Министерство образования Республики Беларусь  
Учреждение образования  
«Брестский государственный технический университет»  
Кафедра ИИТ

Лабораторная работа №6  
По дисциплине: «Создание приложения для работы с БД организация пользовательского интерфейса: табличное представление и отчеты»

Выполнил:  
студент 3 курса  
группы ИИ-21  
Литвинюк Т. В.

Проверил:  
Савонюк В. А.

Брест 2023

**Цель работы**: получить навыки разработки приложений БД и организации пользовательского интерфейса: табличного представления информации и построения отчетов.

**Задание**

1. Изучить материал, приведенный в “К лабораторной работе 6.doc”.

2. На основании логической модели (в соответствии с вариантом Лабораторной работы №1) создать приложение и организовать пользовательский интерфейс, используя соответствующие элементы и методы: организовать табличное представление информации и построить отчеты не менее 4.

**Код программы:**

import tkinter as tk

from prettytable import PrettyTable

from main3 import DatabaseApp

class App(DatabaseApp):

def \_\_init\_\_(self, root):

super().\_\_init\_\_(root)

self.search\_value = ""

def display\_table(self):

def choose\_sort():

def f():

self.query += f" ORDER BY [{self.var.get()}] {'ASC' if self.var2.get() else 'DESC'}"

win2.destroy()

win2 = tk.Tk()

self.var = tk.StringVar(win2, columns[0].strip())

cb = [tk.Radiobutton(win2, text=column, variable=self.var, value=column.strip()) for column in columns]

for el in cb:

el.pack()

self.var2 = tk.BooleanVar(win2, False)

but1 = tk.Radiobutton(win2, text="По убыванию", variable=self.var2, value=False)

but2 = tk.Radiobutton(win2, text="По возрастанию", variable=self.var2, value=True)

but1.pack()

but2.pack()

win2.protocol("WM\_DELETE\_WINDOW", f)

def show():

table = PrettyTable(columns)

self.cursor.execute(self.query)

for item in self.cursor.fetchall():

table.add\_row(item)

string = self.query + "\n\n"

string += str(table)

win2 = tk.Tk()

win2.title("Отчет")

text = tk.Label(win2, text=string, font=("Consolas", 15))

text.pack()

win2.mainloop()

def choose\_find():

def f():

self.search\_value = field.get()

self.query += f" WHERE [{var.get()}] LIKE '%{self.search\_value}%'"

win2.destroy()

win2 = tk.Tk()

var = tk.StringVar(win2, columns[0].strip())

cb = [tk.Radiobutton(win2, text=column, variable=var, value=column.strip()) for column in columns]

for el in cb:

el.pack()

field = tk.Entry(win2)

field.pack()

win2.protocol("WM\_DELETE\_WINDOW", f)

def reset():

self.query = f"SELECT \* FROM {selected\_table}"

selected\_table = self.table\_var.get()

if selected\_table == "Выберите таблицу":

return

columns = self.get\_columns(selected\_table)

table = PrettyTable()

self.query = f"SELECT \* FROM {selected\_table}"

win = tk.Tk()

sort\_btn = tk.Button(win, text="sort", command=choose\_sort)

show\_btn = tk.Button(win, text="show", command=show)

reset\_btn = tk.Button(win, text="reset", command=reset)

find\_btn = tk.Button(win, text="find", command=choose\_find)

sort\_btn.pack()

show\_btn.pack()

reset\_btn.pack()

find\_btn.pack()

win.mainloop()

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

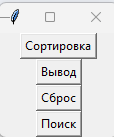
root = tk.Tk()

app = App(root)

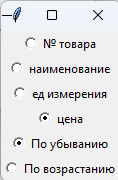
root.mainloop()

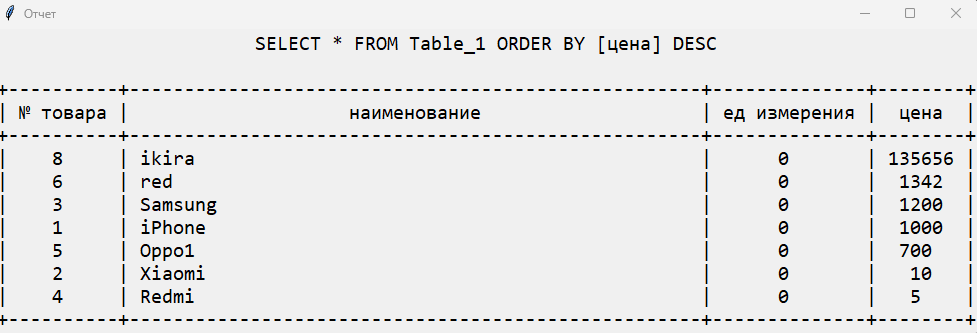
**Описание работы программы:**

Окно для выбора записей в таблице:

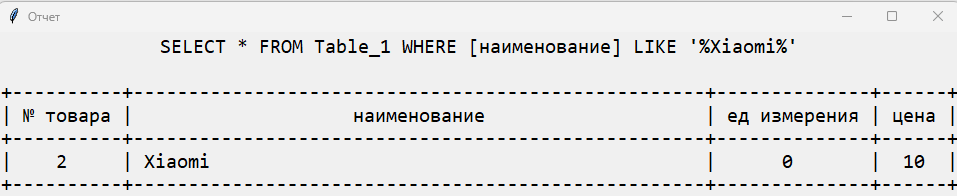
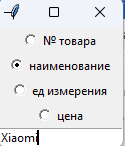


Окно с выбором сортировки:





Окно с заполнением параметров поиска:



**Вывод**: в ходе лабораторной работы я создал приложение с графическим интерфейсом для управления базой данных.