível exibir um texto dentro do botão:



Dentro de um botão é possível definir uma função com o “**command**=” para dar uma utilidade ao botão (a função não pode ter parênteses) :



**Criação de caixas de entrada**

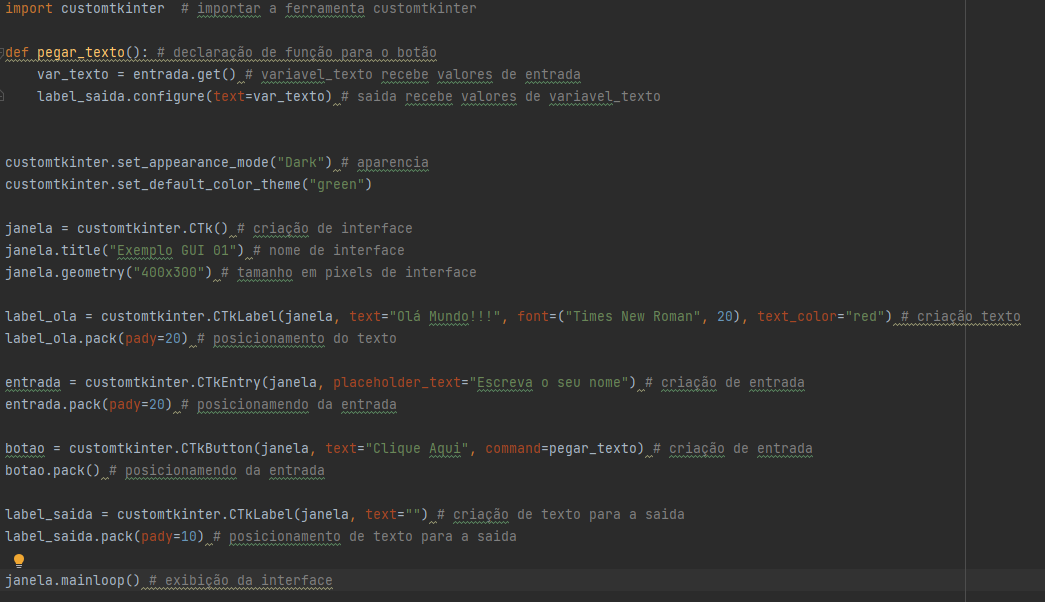
Para criar uma caixa onde o usuário possa interagir com uma entrada se utiliza o comando “**CTkEntry**”. (nota: é necessário, assim como o texto, posicionar a caixa na janela, utilizando o “.**pack**”):



Também é possível colocar um texto de instrução para o usuário saber o que escrever dentro da entrada utilizando o “**placeholder**\_**text=**”



Este é um código funcional que pega o texto inserido pelo usuário e o exibe abaixo do botão





Entendendo o “Forget”(Esquecer): o “FRAME1 – ATIVO” ou seja o “MENU” é o primeiro quadro do sistema e é fixo.  
O “FRAME 2 – ATIVO” é o quadro que contém as mesma dimensões e demais propriedades iguais como por exemplo: row, column, padx, pady, stickie. Enquanto isso O “FRAME 3 até o FRAME 6” ficam inativos(esquecidos, dormindo) esperando novas associações com os botões do MENU.

Diagrama

O conteúdo gerado por IA pode estar incorreto.

Ao utilizarmos essa função “**def abrir\_frame2()”** ela abre um “Frame” (quadro) de acordo com a que foi declarada. No nosso exemplo o “**Frame** 2”, enquanto os demais ficam dormindo, na seguinte sintaxe:

Texto

O conteúdo gerado por IA pode estar incorreto.

Fechando os FRAMES:

Texto

O conteúdo gerado por IA pode estar incorreto.

Para aplainar as nossas idéias sigamos a imagem abaixo:

O método .grid\_propagate(False) no Tkinter impede que um Frame ajuste automaticamente o seu tamanho com base no conteúdo interno. Isso significa que, mesmo que os widgets dentro dele sejam maiores ou menores, o Frame manterá o tamanho especificado.

O frame2 abre com duas funções “**propagate e o grid de edição**”:

Tela preta com letras brancas

O conteúdo gerado por IA pode estar incorreto.

O “row” é a linha, ”column” é a coluna, padx é a distância na horizontal e o pady é na vertical

System de Gerenciamento – Modo apresentação

FRAME 2 - ATIVO (CADASTRO)

FRAME 1- – ATIVO (MENU)

Tela de celular com aplicativo aberto

O conteúdo gerado por IA pode estar incorreto.

Obs: O FRAME 2 será substituído pelos demais frames de acordo com integração interativa de cada botão do menu. Ex: pressionando o botão “ Editar” exibe o Frame com o Texto “Editar produto:”

Tela de celular

O conteúdo gerado por IA pode estar incorreto.

01 – Definindo a aparência do System no caso “Dark - Escuro” e logo abaixo o tema da cor dos botões (“blue – Azul”)

Tela de celular

O conteúdo gerado por IA pode estar incorreto.

Trocando as cores do modo e do tema

Tela preta com letras brancas

O conteúdo gerado por IA pode estar incorreto.

Na prática:

Interface gráfica do usuário, Texto, Aplicativo, chat ou mensagem de texto

O conteúdo gerado por IA pode estar incorreto.

Designando o “**janela.title** – título: Sistema de Gerenciamento” da janela e editando o seu tamanho respectivo “janela.geometry(“800x400”) em pixel:

Interface gráfica do usuário, Texto

O conteúdo gerado por IA pode estar incorreto.

Titulo: Sistema de Gerenciamento

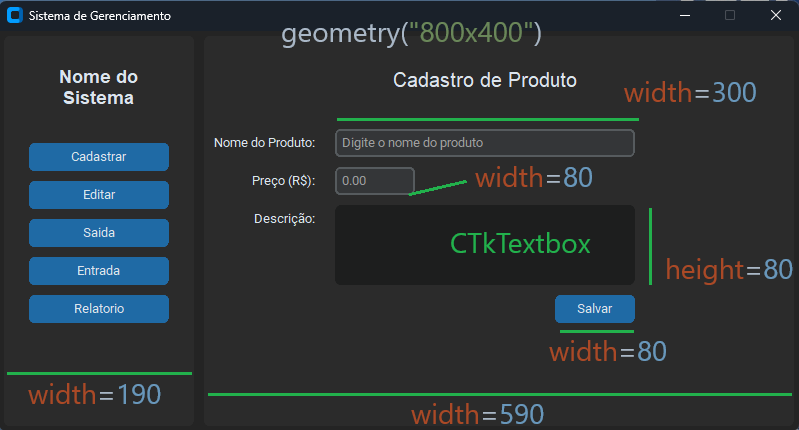
Largura: 800 Pixel

Interface gráfica do usuário, Texto, Aplicativo, chat ou mensagem de texto

O conteúdo gerado por IA pode estar incorreto.

Comprimento: 400 Pixel

Configurações dos Frames, botões e etc



widht (largura), height(altura)

**Coluna “1” do Frame 2**

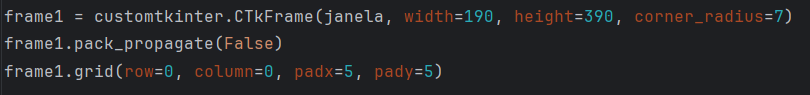
**Coluna “0” do Frame 1**

Interface gráfica do usuário, Texto, Aplicativo, chat ou mensagem de texto

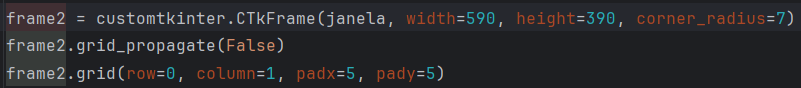
O conteúdo gerado por IA pode estar incorreto.

O CTkFrame cria um frame que abre em uma “**janela”** width=190(pixel de largura), e height=390(pixel de altura). Obs: O “corner\_radius” cria uma curvatura nos quatros cantos do frame, assemelhando-se a um contorno no quadro, quebrando a quina do retângulo, de uma similaridade das arestas de um botão.

Leitura do código do frame1: “row=0” (linha=0), column (coluna=0), “padx=5”, (distância horizontal = eixo x), “pady=5”(distância vertical=eixo y)

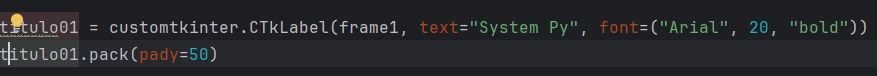


Leitura do código do frame2: “row=0” (linha=0), column (coluna=1), “padx=5”, (distância horizontal = eixo x), “pady=5”(distância vertical=eixo y). As atribuições s do Frame 1 são as mesmas do frame2, o que muda é a coluna, no caso =1.



O frame2.grid (row=0, column=1, padx=5, pady=5) mantém ativo quadro de inicialização sistema, no caso o frame2.

Widgets frame1 CTkLabel pode abrir em outro quadro, no caso frame1. Texto a ser exibido no alto do frame é um titulo “System Py”, fonte=(”Arial, tamanho 20, “bold = negrito) e o titulo.pack ou seja empilhado a (pady=50 na posição do eixo y = 50)



A criação do botão deve obedecer dentro do widget um posicionamento empilhado de forma lógica afim de não se tornar desorganizado e sem visão de novas resoluções pros problemas dispostos

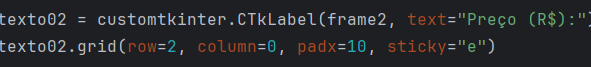


Observando a última imagem e adicionando a figura abaixo percebemos que ele está criando um botão que abre na (janela = frame1, com seguinte texto = “Cadastrar” e o (command=abrir\_frame2), mas você lembra de onde vem essa função definida, ela foi declarada desde o início do projeto ou seja “def abrir\_frame2():

)Texto

O conteúdo gerado por IA pode estar incorreto.

O CTkLabel é uma ferramenta que rotula o texto, no caso no frame2 é aberto com as configurações q vc já sabe e vou acrescentar apenas a (stick=”e”) que no caso é “e=leste”, pois pode -se atribuir os valores cardeais como : Norte, Sul, Leste Oeste e demais associações.



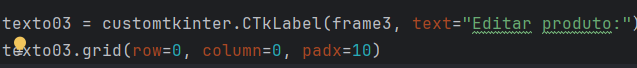
O CTkEntry é um campo de inserção de texto a ser oferecida em um “placeholder\_text=”digite o nome do produto”



Na prática:



O texto03 é chamado à execução no CTkLabel abrindo no frame3 o referido texto=”Editar Produto”



Na prática

Interface gráfica do usuário, Aplicativo

O conteúdo gerado por IA pode estar incorreto.

O CTkTextbox é uma ferramenta de inserção de texto dentro do frame2 de espaçamento e altura a serem definidos pelo usuário como por exemplo a descrição do produto. Diiferente do CTkEntry que oferece um pequeno campo de pesquisa por exemplo.



Na prática:

Interface gráfica do usuário

O conteúdo gerado por IA pode estar incorreto.

* O argumento columnspan permite que o widget se estenda por várias colunas.
* O argumento rownspan permite que o widget se estenda por várias linhas.

Ordenação de widgets:

Edição do Frame 1

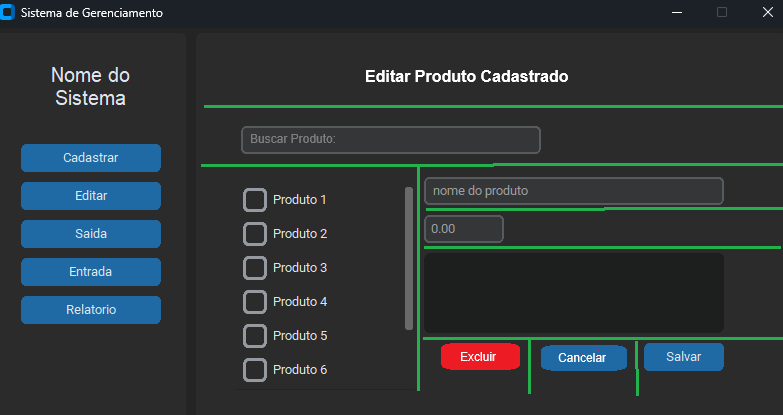
columnspan=4

Interface gráfica do usuário, Aplicativo, Tabela, Excel

O conteúdo gerado por IA pode estar incorreto.

rowspan=4

A divisão do Frame 1 exposta acima pode ser melhor elucidada com a figura abaixo do projeto afim de que comparem e obtenham o entendimento



**Produzido por: Lucas Gabriel da Silva e Arilson Bessa da Silva**

Termina a edição do sistema

Texto, Logotipo

O conteúdo gerado por IA pode estar incorreto.