

КП.АС-56.190261 - 02 12 00

ПРИЛОЖЕНИЕ Г. ТЕКСТ ПРОГРАММЫ

```

import sqlite3
from sqlite3 import Error
import tkinter as tk
from tkinter import messagebox
import tkinter.ttk as ttk
import time
import mysql.connector
from ctypes import *

window = tk.Tk()
widgets_list = []
screen_width = windll.user32.GetSystemMetrics(0)
screen_height = windll.user32.GetSystemMetrics(1)
window.maxsize(screen_width, screen_height)

window_bg = '#769'
buttons_bg = '#879'
tree_font_size = 13

try:
    connection = mysql.connector.connect(host='localhost',
                                         port=3306,
                                         user='root',
                                         passwd='ff33039f',
                                         database='awp')

    cursor = connection.cursor()
except Error as e:
    print("Error while connecting to MySQL", e)

def destroy_widgets():
    for widget in widgets_list:
        widget.destroy()

def view_tabel2(table_name, table_columns, meta_table_columns, del_args,
title, prev_screen):
    destroy_widgets()
    window.title(str(title))
    # window.attributes('-fullscreen', True)

    window.rowconfigure([0, 1, 2, 3, 4, 5, 6, 7, 8, 9], minsize=0, weight=0)
    window.columnconfigure([0, 1, 2], minsize=0, weight=0)

    window.rowconfigure(0, weight=1)
    window.rowconfigure(1, weight=8)
    window.rowconfigure(2, weight=1)
    window.columnconfigure([0, 1, 2, 3, 4], weight=1)

    frame1 = tk.Frame(window, bg=window_bg)
    frame1.grid(row=0, column=0, columnspan=5, sticky='nsew')
    frame2 = tk.Frame(window, bg='#579')
    frame2.grid(row=1, column=0, columnspan=5, sticky='nsew')
    frame31 = tk.Frame(window, bg=window_bg)
    frame31.grid(row=2, column=0, sticky='nsew')
    frame32 = tk.Frame(window, bg=window_bg)
    frame32.grid(row=2, column=1, sticky='nsew')
    frame33 = tk.Frame(window, bg=window_bg)
    frame33.grid(row=2, column=2, sticky='nsew')

```

КП.АС-56.190261 - 02 12 00

```

frame34 = tk.Frame(window, bg=window_bg)
frame34.grid(row=2, column=3, sticky='nsew')
frame35 = tk.Frame(window, bg=window_bg)
frame35.grid(row=2, column=4, sticky='nsew')

widgets_list.append(frame1)
widgets_list.append(frame2)
widgets_list.append(frame31)
widgets_list.append(frame32)
widgets_list.append(frame33)
widgets_list.append(frame34)
widgets_list.append(frame35)

title_label = tk.Label(frame1, text=title, bg=window_bg, font='Arial 18')
title_label.pack(expand=1)
if table_name == "Отчет по ТТН":
    cursor.execute("""Select ttn.`Номер ТТН`, ttn.`Дата прихода`
, ttn.`Код товара`, kt.`Название товара` ,
    kt.`Единицы измерения` ,
    kt.`Цены единицы`, kt.`Цены единицы`*9 as `количество` ,
    kt.`Цены единицы`*9 * kt.`Цены единицы` as `цена всего`
from `Товарно-транспортная накладная` ttn

left join `каталог товаров` kt on ttn.`Код товара`= kt.`Код товара`
""")
if table_name == "Отчет договора поставки":
    cursor.execute("""SELECT dt.`Номер договора`, dt.`Дата
заклучения`, dt.`Учетный номер поставщика` , dt.`Код товара`, dt.`Количество
единиц`, sp.`Название поставщика`, kt.`Название товара`, kt.`Единицы
измерения`, kt.`Цены единицы`, kt.`Цены единицы`*100 as `Количество единиц`
, sum(dt.`Количество единиц` * kt.`Цены единицы`*100) over (PARTITION BY
dt.`Номер договора`) `Всего` FROM `Договор Поставка товаров` dt
left join `каталог товаров` kt on dt.`Код товара`= kt.`Код товара`
left join `справочник поставщики` sp on sp.`Учетный номер поставщика`=
dt.`Учетный номер поставщика`
""")
records = cursor.fetchall()

columns = []

for i in range(1, len(table_columns) + 1):
    columns.append('#' + str(i))
window.tree = ttk.Treeview(frame2, show='headings', columns=columns)
ttk.Style().configure('Treeview', rowheight=tree_font_size * 2)
ttk.Style().configure("Treeview.Heading", font=('Arial', tree_font_size))
ttk.Style().configure("Treeview", font=('Arial', tree_font_size))
for i in range(0, len(columns)):
    window.tree.heading(columns[i], text=meta_table_columns[i])
    if i == 0 or i == 3: window.tree.column(columns[i], minwidth=0,
width=tree_font_size * 12, stretch='no')
    if i == 1: window.tree.column(columns[i], minwidth=0,
width=tree_font_size * 25, stretch='no')
    if i == 2: window.tree.column(columns[i], minwidth=0,
width=tree_font_size * 15, stretch='no')
    if i >= 4 and i < 6: window.tree.column(columns[i], minwidth=0,
width=tree_font_size * 6, stretch='no')
    if i >= 6 and i < 8: window.tree.column(columns[i], minwidth=0,
width=tree_font_size * 11, stretch='no')
    if i == 8: window.tree.column(columns[i], minwidth=0,
width=tree_font_size * 10, stretch='no')
    if i >= 9 and i < 40: window.tree.column(columns[i], minwidth=0,
width=int(tree_font_size * 2.2), stretch='no')

```

КП.AC-56.190261 - 02 12 00

```

        if i >= 40 and i < 47: window.tree.column(columns[i], minwidth=0,
width=tree_font_size * 14, stretch='no')

        ysb = ttk.Scrollbar(frame2, orient=tk.VERTICAL,
command=window.tree.yview)
        xsb = ttk.Scrollbar(frame2, orient=tk.HORIZONTAL,
command=window.tree.xview)
        window.tree.configure(yscroll=ysb.set)
        window.tree.configure(xscroll=xsb.set)

    for row in records:
        window.tree.insert('', tk.END, values=row)
    ysb.pack(side=tk.RIGHT, fill=tk.Y)
    window.tree.pack(side=tk.TOP, expand=1, fill=tk.BOTH)
    xsb.pack(side=tk.BOTTOM, fill=tk.X)

def add():
    add_window = tk.Toplevel()
    add_screen_height = (len(table_columns) + 1) * 50
    add_screen_width = 400
    add_window.maxsize(screen_width, screen_height)
    if add_screen_height > screen_height - 100:
        add_screen_height = screen_height - 100
    add_window.geometry(str(add_screen_width) + 'x' +
str(add_screen_height))
    add_window.title('Добавление записи')

    rows = []
    for i in range(0, len(meta_table_columns) + 1):
        rows.append(i + 1)
    add_window.columnconfigure([0, 1], weight=1)
    add_window.rowconfigure(rows, weight=1)

    labels = []
    entrys = []

    tabel_numbers = []
    cursor.execute("""SELECT worker_tabel_number FROM workers""")
    tabel_numbers_tuple = cursor.fetchall()
    for tabel_number in tabel_numbers_tuple:
        tabel_numbers.append(tabel_number[0])

    print(tabel_numbers)

    for i in range(0, len(table_columns)):
        labels.append(tk.Label(add_window,
text=str(meta_table_columns[i])))
        labels[i].grid(row=i, column=0, sticky='e')
        if i == 0:
            entrys.append(ttk.Combobox(add_window, width=30,
values=tabel_numbers))
        elif i >= 1 and i < 8:
            entrys.append(tk.Entry(add_window, width=30,
state='disabled'))
        elif i >= 9:
            entrys.append(tk.Entry(add_window, width=5))
        else:
            entrys.append(tk.Entry(add_window, width=30))
        entrys[i].grid(row=i, column=1, sticky='w')

def insert_data():
    ins_columns = []

```

КП.AC-56.190261 - 02 12 00

```

for i in range(0, len(table_columns)):
    ins_columns.append(entrys[i].get())

sql_command = 'INSERT INTO '
sql_command += str(table_name)
sql_command += ' ('
for i in range(0, len(table_columns)):
    sql_command += str(table_columns[i])
    if i != len(table_columns) - 1: sql_command += ', '
sql_command += ') VALUES ('
for i in range(0, len(table_columns)):
    sql_command += '?'
    if i != len(table_columns) - 1: sql_command += ', '
sql_command += ')'

try:
    cursor.execute(sql_command, tuple(ins_columns))
    connection.commit()
    view_tabel2(table_name, table_columns, meta_table_columns,
del_args, title, prev_screen)
except:
    messagebox.showerror(title='Ошибка', message='Не удалось
добавить запись в базу данных')

add_frame = tk.Frame(add_window, bg=window_bg)
add_frame.grid(row=len(meta_table_columns) + 1, column=0,
columnspan=2, sticky='nsew')

add_btn = tk.Button(add_frame, text='Добавить', command=insert_data,
width=20, background=buttons_bg)
cancel_btn = tk.Button(add_frame, text='Отмена',
command=add_window.destroy,
width=20, background=buttons_bg)
add_btn.pack(expand=1, side='left')
cancel_btn.pack(expand=1, side='right')

def delete():
    sql_command = 'DELETE FROM '
    sql_command += str(table_name)
    sql_command += ' WHERE '
    for i in range(0, len(del_args)):
        sql_command += str(del_args[i])
        sql_command += '='
        if i != len(del_args) - 1: sql_command += ' AND '

    del_value = []
    for selected_item in window.tree.selection():
        for i in range(0, len(del_args)):
            del_value.append(window.tree.set(selected_item, '#' +
str(table_columns.index(str(del_args[i])) + 1)))
        cursor.execute(sql_command, del_value)
        del_value = []
    connection.commit()
    view_tabel2(table_name, table_columns, meta_table_columns, del_args,
title, prev_screen)

def edit():
    edit_window = tk.Toplevel()
    edit_screen_height = (len(table_columns) + 1) * 50
    edit_screen_width = 400
    edit_window.maxsize(screen_width, screen_height)
    if edit_screen_height > screen_height - 100:

```

КП.АС-56.190261 - 02 12 00

```

        edit_screen_height = screen_height - 100
        edit_window.geometry(str(edit_screen_width) + 'x' +
str(edit_screen_height))
        edit_window.title('Редактирование записи')

        edit_value = []
        rows = []
        for i in range(0, len(table_columns) + 1):
            rows.append(i + 1)
        edit_window.columnconfigure([0, 1], weight=1)
        edit_window.rowconfigure(rows, weight=1)

        labels = []
        entrys = []

        for i in range(0, len(table_columns)):
            labels.append(tk.Label(edit_window,
text=str(meta_table_columns[i])))
            labels[i].grid(row=i, column=0)
            entrys.append(tk.Entry(edit_window, width=30))
            entrys[i].grid(row=i, column=1)

        try:
            entrys[0].insert(0, window.tree.set(window.tree.selection()[0],
'#1'))
        except:
            messagebox.showerror(title='Ошибка', message='Запись не выбрана')
            edit_window.destroy()
            return None
        for i in range(1, len(table_columns)):
            entrys[i].insert(0, window.tree.set(window.tree.selection()[0],
columns[i]))

    def update_data():
        upd_columns = []
        for i in range(0, len(table_columns)):
            upd_columns.append(entrys[i].get())
        sql_command = 'UPDATE '
        sql_command += str(table_name)
        sql_command += ' SET '
        for i in range(0, len(table_columns)):
            sql_command += str(table_columns[i])
            sql_command += '='
            if i != len(table_columns) - 1: sql_command += ', '
        sql_command += ' WHERE '
        for i in range(0, len(del_args)):
            sql_command += str(del_args[i])
            sql_command += '='
            if i != len(del_args) - 1: sql_command += ' AND '
        upd_columns.append(
            window.tree.set(window.tree.selection()[0], '#' +
str(table_columns.index(str(del_args[i])) + 1)))

        try:
            cursor.execute(sql_command, upd_columns)
            connection.commit()
            view_tabel2(table_name, table_columns, meta_table_columns,
del_args, title, prev_screen)
        except:
            messagebox.showerror(title='Ошибка', message='Не удалось
редактировать запись базы данных')
            edit_window.destroy()

```

КП.AC-56.190261 - 02 12 00

```

edit_frame = tk.Frame(edit_window, bg=window_bg)
edit_frame.grid(row=len(meta_table_columns) + 1, column=0,
columnspan=2, sticky='nsew')

add_btn = tk.Button(edit_frame, text='Сохранить',
command=update_data,
width=20, background=buttons_bg)
cancel_btn = tk.Button(edit_frame, text='Отмена',
command=edit_window.destroy,
width=20, background=buttons_bg)
add_btn.pack(expand=1, side='left')
cancel_btn.pack(expand=1, side='right')

btn_edit = tk.Button(frame32, text='Печать',
width=30, background=buttons_bg)
widgets_list.append(btn_edit)
btn_edit.pack(expand=1)

btn_exit = tk.Button(frame34, text='Выход', command=exit_window,
width=30, background=buttons_bg)
widgets_list.append(btn_exit)
btn_exit.pack(expand=1)

btn_back = tk.Button(frame35, text='Назад', command=prev_screen,
width=30, background=buttons_bg)
widgets_list.append(btn_back)
btn_back.pack(expand=1)

def view(table_name, table_columns, meta_table_columns, del_args, title,
prev_screen):
    destroy_widgets()
    window.title(str(title))
    # window.attributes('-fullscreen', True)

    window.rowconfigure([0, 1, 2, 3, 4, 5, 6, 7, 8, 9], minsize=0, weight=0)
    window.columnconfigure([0, 1, 2], minsize=0, weight=0)

    window.rowconfigure(0, weight=1)
    window.rowconfigure(1, weight=8)
    window.rowconfigure(2, weight=1)
    window.columnconfigure([0, 1, 2, 3, 4], weight=1)

    frame1 = tk.Frame(window, bg=window_bg)
    frame1.grid(row=0, column=0, columnspan=5, sticky='nsew')
    frame2 = tk.Frame(window, bg='#579')
    frame2.grid(row=1, column=0, columnspan=5, sticky='nsew')
    frame31 = tk.Frame(window, bg=window_bg)
    frame31.grid(row=2, column=0, sticky='nsew')
    frame32 = tk.Frame(window, bg=window_bg)
    frame32.grid(row=2, column=1, sticky='nsew')
    frame33 = tk.Frame(window, bg=window_bg)
    frame33.grid(row=2, column=2, sticky='nsew')
    frame34 = tk.Frame(window, bg=window_bg)
    frame34.grid(row=2, column=3, sticky='nsew')
    frame35 = tk.Frame(window, bg=window_bg)
    frame35.grid(row=2, column=4, sticky='nsew')

    widgets_list.append(frame1)
    widgets_list.append(frame2)
    widgets_list.append(frame31)

```

КП.AC-56.190261 - 02 12 00

```

widgets_list.append(frame32)
widgets_list.append(frame33)
widgets_list.append(frame34)
widgets_list.append(frame35)

title_label = tk.Label(frame1, text=title, bg=window_bg, font='Arial 24')
title_label.pack(expand=1)

cursor.execute("""SELECT * FROM """ + str(table_name))
records = cursor.fetchall()

columns = []

for i in range(1, len(table_columns) + 1):
    columns.append('#' + str(i))
window.tree = ttk.Treeview(frame2, show='headings', columns=columns)
ttk.Style().configure('Treeview', rowheight=tree_font_size * 2)
ttk.Style().configure("Treeview.Heading", font=('Arial', tree_font_size))
ttk.Style().configure("Treeview", font=('Arial', tree_font_size))
for i in range(0, len(columns)):
    window.tree.heading(columns[i], text=meta_table_columns[i])

ysb = ttk.Scrollbar(frame2, orient=tk.VERTICAL,
command=window.tree.yview)
xsb = ttk.Scrollbar(frame2, orient=tk.HORIZONTAL,
command=window.tree.xview)
window.tree.configure(yscroll=ysb.set)
window.tree.configure(xscroll=xsb.set)

for row in records:
    window.tree.insert('', tk.END, values=row)
ysb.pack(side=tk.RIGHT, fill=tk.Y)
window.tree.pack(side=tk.TOP, expand=1, fill=tk.BOTH)
xsb.pack(side=tk.BOTTOM, fill=tk.X)

def add():
    add_window = tk.Toplevel()
    add_screen_height = (len(table_columns) + 1) * 50
    add_screen_width = 400
    add_window.maxsize(screen_width, screen_height)
    if add_screen_height > screen_height - 100:
        add_screen_height = screen_height - 100
    add_window.geometry(str(add_screen_width) + 'x' +
str(add_screen_height))
    add_window.title('Добавление записи')

    rows = []
    for i in range(0, len(meta_table_columns) + 1):
        rows.append(i + 1)
    add_window.columnconfigure([0, 1], weight=1)
    add_window.rowconfigure(rows, weight=1)

    labels = []
    entrys = []

    for i in range(0, len(table_columns)):
        labels.append(tk.Label(add_window,
text=str(meta_table_columns[i])))
        labels[i].grid(row=i, column=0)
        entrys.append(tk.Entry(add_window, width=30))
        entrys[i].grid(row=i, column=1)

```

КП.AC-56.190261 - 02 12 00

```

def insert_data():
    ins_columns = []
    for i in range(0, len(table_columns)):
        ins_columns.append(entrys[i].get())

    sql_command = 'INSERT INTO '
    sql_command += str(table_name)
    sql_command += ' ('
    for i in range(0, len(table_columns)):
        sql_command += '`'
        sql_command += str(table_columns[i])
        sql_command += '`'
        if i != len(table_columns) - 1: sql_command += ', '
    sql_command += ') VALUES ('
    for i in range(0, len(ins_columns)):
        sql_command += "'"
        sql_command += ins_columns[i]
        sql_command += "'"
        if i != len(ins_columns) - 1: sql_command += ', '
    sql_command += ')'

    try:
        cursor.execute(sql_command)
        connection.commit()
        view(table_name, table_columns, meta_table_columns, del_args,
title, prev_screen)
    except:
        messagebox.showerror(title='Ошибка', message='Не удалось
добавить запись в базу данных')

    add_frame = tk.Frame(add_window, bg=window_bg)
    add_frame.grid(row=len(meta_table_columns) + 1, column=0,
columnspan=2, sticky='nsew')

    add_btn = tk.Button(add_frame, text='Добавить', command=insert_data,
                        width=20, background=buttons_bg, font='Arial 12')
    cancel_btn = tk.Button(add_frame, text='Отмена',
command=add_window.destroy,
                        width=20, background=buttons_bg, font='Arial
12')

    add_btn.pack(expand=1, side='left')
    cancel_btn.pack(expand=1, side='right')

def delete():
    sql_command = 'DELETE FROM '
    sql_command += str(table_name)
    sql_command += ' WHERE '
    for i in range(0, len(del_args)):
        sql_command += '`'
        sql_command += str(del_args[i])
        sql_command += '`'

        if i != len(del_args) - 1: sql_command += ' AND '

    del_value = []
    for selected_item in window.tree.selection():
        print(selected_item)
        for i in range(0, len(del_args)):
            del_value.append(window.tree.set(selected_item, '#' +
str(table_columns.index(str(del_args[i])) + 1)))
        sql_command += '='
    sql_command += del_value[0]

```


КП.АС-56.190261 - 02 12 00

```

sql_command += '''

cursor.execute(sql_command)
del_value = []

connection.commit()
view(table_name, table_columns, meta_table_columns, del_args, title,
prev_screen)

def edit():
    edit_window = tk.Toplevel()
    edit_screen_height = (len(table_columns) + 1) * 50
    edit_screen_width = 400
    edit_window.maxsize(screen_width, screen_height)
    if edit_screen_height > screen_height - 100:
        edit_screen_height = screen_height - 100
    edit_window.geometry(str(edit_screen_width) + 'x' +
str(edit_screen_height))
    edit_window.title('Редактирование записи')

    edit_value = []
    rows = []
    for i in range(0, len(table_columns) + 1):
        rows.append(i + 1)
    edit_window.columnconfigure([0, 1], weight=1)
    edit_window.rowconfigure(rows, weight=1)

    labels = []
    entrys = []

    for i in range(0, len(table_columns)):
        labels.append(tk.Label(edit_window,
text=str(meta_table_columns[i])))
        labels[i].grid(row=i, column=0)
        entrys.append(tk.Entry(edit_window, width=30))
        entrys[i].grid(row=i, column=1)

    try:
        entrys[0].insert(0, window.tree.set(window.tree.selection()[0],
'#1'))
    except:
        messagebox.showerror(title='Ошибка', message='Запись не выбрана')
        edit_window.destroy()
        return None
    for i in range(1, len(table_columns)):
        entrys[i].insert(0, window.tree.set(window.tree.selection()[0],
columns[i]))
    print(columns)

def update_data():

    upd_columns = []
    for i in range(0, len(table_columns)):
        upd_columns.append(entrys[i].get())

    sql_command = 'UPDATE '
    sql_command += str(table_name)
    sql_command += ' SET '

    for i in range(0, len(table_columns)):
        sql_command += "`"
        sql_command += str(table_columns[i])

```

КП.AC-56.190261 - 02 12 00

```

        sql_command += "`"
        sql_command += "="
        sql_command += "'"
        sql_command += upd_columns[i]
        sql_command += "'"
        if i != len(table_columns) - 1: sql_command += ', '

    sql_command += ' WHERE '
    for i in range(0, len(del_args)):
        sql_command += "`"
        sql_command += str(del_args[i])
        sql_command += "`"
        sql_command += "="
        sql_command += "'"
        sql_command += upd_columns[0]
        sql_command += "'"
        if i != len(del_args) - 1: sql_command += ' AND '
    upd_columns.append(
        window.tree.set(window.tree.selection()[0], '#' +
        str(table_columns.index(str(del_args[i])) + 1)))
    print(sql_command)

    try:
        cursor.execute(sql_command)
        connection.commit()
        view(table_name, table_columns, meta_table_columns, del_args,
        title, prev_screen)
    except:
        messagebox.showerror(title='Ошибка', message='Не удалось
        редактировать запись базы данных')
        edit_window.destroy()

    edit_frame = tk.Frame(edit_window, bg=window_bg)
    edit_frame.grid(row=len(meta_table_columns) + 1, column=0,
    columnspan=2, sticky='nsew')

    add_btn = tk.Button(edit_frame, text='Сохранить',
    command=update_data,
                                width=20, background=buttons_bg, font='Arial 12')
    cancel_btn = tk.Button(edit_frame, text='Отмена',
    command=edit_window.destroy,
                                width=20, background=buttons_bg, font='Arial
    12')
    add_btn.pack(expand=1, side='left')
    cancel_btn.pack(expand=1, side='right')

    btn_add = tk.Button(frame31, text='Добавить', command=add,
                                width=10, background=buttons_bg, font='Arial 12')
    widgets_list.append(btn_add)
    btn_add.pack(expand=1)

    btn_edit = tk.Button(frame32, text='Изменить', command=edit,
                                width=10, background=buttons_bg, font='Arial 12')
    widgets_list.append(btn_edit)
    btn_edit.pack(expand=1)

    btn_del = tk.Button(frame33, text='Удалить', command=delete,
                                width=10, background=buttons_bg, font='Arial 12')
    widgets_list.append(btn_del)
    btn_del.pack(expand=1)

    btn_back = tk.Button(frame34, text='Назад', command=prev_screen,

```

КП.АС-56.190261 - 02 12 00

```

        width=10, background=buttons_bg, font='Arial 12')
widgets_list.append(btn_back)
btn_back.pack(expand=1)

btn_exit = tk.Button(frame35, text='Выход', command=exit_window,
                    width=10, background=buttons_bg, font='Arial 12')
widgets_list.append(btn_exit)
btn_exit.pack(expand=1)

def docs_list():
    destroy_widgets()

    window.rowconfigure([0, 1, 2, 3, 4, 5, 6, 7, 8, 9], minsize=0, weight=0)
    window.columnconfigure([0, 1, 2, 3, 4, 5, 6, 7, 8, 9], minsize=0,
weight=0)

    window.rowconfigure([0, 1, 2, 3, 4, 5, 6, 7, 8, 9], minsize=0, weight=1)
    window.columnconfigure([0, 1, 2], minsize=0, weight=1)

    def view_wp10():
        view(" аwp.`план работы кружка 10`",
            ['Номер мероприятия', 'Название мероприятия', 'Дата
мероприятия'],
            ['Номер мероприятия', 'Название мероприятия', 'Дата
мероприятия'],
            ['Номер мероприятия'], 'План работы кружка 10', docs_list)

    def view_wp11():
        view(" аwp.`план работы кружка 11`",
            ['Номер мероприятия', 'Название мероприятия', 'Дата
мероприятия'],
            ['Номер мероприятия', 'Название мероприятия', 'Дата
мероприятия'],
            ['Номер мероприятия'], 'План работы кружка 11', docs_list)

    def view_juupm10():
        view("аwp.`жуупм по кружку 10`",
            ['Номер кружка', 'Табельный номер учителя', 'Код учащегося',
'Номер мероприятия'],
            ['Номер кружка', 'Табельный номер учителя', 'Код учащегося',
'Номер мероприятия'],
            ['Номер кружка'], 'ЖУУПМ по кружку 10', docs_list)

    def view_juupm11():
        view("аwp.`жуупм по кружку 11`",
            ['Номер кружка', 'Табельный номер учителя', 'Код учащегося',
'Номер мероприятия'],
            ['Номер кружка', 'Табельный номер учителя', 'Код учащегося',
'Номер мероприятия'],
            ['Номер кружка'], 'ЖУУПМ по кружку 11', docs_list)

    btn_tabel = tk.Button(window, text='План работы кружка 10',
command=view_wp10,
                    width=50, height=2, background=buttons_bg,
font='Arial 12')
    widgets_list.append(btn_tabel)
    btn_tabel.grid(row=0, column=0)

    btn_departments = tk.Button(window, text='План работы кружка 11',
command=view_wp11,
                    width=50, height=2, background=buttons_bg,

```


КП.АС-56.190261 - 02 12 00

```

        'Общее количество мероприятий, в которых участвовал'],
        ['Номер кружка'], 'Отчет о работе учащегося',
reports_list)

def view_working_conditions():
    view_tabel2('Отчет по школе', ['Назначение кружка', 'Общее количество
членов',
                                'Общее количество проведенных мероприятий'], ['Назначение
кружка', 'Общее количество членов',
                                'Общее количество проведенных мероприятий'],
        ['Назначение кружка'], 'Отчет по школе', reports_list)

    btn_positions = tk.Button(window, text='Отчет по участию 10А класса в
кружках',
                                width=50, height=2, background=buttons_bg,
font='Arial 12')
    widgets_list.append(btn_positions)
    btn_positions.grid(row=0, column=0)

    btn_departments = tk.Button(window, text='Отчет о работе кружка 10',
                                width=50, height=2, background=buttons_bg,
font='Arial 12')
    widgets_list.append(btn_departments)
    btn_departments.grid(row=1, column=0)

    btn_tariff = tk.Button(window, text='Отчет о работе учащегося ',
                                width=50, height=2, background=buttons_bg,
font='Arial 12')
    widgets_list.append(btn_tariff)
    btn_tariff.grid(row=2, column=0)

    btn_working_conditions = tk.Button(window, text='Отчет по школе ',
                                width=50, height=2,
background=buttons_bg, font='Arial 12')
    widgets_list.append(btn_working_conditions)
    btn_working_conditions.grid(row=3, column=0)

    btn_back = tk.Button(window, text='Назад', command=start_screen,
                                width=20, height=2, background=buttons_bg,
font='Arial 12')
    widgets_list.append(btn_back)
    btn_back.grid(row=4, column=0)

    btn_exit = tk.Button(window, text='Выход', command=exit_window,
                                width=20, height=2, background=buttons_bg,
font='Arial 12')
    widgets_list.append(btn_exit)
    btn_exit.grid(row=4, column=1)

def catalog_list():
    destroy_widgets()

    window.rowconfigure([0, 1, 2, 3, 4, 5, 6, 7, 8, 9], minsize=0, weight=0)
    window.columnconfigure([0, 1, 2, 3, 4, 5, 6, 7, 8, 9], minsize=0,
weight=0)

    window.rowconfigure([0, 1, 2, 3, 4, 5, 6, 7, 8, 9], minsize=0, weight=1)
    window.columnconfigure([0, 1, 2], minsize=0, weight=1)

def view_stl():
    view("awp. `список учителей школы`",

```

КП.АС-56.190261 - 02 12 00

```

        ['Табельный номер учителя', 'ФИО учителя', 'Должность учителя',
        'Дата приема на работу', 'Адрес учителя', 'Номер телефона учителя'],
        ['Табельный номер учителя', 'ФИО учителя', 'Должность учителя',
        'Дата приема на работу', 'Адрес учителя', 'Номер телефона учителя'],
        ['Табельный номер учителя'], 'Список учителей школы',
catalog_list)

def view_scl():
    view("авр. `список классов школы`",
        ['Номер класса', 'Табельный номер учителя', 'ФИО учителя',
        'Количество учащихся'],
        ['Номер класса', 'Табельный номер учителя', 'ФИО учителя',
        'Количество учащихся'],
        ['Номер класса'], 'Список классов школы', catalog_list)

def view_krl():
    view("авр. `список кружков`",
        ['Номер кружка', 'Название кружка', 'Табельный номер учителя'],
        ['Номер кружка', 'Название кружка', 'Табельный номер учителя'],
        ['Номер кружка'], 'Список кружков', catalog_list)

def view_pkrl10():
    view("авр. `список членов кружка 10`",
        ['Код учащегося', 'ФИО учащегося', 'Номер класса', 'Дата приема
в кружок'],
        ['Код учащегося', 'ФИО учащегося', 'Номер класса', 'Дата приема
в кружок'],
        ['Код учащегося'], 'Список членов кружка 10', catalog_list)

def view_pkrl11():
    view("авр. `список членов кружка 11`",
        ['Код учащегося', 'ФИО учащегося', 'Номер класса', 'Дата приема
в кружок'],
        ['Код учащегося', 'ФИО учащегося', 'Номер класса', 'Дата приема
в кружок'],
        ['Код учащегося'], 'Список членов кружка 11', catalog_list)

    btn_positions = tk.Button(window, text='Список учителей школы',
command=view_stl,
                                width=50, height=2, background=buttons_bg,
font='Arial 12')
    widgets_list.append(btn_positions)
    btn_positions.grid(row=0, column=0)

    btn_departments = tk.Button(window, text='Список классов школы',
command=view_scl,
                                width=50, height=2, background=buttons_bg,
font='Arial 12')
    widgets_list.append(btn_departments)
    btn_departments.grid(row=1, column=0)

    btn_tariff = tk.Button(window, text='Список кружков', command=view_krl,
width=50, height=2, background=buttons_bg,
font='Arial 12')
    widgets_list.append(btn_tariff)
    btn_tariff.grid(row=2, column=0)

    btn_working_conditions = tk.Button(window, text='Список членов кружка
10', command=view_pkrl10,
width=50, height=2,
background=buttons_bg, font='Arial 12')
    widgets_list.append(btn_working_conditions)

```

КП.АС-56.190261 - 02 12 00

```

    btn_working_conditions.grid(row=3, column=0)

    btn_working_conditions = tk.Button(window, text='Список членов кружка
11', command=view_pkr111,
                                width=50, height=2,
background=buttons_bg, font='Arial 12')
    widgets_list.append(btn_working_conditions)
    btn_working_conditions.grid(row=4, column=0)

    btn_back = tk.Button(window, text='Назад', command=start_screen,
                        width=20, height=2, background=buttons_bg,
font='Arial 12')
    widgets_list.append(btn_back)
    btn_back.grid(row=5, column=0)

    btn_exit = tk.Button(window, text='Выход', command=exit_window,
                        width=20, height=2, background=buttons_bg,
font='Arial 12')
    widgets_list.append(btn_exit)
    btn_exit.grid(row=5, column=1)

def exits():
    window.destroy()

def exit_window():
    destroy_widgets()
    window.geometry('300x150')
    window.title('Выход')

    window.rowconfigure([0, 1, 2, 3, 4], minsize=50, weight=1)
    window.columnconfigure([0, 1, 2, 3], minsize=50, weight=1)

    labell = tk.Label(window, text='Выйти из программы?',
                        font=("Arial Bold", 12))
    widgets_list.append(labell)
    labell.grid(row=0, column=1)

    btn_1 = tk.Button(window, text='Да', command=exits,
                        width=10, height=1, background=buttons_bg,
font='Arial 12')
    widgets_list.append(btn_1)
    btn_1.grid(row=1, column=0)

    btn_2 = tk.Button(window, text='Нет', command=start_screen,
                        width=10, height=1, background=buttons_bg,
font='Arial 12')
    widgets_list.append(btn_2)
    btn_2.grid(row=1, column=1)

def vosst():
    message = tk.messagebox.showwarning(message='Восстановление базы данных
прошло успешно!')

def spravka():
    destroy_widgets()
    window.geometry('700x450')
    window.title('Справка')

```

КП.АС-56.190261 - 02 12 00

```

window.rowconfigure([0, 1, 2, 3, 4, 5, 6, 7, 8, 9], minsize=0, weight=0)
window.columnconfigure([0, 1, 2, 3, 4, 5, 6, 7, 8, 9], minsize=0,
weight=0)

window.rowconfigure([0, 1, 2, 3, 4], minsize=50, weight=1)
window.columnconfigure([0, 1, 2, 3], minsize=50, weight=1)

label1 = tk.Label(window, text='Название АРМ:', font=("Arial Bold", 12))
widgets_list.append(label1)
label1.grid(row=0, column=0)

label2 = tk.Label(window, text='АРМ "Руководитель по внеклассной
работе"', font=("Arial Bold", 12))
widgets_list.append(label2)
label2.grid(row=0, column=1)

label3 = tk.Label(window, text='Версия программы:', font=("Arial Bold",
12))
widgets_list.append(label3)
label3.grid(row=1, column=0)

label4 = tk.Label(window, text='1.0.0.0', font=("Arial Bold", 12))
widgets_list.append(label4)
label4.grid(row=1, column=1)

label5 = tk.Label(window, text='Информация о разработчике:', font=("Arial
Bold", 12))
widgets_list.append(label5)
label5.grid(row=2, column=0)

label6 = tk.Label(window, text='Студентка БрГТУ, \nФакультет ЭИС, \n'
                               'Специальность "АСОИ", \nГруппа АС-56, 3
курс\nКарпенко Маргарита', font=("Arial Bold", 12))
widgets_list.append(label6)
label6.grid(row=2, column=1)

label7 = tk.Label(window, text='Описание:', font=("Arial Bold", 12))
widgets_list.append(label7)
label7.grid(row=3, column=0)

label8 = tk.Label(window, text='Данная АРМ разработана для\nсокращения
трудо затрат по ведению \n'
                               'информации и отчетных документов \nпри
решении комплекса задач \n'
                               'по внеклассной работе учителей',
font=("Arial Bold", 12))
widgets_list.append(label8)
label8.grid(row=3, column=1)

btn_back = tk.Button(window, text='Назад', command=start_screen,
                      width=20, height=2, background=buttons_bg,
font='Arial 12')
widgets_list.append(btn_back)
btn_back.grid(row=4, column=0)

def start_screen():
    destroy_widgets()
    window.title('АРМ "Руководитель по внеклассной работе"')

window.rowconfigure([0, 1, 2, 3, 4, 5, 6, 7, 8, 9], minsize=0, weight=0)
window.columnconfigure([0, 1, 2, 3, 4, 5, 6, 7, 8, 9], minsize=0,

```


КП.АС-56.190261 - 02 12 00

```

weight=0)

window.rowconfigure([0, 1, 2, 3, 4], minsize=50, weight=1)
window.columnconfigure([0, 1, 2, 3], minsize=50, weight=1)

btn_catalogs = tk.Button(window, text='Справочные документы',
command=catalog_list,
width=50, height=2, background=buttons_bg,
font='Arial 12')
widgets_list.append(btn_catalogs)
btn_catalogs.grid(row=0, column=0)

btn_docs = tk.Button(window, text='Оперативные документы',
command=docs_list,
width=50, height=2, background=buttons_bg,
font='Arial 12')
widgets_list.append(btn_docs)
btn_docs.grid(row=1, column=0)

btn_reports = tk.Button(window, text='Отчетные документы',
command=reports_list,
width=50, height=2, background=buttons_bg,
font='Arial 12')
widgets_list.append(btn_reports)
btn_reports.grid(row=2, column=0)

btn_spravka = tk.Button(window, text='Справка', command=spravka,
width=50, height=2, background=buttons_bg,
font='Arial 12')
widgets_list.append(btn_spravka)
btn_spravka.grid(row=3, column=0)

btn_vosst = tk.Button(window, text='Восстановление', command=vosst,
width=15, height=2, background=buttons_bg,
font='Arial 12')
widgets_list.append(btn_vosst)
btn_vosst.grid(row=4, column=0)

btn_exit = tk.Button(window, text='Выход', command=exit_window,
width=15, height=2, background=buttons_bg,
font='Arial 12')
widgets_list.append(btn_exit)
btn_exit.grid(row=4, column=1)

def pre_start():
    window.geometry('700x450')
    start_screen()

def login_screen():
    destroy_widgets()
    window.title('Аутентификация')

    window.rowconfigure([0, 1, 2, 3, 4, 5, 6, 7, 8], minsize=0, weight=0)
    window.columnconfigure([0, 1, 2, 3, 4, 5, 6, 7, 8], minsize=0, weight=0)
    window.rowconfigure([0, 1, 2], minsize=50, weight=1)
    window.columnconfigure([0, 1], minsize=50, weight=1)

    login_text = tk.Label(window, text='Логин: ', font='Arial 12')
    login_text.grid(row=0, column=0)
    widgets_list.append(login_text)

```

КП.АС-56.190261 - 02 12 00

```

login_folder = tk.Entry(window, width=20)
login_folder.grid(row=0, column=1)
widgets_list.append(login_folder)

password_text = tk.Label(window, text='Пароль: ', font='Arial 12')
password_text.grid(row=1, column=0)
widgets_list.append(password_text)
password_folder = tk.Entry(window, width=20)
password_folder.grid(row=1, column=1)
widgets_list.append(password_folder)

def user_check():
    login = login_folder.get()
    password = password_folder.get()
    if (login == '' and password == ''):
        pre_start()
    else:
        message = tk.messagebox.showwarning(message='Неверное имя
пользователя или пароль')

    btn_sign_in = tk.Button(window, text='Войти', command=user_check,
background=buttons_bg, width=20, font='Arial 12')
    btn_sign_in.grid(row=2, column=1)
    widgets_list.append(btn_sign_in)

def splash_screen():
    window.geometry('300x150')
    window.title(' ')

    labell = tk.Label(window, text='Автоматизированное рабочее
место\n"Руководитель по внеклассной работе"', font=("Arial Bold", 12))
    widgets_list.append(labell)
    labell.pack(expand='YES', fill='both')
    window.update()
    time.sleep(5)
    login_screen()

def main():
    splash_screen()
    window.mainloop()
    cursor.close()
    connection.close()
    print('Connection with DB has been close')

if __name__ == '__main__':
    main()

```