

Практическое занятие №17

Тема: составление программ с использованием GUI

Tkinter в IDE PyCharm Community, изучение возможностей модуля OS

Цель: закрепить усвоенные знания, понятия, алгоритмы,

основные принципы составления программ, приобрести навыки составления

программ с использованием GUI Tkinter в IDE PyCharm Community, изучить

возможности модуля OS.

Постановка задачи:

Задание 1. В соответствии с номером варианта перейти по ссылке на прототип. Реализовать его в IDE PyCharm Community с применением пакета tk. Получить интерфейс максимально приближенный к оригиналу (см. таблицу 1).

[https://lh5.googleusercontent.com/-](https://lh5.googleusercontent.com/-wG_YNAIbVZU/Ud696wJg0FI/AAAAAAAAACP4/eaIzPTZRixE/w596-h642-no/4_3.png)

[wG_YNAIbVZU/Ud696wJg0FI/AAAAAAAAACP4/eaIzPTZRixE/w596-h642-no/4_3.png](https://lh5.googleusercontent.com/-wG_YNAIbVZU/Ud696wJg0FI/AAAAAAAAACP4/eaIzPTZRixE/w596-h642-no/4_3.png)

Задание 2. Разработать программу с применением пакета tk, взяв в качестве условия одну любую задачу из ПЗ №№ 2 – 9.

Задание 3.

Задание предполагает, что у студента есть проект с практическими работами (№№ 2-13),

оформленный согласно требованиям. Все задания выполняются с использованием модуля

OS:

✓ перейдите в каталог PZ11. Выведите список всех файлов в этом каталоге. Имена вложенных подкаталогов выводить не нужно.

✓ перейти в корень проекта, создать папку с именем test. В ней создать еще одну папку test1. В папку test переместить два файла из ПЗ6, а в папку test1 - один файл из ПЗ7. Файл из ПЗ7 переименовать в test.txt. Вывести в консоль информацию о размере файлов в папке test.

✓ перейти в папку с PZ11, найти там файл с самым коротким именем, имя вывести в консоль. Использовать функцию basename () (os.path.basename()).

✓ перейти в любую папку где есть отчет в формате .pdf и «запустите» файл в привязанной к нему программе. Использовать функцию `os.startfile()`.

✓ удалить файл `test.txt`.

Текст программы:

```
import tkinter as tk
from tkinter import ttk

# Функция для обработки нажатия кнопки "Данные подтверждаю"
def submit():
    print("Данные подтверждены")

# Функция для обработки нажатия кнопки "Отменить ввод"
def cancel():
    root.destroy()

# Создаем главное окно
root = tk.Tk()
root.title("Форма")

# Метки и поля ввода для имени
tk.Label(root, text="Ваше имя:").grid(row=0, column=0, sticky=tk.W, padx=10, pady=5)
name_entry = tk.Entry(root)
name_entry.grid(row=0, column=1, padx=10, pady=5)

# Метки и поля ввода для пароля
tk.Label(root, text="Пароль:").grid(row=1, column=0, sticky=tk.W, padx=10, pady=5)
password_entry = tk.Entry(root, show="*")
password_entry.grid(row=1, column=1, padx=10, pady=5)

# Метки и поля ввода для возраста
tk.Label(root, text="Возраст:").grid(row=2, column=0, sticky=tk.W, padx=10, pady=5)
age_entry = tk.Entry(root)
age_entry.grid(row=2, column=1, padx=10, pady=5)

# Радиокнопки для выбора пола
tk.Label(root, text="Пол:").grid(row=3, column=0, sticky=tk.W, padx=10, pady=5)
gender_var = tk.StringVar()
tk.Radiobutton(root, text="Мужской", variable=gender_var, value="Мужской").grid(row=3, column=1,
sticky=tk.W)
tk.Radiobutton(root, text="Женский", variable=gender_var, value="Женский").grid(row=3, column=2,
sticky=tk.W)

# Флажки для выбора увлечений
tk.Label(root, text="Ваши увлечения:").grid(row=4, column=0, sticky=tk.W, padx=10, pady=5)
music_var = tk.BooleanVar()
video_var = tk.BooleanVar()
```

```

drawing_var = tk.BooleanVar()
tk.Checkbutton(root, text="Музыка", variable=music_var).grid(row=4, column=1, sticky=tk.W)
tk.Checkbutton(root, text="Видео", variable=video_var).grid(row=4, column=2, sticky=tk.W)
tk.Checkbutton(root, text="Рисование", variable=drawing_var).grid(row=4, column=3, sticky=tk.W)

# Выпадающие списки для выбора страны и города
tk.Label(root, text="Ваша страна:").grid(row=5, column=0, sticky=tk.W, padx=10, pady=5)
country_var = tk.StringVar()
country_combo = ttk.Combobox(root, textvariable=country_var)
country_combo['values'] = ("Country1", "Country2", "Country3") # Здесь замените на реальные значения
country_combo.grid(row=5, column=1, padx=10, pady=5)

tk.Label(root, text="Ваш город:").grid(row=6, column=0, sticky=tk.W, padx=10, pady=5)
city_var = tk.StringVar()
city_combo = ttk.Combobox(root, textvariable=city_var)
city_combo['values'] = ("City1", "City2", "City3") # Здесь замените на реальные значения
city_combo.grid(row=6, column=1, padx=10, pady=5)

# Поле для краткого описания о себе
tk.Label(root, text="Кратко о себе:").grid(row=7, column=0, sticky=tk.W, padx=10, pady=5)
about_text = tk.Text(root, width=30, height=5)
about_text.grid(row=7, column=1, columnspan=3, padx=10, pady=5)

# Метка и поле ввода для решения примера
tk.Label(root, text="Решите пример, запишите результат в поле ниже:").grid(row=8, column=0, sticky=tk.W,
padx=10, pady=5)
example_entry = tk.Entry(root)
example_entry.grid(row=8, column=1, padx=10, pady=5)

# Кнопки "Данные подтверждаю" и "Отменить ввод"
submit_button = tk.Button(root, text="Данные подтверждаю", command=submit)
submit_button.grid(row=9, column=2, padx=10, pady=10)

cancel_button = tk.Button(root, text="Отменить ввод", command=cancel)
cancel_button.grid(row=9, column=1, padx=10, pady=10)

# Запуск главного цикла обработки событий
root.mainloop()

```

2)

```

"""

```

*Разработать программу с применением пакета tk, взяв в качестве условия одну
любую задачу из ПЗ №№2 – 9.*

```

"""

```

```

import tkinter as tk
from tkinter import messagebox

```

```

def check():

```

```

try:
    A = int(entry.get())
    if A < 1000:
        result = f"Число {A} неправильное число"
    else:
        y = A // 100
        A = y % 10
        result = f"Число {A} соответствует разряду сотен"
        messagebox.showinfo("Результат", result)
except ValueError:
    print("Вы ввели не то, что нужно. Пожалуйста, попробуйте ещё раз!\n")

root = tk.Tk()
root.title("Проверка чётности числа")

label = tk.Label(root, text="Введите число более 999:")
label.pack(pady=10, padx=10)

entry = tk.Entry(root)
entry.pack(pady=5)

button = tk.Button(root, text="Проверить", command=check)
button.pack(pady=20)

root.mainloop()

```

3)

```

import os
import subprocess

# 1. Перейти в каталог PZ_11 и вывести список всех
# файлов в этом каталоге.
os.chdir("../PZ_11")
files_in_pz11 = [f for f in os.listdir() if os.path.isfile(f)]
print("Список файлов в PZ_11:", files_in_pz11)

"""
2. Перейти в корень проекта, создать папку test и в
ней папку test1. Переместить два файла из PZ_6 в test
и один файл из PZ_7 в test1. Переименовать файл из
PZ_7 в test.txt. Вывести размер файлов в папке test.
"""

# Переход в корень проекта
os.chdir("..")
# Создание папок test и test1

```

```

os.makedirs("test/test1", exist_ok=True)

# Перемещение файлов из PZ_6 в test
pz6_files = os.listdir("PZ_6")[:2]
for file in pz6_files:
    os.rename(os.path.join("PZ_6", file), os.path.join("test", file))

# Перемещение и переименование файла из PZ_7 в test1
pz7_file = os.listdir("PZ_7")[0]
os.rename(os.path.join("PZ_7", pz7_file), "test/test1/test.txt")

# Вывод информации о размере файлов в папке test
test_files = [f for f in os.listdir("test")
               if os.path.isfile(os.path.join("test", f))]
for file in test_files:
    print(f"Файл: {file}, Размер: "
          f"{os.path.getsize(os.path.join('test', file))} байт")

# 3. Перейти в папку PZ_11 и найти файл с самым
# коротким именем, имя вывести в консоль.
os.chdir("PZ_11")
shortest_file = min(files_in_pz11, key=len)
print("Файл с самым коротким именем в PZ_11:",
      os.path.basename(shortest_file))

"""
4. Перейти в любую папку, где есть отчет в формате .pdf
и «запустить» файл в привязанной к нему программе.
(Для примера, предположим, что PDF файл находится в
папке PZ_8)
"""
os.chdir("../PZ_8") # Переход в папку PZ_8
pdf_files = [f for f in os.listdir() if f.endswith(".pdf")]
if pdf_files:
    # Используем subprocess для запуска файла
    subprocess.run(["open", pdf_files[0]]) # Для macOS
    # subprocess.run(["xdg-open", pdf_files[0]]) # Для Linux

# 5. Удалить файл test.txt.
os.chdir('../test/test1')

os.remove('test.txt')
print("Файл test.txt был удален")

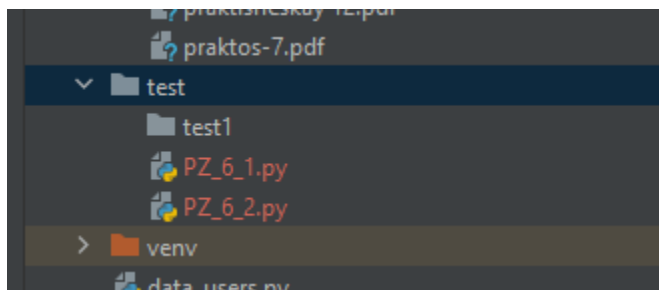
```

Протокол программы:

1)

2)

3)



Вывод: В данном практическом занятии я закрепил усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрести навыки составления программ с использованием GUI Tkinter в IDE PyCharm Community, изучить возможности модуля OS

