import tkinter as tk

from tkinter import messagebox

import threading

import random

import time

*# Функция для генерации случайной температуры*

**def** generate\_random\_temp():

    return random.uniform(20, 40)

*# Функция для отправки предупреждения о критической температуре*

**def** send\_warning(*temp*):

    messagebox.showwarning("Предупреждение", **f**"Критическая температура: {temp} °C")

*# Функция, которая будет выполняться в отдельном потоке*

**def** temp\_monitor():

    critical\_temp = 30  *# Критическая температура (в градусах Цельсия)*

    while True:

        temp = generate\_random\_temp()

        if temp >= critical\_temp:

            send\_warning(temp)

        time.sleep(5)  *# Пауза перед повторным считыванием температуры*

*# Создание GUI*

**def** create\_gui():

    root = tk.Tk()

    root.title("Контроль температуры")

*# Создание метки для отображения текущей температуры*

    temp\_label = tk.Label(root, *text*="Текущая температура:")

    temp\_label.pack()

*# Создание метки для отображения текущей температуры (значение)*

    temp\_value\_label = tk.Label(root, *text*="")

    temp\_value\_label.pack()

*# Создание и запуск потока для мониторинга температуры*

    temp\_thread = threading.Thread(*target*=temp\_monitor)

    temp\_thread.daemon = True  *# Поток будет завершен, когда основное окно будет закрыто*

    temp\_thread.start()

*# Функция для обновления значения температуры на метке*

**def** update\_temp\_label():

        temp = generate\_random\_temp()

        temp\_value\_label.config(*text*=**f**"{temp**:.2f**} °C")

        temp\_value\_label.after(1000, update\_temp\_label)  *# Обновление каждую секунду*

    update\_temp\_label()

    root.mainloop()

if \_\_name\_\_ == "\_\_main\_\_":

    create\_gui()