

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.

Задача 1.

Постановка задачи:

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).

Текст программы:

```
import tkinter as tk
from tkinter import ttk

def send_form():
    name = name_entry.get()
    email = email_entry.get()
    message = message_entry.get("1.0", "end-1c")
    inquiry = inquiry_var.get()

    if name and email and message and inquiry:
        print("Name:", name)
        print("Email:", email)
        print("Message:", message)
        print("Inquiry:", inquiry)
    else:
        status_label.config(text="Please fill all entries.", fg="red")

root = tk.Tk()
root.title("Contact Form")
root.geometry("400x300")

form_title_label = tk.Label(root, text="Contact Form", font=("Helvetica", 16, "bold"))
form_title_label.grid(row=0, column=0, columnspan=2, sticky="w")

separator = ttk.Separator(root, orient="horizontal")
separator.grid(row=2, column=0, columnspan=2, sticky="ew", pady=5)

name_label = tk.Label(root, text="Name:")
name_label.grid(row=3, column=0, sticky="w")
name_entry = tk.Entry(root)
name_entry.grid(row=3, column=1, pady=5)

email_label = tk.Label(root, text="Email:")
email_label.grid(row=4, column=0, sticky="w")
email_entry = tk.Entry(root)
email_entry.grid(row=4, column=1, pady=5)

message_label = tk.Label(root, text="Message:")
```

```

message_label.grid(row=5, column=0, sticky="w")
message_entry = tk.Text(root, height=5, width=30)
message_entry.grid(row=5, column=1, pady=5)

inquiry_label = tk.Label(root, text="Subject:")
inquiry_label.grid(row=6, column=0, sticky="w")
inquiry_var = tk.StringVar()
inquiry_dropdown = ttk.Combobox(root, textvariable=inquiry_var, values=["Product Inquiry",
"Product Inquiry2", "Product Inquiry3"])
inquiry_dropdown.grid(row=6, column=1, pady=5)

send_button = tk.Button(root, text="Send", command=send_form, bg="gray", fg="white")
send_button.grid(row=7, column=0, columnspan=2, pady=5)

status_label = tk.Label(root, text="Please fill all entries.")
status_label.grid(row=1, column=0, columnspan=2, pady=5, sticky="w")

root.mainloop()

```

Задача 2.

Постановка задачи:

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

Дано целое число А. Проверить истинность высказывания: «Число А является положительным».

Текст программы:

```

import tkinter as tk

def check_positive():
    try:
        number = int(entry.get())
        if number > 0:
            result_label.config(text="Число является положительным")
        else:
            result_label.config(text="Число не является положительным")
    except ValueError:
        result_label.config(text="Введите целое число")

root = tk.Tk()
root.title("Проверка числа на положительность")

label = tk.Label(root, text="Введите число:")
label.pack()

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

check_button = tk.Button(root, text="Проверить", command=check_positive)
check_button.pack()

result_label = tk.Label(root, text="")
result_label.pack()

root.mainloop()

```

Задача 3.

Постановка задачи:

Задание предполагает, что у студента есть проект с практическими работами (NoNo 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 os
import subprocess

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

os.chdir(r'/home/student/Документы/PycharmProjects/IS-22/Engibaryan/PZ-11')
files = [f for f in os.listdir() if os.path.isfile(f)]
print("Список файлов в PZ_11:")
for file in files:
    print(file)

# 2. 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.

os.chdir(r'/home/student/Документы/PycharmProjects/IS-22/Engibaryan')
os.makedirs('test/test1')
os.replace(r'PZ_6/PZ_6_1.py', 'test/PZ_6_1.py')
os.replace(r'PZ_6/PZ_6_2.py', 'test/PZ_6_2.py')
os.replace(r'PZ_7/PZ_7_1.py', 'test/test1/test.txt')

total_size = sum(os.path.getsize(os.path.join('test', f)) for f in os.listdir('test'))
print(f"Размер файлов в папке test: {total_size} байт")

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

os.chdir(r'/home/student/Документы/PycharmProjects/IS-22/Engibaryan/PZ-11')
files = os.listdir()

shortest_filename = None

for file in files:
```

```
if shortest_filename is None or len(file) < len(shortest_filename):
    shortest_filename = file

print('Самое короткое имя имеет файл:', os.path.basename(shortest_filename))

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

pdf_file = r'/home/student/Документы/PycharmProjects/IS-22/Engibaryan/PZ-16/PZ-16.pdf'
subprocess.call(['xdg-open', pdf_file])

# 5. удалить файл test.txt.

os.remove("test/test1/test.txt")
```

Вывод: я закрепила усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрела навыки составления программ с использованием GUI Tkinter в IDE PyCharm Community, изучила возможности модуля OS. Выполнены разработка кода, отладка, тестирование, оптимизация программного кода. Готовые программные коды выложены на GitHub.