Лабораторная работа №9

**Цель работы**: рассмотреть возможности библиотеки Tkinter, её базовые виджеты и изучить основные принципы создания приложений с графическим интерфейсом с помощью данной библиотеки.

**Задание:** 19)Написать GUI приложение, которое представляет собой библиотеку с поиском книг по автору и наименованию, добавлению новых книг и чтению книг в форматах epub, fb2 и pdf.

**Выполнение задания:** Ознакомившись с материалом приступаем к работе. Для работы кода потребуется ebooklib для работы с EPUB и FB2, а также PyPDF2 для работы с PDF.   
  
Листинг кода:

import tkinter as tk  
from tkinter import messagebox, filedialog  
import os  
from ebooklib import epub  
import PyPDF2  
  
class LibraryApp:  
 def \_\_init\_\_(self, root):  
 self.root = root  
 self.root.title("Библиотека")  
  
 self.books = []  
  
 # Создание интерфейса  
 self.create\_widgets()  
  
 def create\_widgets(self):  
 # Поля для ввода данных о книге  
 tk.Label(self.root, text="Название книги:").grid(row=0, column=0)  
 self.title\_entry = tk.Entry(self.root)  
 self.title\_entry.grid(row=0, column=1)  
  
 tk.Label(self.root, text="Автор:").grid(row=1, column=0)  
 self.author\_entry = tk.Entry(self.root)  
 self.author\_entry.grid(row=1, column=1)  
  
 # Кнопка для добавления книги  
 self.add\_button = tk.Button(self.root, text="Добавить книгу", command=self.add\_book)  
 self.add\_button.grid(row=2, column=0, columnspan=2)  
  
 # Поле для поиска  
 tk.Label(self.root, text="Поиск:").grid(row=3, column=0)  
 self.search\_entry = tk.Entry(self.root)  
 self.search\_entry.grid(row=3, column=1)  
  
 # Кнопка для поиска книги  
 self.search\_button = tk.Button(self.root, text="Поиск", command=self.search\_book)  
 self.search\_button.grid(row=4, column=0, columnspan=2)  
  
 # Кнопка для чтения книги  
 self.read\_button = tk.Button(self.root, text="Читать книгу", command=self.read\_book)  
 self.read\_button.grid(row=5, column=0, columnspan=2)  
  
 # Список книг  
 self.book\_listbox = tk.Listbox(self.root)  
 self.book\_listbox.grid(row=6, column=0, columnspan=2)  
  
 def add\_book(self):  
 title = self.title\_entry.get()  
 author = self.author\_entry.get()  
  
 if title and author:  
 file\_path = filedialog.askopenfilename(title="Выберите файл книги", filetypes=[("EPUB files", "\*.epub"), ("FB2 files", "\*.fb2"), ("PDF files", "\*.pdf")])  
 if file\_path:  
 self.books.append({"title": title, "author": author, "path": file\_path})  
 self.book\_listbox.insert(tk.END, f"{title} - {author}")  
 self.title\_entry.delete(0, tk.END)  
 self.author\_entry.delete(0, tk.END)  
 messagebox.showinfo("Успех", "Книга добавлена!")  
 else:  
 messagebox.showwarning("Ошибка", "Пожалуйста, заполните все поля!")  
  
 def search\_book(self):  
 query = self.search\_entry.get().lower()  
 self.book\_listbox.delete(0, tk.END)  
  
 for book in self.books:  
 if query in book["title"].lower() or query in book["author"].lower():  
 self.book\_listbox.insert(tk.END, f"{book['title']} - {book['author']}")  
  
 def read\_book(self):  
 selected\_index = self.book\_listbox.curselection()  
 if selected\_index:  
 book = self.books[selected\_index[0]]  
 os.startfile(book["path"]) # Открывает файл с помощью связанного приложения  
 else:  
 messagebox.showwarning("Ошибка", "Пожалуйста, выберите книгу для чтения.")  
  
if \_\_name\_\_ == "\_\_main\_\_":  
 root = tk.Tk()  
 app = LibraryApp(root)  
 root.mainloop()

Результаты работы кода:



Рис. 1 - Интерфейс и работа кода