



**INSTITUTO
FEDERAL**
Mato Grosso

Campus
Pontes e Lacerda
Fronteira Oeste

Aluna: Mayerly surubi roca
Data: 06/10/25

```
import tkinter as tk
from tkinter import messagebox
import tkinter as tk
from tkinter import messagebox

ARQUIVO = "dados.txt"

def carregar_dados():
    try:
        with open(ARQUIVO, "r", encoding="utf-8") as f:
            linhas = f.readlines()
            return [linha.strip().split(";") for linha in linhas]
    except FileNotFoundError:
        return []

def salvar_dados(dados):
    with open(ARQUIVO, "w", encoding="utf-8") as f:
        for registro in dados:
            f.write(";".join(registro) + "\n")

def atualizar_lista():
    lista.delete(0, tk.END)
    for i, (nome, idade, email) in enumerate(carregar_dados(), start=1):
        lista.insert(tk.END, f"{i} - Nome: {nome}, Idade: {idade}, email {email}")

def adicionar():
```

```

nome = entry_nome.get().strip()
idade = entry_idade.get().strip()
email = entry_email.get().strip()
if not nome or not idade:
    messagebox.showwarning("Atenção", "Preencha todos os campos!")
    return
dados = carregar_dados()
dados.append([nome, idade, email])
salvar_dados(dados)
atualizar_lista()
entry_nome.delete(0, tk.END)
entry_idade.delete(0, tk.END)
entry_email.delete(0, tk.END)

def editar():
    selecionado = lista.curselection()
    if not selecionado:
        messagebox.showwarning("Atenção", "Selecione um item para
editar!")
        return
    indice = selecionado[0]
    dados = carregar_dados()
    nome = entry_nome.get().strip()
    idade = entry_idade.get().strip()
    email = entry_email.get().strip()
    if not nome or not idade:
        messagebox.showwarning("Atenção", "Preencha todos os campos
para editar!")
        return
    dados[indice] = [nome, idade, email]
    salvar_dados(dados)
    atualizar_lista()

def deletar():
    selecionado = lista.curselection()
    if not selecionado:
        messagebox.showwarning("Atenção", "Selecione um item para
deletar!")
        return
    indice = selecionado[0]
    dados = carregar_dados()
    dados.pop(indice)
    salvar_dados(dados)

```

```

    atualizar_lista()

def carregar_campos(event):
    selecionado = lista.curselection()
    if selecionado:
        indice = selecionado[0]
        dados = carregar_dados()
        nome, idade, email = dados[indice]
        entry_nome.delete(0, tk.END)
        entry_nome.insert(0, nome)
        entry_idade.delete(0, tk.END)
        entry_idade.insert(0, idade)
        entry_email.delete(0, tk.END)
        entry_email.insert(0, email)

root = tk.Tk()
root.title("CRUD com Tkinter e TXT")

tk.Label(root, text="Nome:").grid(row=0, column=0, padx=5, pady=5)
entry_nome = tk.Entry(root)
entry_nome.grid(row=0, column=1, padx=5, pady=5)

tk.Label(root, text="Idade:").grid(row=1, column=0, padx=5, pady=5)
entry_idade = tk.Entry(root)
entry_idade.grid(row=1, column=1, padx=5, pady=5)

tk.Label(root, text="Email:").grid(row=2, column=0, padx=5, pady=5)
entry_email = tk.Entry(root)
entry_email.grid(row=2, column=1, padx=5, pady=5)

tk.Button(root, text="Adicionar", command=adicionar).grid(row=3,
column=0, padx=5, pady=5)
tk.Button(root, text="Editar", command=editar).grid(row=3, column=1,
padx=5, pady=5)
tk.Button(root, text="Deletar", command=deletar).grid(row=3, column=2,
padx=5, pady=5)

lista = tk.Listbox(root, width=50)
lista.grid(row=4, column=0, columnspan=3, padx=5, pady=5)
lista.bind("<<ListboxSelect>>", carregar_campos)

atualizar_lista()

```

```
root.mainloop()

ARQUIVO = "dados.txt"

def carregar_dados():
    try:
        with open(ARQUIVO, "r", encoding="utf-8") as f:
            linhas = f.readlines()
            return [linha.strip().split(";") for linha in linhas]
    except FileNotFoundError:
        return []

def salvar_dados(dados):
    with open(ARQUIVO, "w", encoding="utf-8") as f:
        for registro in dados:
            f.write(";".join(registro) + "\n")

def atualizar_lista():
    lista.delete(0, tk.END)
    for i, (nome, idade, email) in enumerate(carregar_dados(), start=1):
        lista.insert(tk.END, f"{i} - Nome: {nome}, Idade: {idade}, email {email}")

def adicionar():
    nome = entry_nome.get().strip()
    idade = entry_idade.get().strip()
    email = entry_email.get().strip()
    if not nome or not idade:
        messagebox.showwarning("Atenção", "Preencha todos os campos!")
        return
    dados = carregar_dados()
    dados.append([nome, idade, email])
    salvar_dados(dados)
    atualizar_lista()
    entry_nome.delete(0, tk.END)
    entry_idade.delete(0, tk.END)
    entry_email.delete(0, tk.END)

def editar():
```

```

selecionado = lista.curselection()
if not selecionado:
    messagebox.showwarning("Atenção", "Selecione um item para
editar!")
    return
indice = selecionado[0]
dados = carregar_dados()
nome = entry_nome.get().strip()
idade = entry_idade.get().strip()
email = entry_email.get().strip()
if not nome or not idade:
    messagebox.showwarning("Atenção", "Preencha todos os campos
para editar!")
    return
dados[indice] = [nome, idade, email]
salvar_dados(dados)
atualizar_lista()

def deletar():
    selecionado = lista.curselection()
    if not selecionado:
        messagebox.showwarning("Atenção", "Selecione um item para
deletar!")
        return
    indice = selecionado[0]
    dados = carregar_dados()
    dados.pop(indice)
    salvar_dados(dados)
    atualizar_lista()

def carregar_campos(event):
    selecionado = lista.curselection()
    if selecionado:
        indice = selecionado[0]
        dados = carregar_dados()
        nome, idade, email= dados[indice]
        entry_nome.delete(0, tk.END)
        entry_nome.insert(0, nome)
        entry_idade.delete(0, tk.END)
        entry_idade.insert(0, idade)
        entry_email.delete(0, tk.END)
        entry_email.insert(0, email)

```

```
root = tk.Tk()
root.title("CRUD com Tkinter e TXT")

tk.Label(root, text="Nome:").grid(row=0, column=0, padx=5, pady=5)
entry_nome = tk.Entry(root)
entry_nome.grid(row=0, column=1, padx=5, pady=5)

tk.Label(root, text="Idade:").grid(row=1, column=0, padx=5, pady=5)
entry_idade = tk.Entry(root)
entry_idade.grid(row=1, column=1, padx=5, pady=5)

tk.Label(root, text="Email:").grid(row=2, column=0, padx=5, pady=5)
entry_email = tk.Entry(root)
entry_email.grid(row=2, column=1, padx=5, pady=5)

tk.Button(root, text="Adicionar", command=adicionar).grid(row=3,
column=0, padx=5, pady=5)
tk.Button(root, text="Editar", command=editar).grid(row=3, column=1,
padx=5, pady=5)
tk.Button(root, text="Deletar", command=deletar).grid(row=3, column=2,
padx=5, pady=5)

lista = tk.Listbox(root, width=50)
lista.grid(row=4, column=0, columnspan=3, padx=5, pady=5)
lista.bind("<<ListboxSelect>>", carregar_campos)

atualizar_lista()

root.mainloop()
```