## 4.2. Листинги

|  |
| --- |
| Окна Messagebox  Окно askyesno  from tkinter import \*  from tkinter import messagebox as mb  def check():  answer = mb.askyesno(title="Вопрос", message="Перенести данные?")  if answer:  s = e.get()  e.delete(0, END)  m['text'] = s  window = Tk()  e = Entry()  e.pack()  b = Button(text="Передать", command=check)  b.pack()  m = Label(height=3)  m.pack()  window.mainloop() |
| Окно askokcancel  from tkinter import \*  from tkinter import messagebox as mb  def check():  answer = mb.askokcancel(title="Вопрос", message="Перенести данные?")  if answer:  s = e.get()  e.delete(0, END)  m['text'] = s  window = Tk()  e = Entry()  e.pack()  b = Button(text="Передать", command=check)  b.pack()  m = Label(height=3)  m.pack()  window.mainloop() |
| Окно askretrycancel  from tkinter import \*  from tkinter import messagebox as mb  def check():  answer = mb.askretrycancel(title="Вопрос", message="Перенести данные?")  if answer:  s = e.get()  e.delete(0, END)  m['text'] = s  window = Tk()  e = Entry()  e.pack()  b = Button(text="Передать", command=check)  b.pack()  m = Label(height=3)  m.pack()  window.mainloop() |
| Игра “Угадай число”  Используем окна showinfo и askretrycancel  from tkinter import \*  from tkinter import messagebox as mb  import random  def check():  s = e.get()  s = int(s)  rnd = random.randint(1, 2)  if s == rnd:  mb.showinfo(title="Результат", message="Угадал!")  else:  mb.showinfo(title="Результат", message="Неудача :(")    answer = mb.askretrycancel(title="Вопрос", message="Загадать еще одно число?")  if answer:  e.delete(0, END)  else:  window.destroy()  window = Tk()  window.title('Игра "Угадай число"')  m1 = Label(height=3, text="Введи число, которое я загадал, и нажми на кнопку")  m1.pack()  e = Entry()  e.pack()  b = Button(text="Передать", command=check)  b.pack()  m = Label(height=3)  m.pack()  window.mainloop() |
| Калькулятор для сложения |
| from tkinter import \*  from tkinter import messagebox as mb  def summ():  s1 = e1.get()  if not s1.lstrip('-').isdigit():  mb.showerror("Ошибка", "В первое поле должно быть введено число")  return # Возвращаемся из функции, если ввод некорректен  s2 = e2.get()  if not s2.lstrip('-').isdigit():  mb.showerror("Ошибка", "Во второе поле должно быть введено число")  return # Возвращаемся из функции, если ввод некорректен  slag1 = int(s1)  slag2 = int(s2)  summa = slag1 + slag2  m1['text']= f"{s1} + {s2} = {str(summa)}"    answer = mb.askretrycancel(title="Вопрос", message="Сложить еще два числа?")  if answer:  e1.delete(0, END)  e2.delete(0, END)  m1['text']= ""  else:  window.destroy()  window = Tk()  window.title("Калькулятор")  m = Label(height=3, text="Введи два числа и нажми на кнопку для вычисления суммы")  m.pack()  e1 = Entry()  e1.pack()  e2 = Entry()  e2.pack()  b = Button(text="Сложить два числа", command=summ)  b.pack()  m1 = Label(height=3)  m1.pack()  window.mainloop() |
| Задание  Сделай программу которая будет складывать три числа. Используй в третьем поле проверку введенного числа. |
| from tkinter import \*  from tkinter import messagebox as mb  def summ():  s1 = e1.get()  if not s1.lstrip('-').isdigit():  mb.showerror("Ошибка", "В первое поле должно быть введено число")  return # Возвращаемся из функции, если ввод некорректен  s2 = e2.get()  if not s2.lstrip('-').isdigit():  mb.showerror("Ошибка", "Во второе поле должно быть введено число")  return # Возвращаемся из функции, если ввод некорректен  s3 = e3.get()  if not s3.lstrip('-').isdigit():  mb.showerror("Ошибка", "В третье поле должно быть введено число")  return # Возвращаемся из функции, если ввод некорректен  slag1 = int(s1)  slag2 = int(s2)  slag3 = int(s3)  summa = slag1 + slag2 + slag3  m1['text'] = f"{s1} + {s2} + {s3} = {str(summa)}"    answer = mb.askretrycancel(title="Вопрос", message="Сложить еще три числа?")  if answer:  e1.delete(0, END)  e2.delete(0, END)  e3.delete(0, END)  m1['text'] = ""  else:  window.destroy()  window = Tk()  window.title("Калькулятор")  m = Label(height=3, text="Введите три числа и нажмите на кнопку для вычисления суммы")  m.pack()  e1 = Entry()  e1.pack()  e2 = Entry()  e2.pack()  e3 = Entry() # Добавление третьего поля для ввода  e3.pack()  b = Button(text="Сложить три числа", command=summ)  b.pack()  m1 = Label(height=3)  m1.pack()  window.mainloop() |
| Главное окно |
| Создаем окно указанного размера  from tkinter import \*  window = Tk()  window.title("Main window")  window.geometry("300x200+200+100") |
| Задание  Создайте окно размером 400 на 300 пикселей и разместите его на расстоянии 300 пикселей вниз от верхней границы экрана и на 500 пикселей правее левой границы экрана. |
| from tkinter import \*  window = Tk()  window.title("Main window")  window.geometry("400x300+500+300") |
| Определим размер монитора с помощью winfo\_screenwidth и winfo\_screenheight.  from tkinter import \*  window = Tk()  window.title("Main window")  window.geometry("400x300+500+300")  w = window.winfo\_screenwidth()  h = window.winfo\_screenheight()  print("Размер вашего экрана ", w, "x", h) |
| Располагаем окно строго в центре экрана.  from tkinter import \*  window = Tk()  window.title("Main window")  w = window.winfo\_screenwidth()  h = window.winfo\_screenheight()  print(f"Размер вашего экрана {w} x {h}")  w2 = w//2 - 200  h2 = h//2 - 150  print(f"Размер вашего экрана {w2} x {h2}")  window.geometry(f"400x300+{w2}+{h2}") |
| Загружаем проект Блокнот |
| from tkinter import \*  def delete():  text.delete(1.0, END)  def insert():  pushkin = "Это текстовое поле можно очистить с помощью кнопки"  text.insert(1.0, pushkin)  window = Tk()  text = Text(width=30, height=8, bg="gray", wrap=WORD)  text.pack(side=LEFT)  scroll = Scrollbar(command=text.yview)  scroll.pack(side=LEFT, fill=Y)  text.config(yscrollcommand=scroll.set)  b1 = Button(text="Вставка текста", command=insert)  b1.pack(side=LEFT)  b2 = Button(text="Удаление текста", command=delete)  b2.pack(side=LEFT)  window.mainloop() |
| Изменим текст в кнопке  b1 = Button(text="Открыть файл", command=insert) |
| Изменяем функцию insert() |
| from tkinter import \*  from tkinter import filedialog as fd  from tkinter import messagebox as mb  def delete():  text.delete(1.0, END)  def insert():  try:  file = fd.askopenfilename()  if file: # Проверяем, выбран ли файл  with open(file, 'r') as f: #encoding='utf-8'  s = f.read()  text.insert(1.0, s)  except FileNotFoundError:  mb.showerror("Ошибка", "Файл не найден")  except Exception as e:  mb.showerror("Ошибка", f"Произошла ошибка: {e}")  window = Tk()  text = Text(window, width=30, height=8, bg="gray", wrap=WORD)  text.pack(side=LEFT)  scroll = Scrollbar(window, command=text.yview)  scroll.pack(side=LEFT, fill=Y)  text.config(yscrollcommand=scroll.set)  b1 = Button(window, text="Открыть файл", command=insert)  b1.pack(side=LEFT)  b2 = Button(window, text="Удаление текста", command=delete)  b2.pack(side=LEFT)  window.mainloop() |
| Добавляем функцию сохранения файла  def save():  try:  file = fd.asksaveasfilename(  filetypes=(("TXT files", "\*.txt"),  ("HTML files", "\*.html;\*.htm"),  ("All files", "\*.\*")))  if file: # Проверяем, выбран ли файл  with open(file, 'w') as f:  s = text.get(1.0, END)  f.write(s)  except FileNotFoundError:  mb.showerror("Ошибка", "Не удалось сохранить файл")  except Exception as e:  mb.showerror("Ошибка", f"Произошла ошибка: {e}") |
| Добавляем меню (заменяем весь код с кнопками на этот)  window = Tk()  mainmenu = Menu(window)  window.config(menu=mainmenu)  filemenu = Menu(mainmenu, tearoff=0)  filemenu.add\_command(label="Открыть", command=insert)  filemenu.add\_command(label="Новый", command=delete)  filemenu.add\_command(label="Сохранить", command=save)  filemenu.add\_separator()  filemenu.add\_command(label="Выход", command=quit)  mainmenu.add\_cascade(label="Файл", menu=filemenu)  text = Text(width=30, height=8, bg="lightgray", wrap=WORD)  text.pack(side=LEFT)  scroll = Scrollbar(command=text.yview)  scroll.pack(side=LEFT, fill=Y)  text.config(yscrollcommand=scroll.set)  window.mainloop() |
| Необходимо добавить функцию для выхода из приложения.  def quit():  window.destroy() |
| Просмотр изображений  from tkinter import \*  from PIL import Image, ImageTk  from tkinter import filedialog as fd    window = Tk()  window.title("PHOTO")  window.geometry("500x500")  mainmenu = Menu(window)  window.config(menu=mainmenu)  filemenu = Menu(mainmenu, tearoff=0)  filemenu.add\_command(label="Открыть...", command=open)  filemenu.add\_separator()  filemenu.add\_command(label="Выход", command=quit)  mainmenu.add\_cascade(label="Файл", menu=filemenu)  window.mainloop() |
| Добавляем функцию открытия файла  from tkinter import \*  from PIL import Image, ImageTk  from tkinter import filedialog as fd  def open():  #global imgconv, l  try:  file = fd.askopenfilename()  if file: # Проверяем, был ли выбран файл  img = Image.open(file)  imgconv = img  # Адаптируем размер изображения под размер окна  window\_width = 500  window\_height = 500  img.thumbnail((window\_width, window\_height))  img = ImageTk.PhotoImage(img)  # Если лейбл уже существует, обновляем его изображение  if 'l' in globals():  l.configure(image=img)  l.image = img  else:  l = Label(image=img)  l.pack()  l.image = img  except FileNotFoundError:  messagebox.showerror("Ошибка", "Файл не найден")  except OSError: # Ошибка при открытии файла или неизображения  messagebox.showerror("Ошибка", "Не удалось открыть файл, возможно, это не изображение")  except Exception as e:  messagebox.showerror("Ошибка", f"Произошла ошибка: {e}")    window = Tk()  window.title("PHOTO")  window.geometry("500x500")  mainmenu = Menu(window)  window.config(menu=mainmenu)  filemenu = Menu(mainmenu, tearoff=0)  filemenu.add\_command(label="Открыть...", command=open)  filemenu.add\_separator()  filemenu.add\_command(label="Выход", command=quit)  mainmenu.add\_cascade(label="Файл", menu=filemenu)  window.mainloop() |