

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 №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).

Тип алгоритма: циклический

Текст программы:

```
"""
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).

"""

import tkinter as tk
from tkinter import ttk

def submit_form():
    print("Форма отправлена")

def clear_form():
    username_entry.delete(0, tk.END)
    password_entry.delete(0, tk.END)
```

```
confirm_password_entry.delete(0, tk.END)
specialization_combobox.set('')
gender_var.set('')
for skill in skills_vars:
    skill.set(0)
additional_info_text.delete('1.0', tk.END)

root = tk.Tk()
root.title("Анкета Web-разработчика")

tk.Label(root, text="Регистрационное имя",
background="#BFBFBF").grid(row=0, column=0, sticky='w',
padx=3, pady=3)

username_entry = tk.Entry(root, relief="solid", bd=1)
username_entry.grid(row=0, column=1, padx=0, pady=0)

tk.Label(root, text="Пароль",
background="#BFBFBF").grid(row=1, column=0, sticky='w',
padx=3, pady=3)

password_entry = tk.Entry(root, show='*', relief="solid",
bd=1)
password_entry.grid(row=1, column=1, padx=0, pady=0)

tk.Label(root, text="Ваша специализация",
background="#BFBFBF").grid(row=3, column=0, sticky='w',
padx=3, pady=3)

specialization_combobox = ttk.Combobox(root,
values=["Web-мастер", "Программист", "Дизайнер"])
specialization_combobox.grid(row=3, column=1, padx=0,
pady=0)
```

```

tk.Label(root, text="Пол",
background="#BFBFBF").grid(row=4, column=0, sticky='w',
padx=3, pady=3)

gender_var = tk.StringVar()

tk.Radiobutton(root, text="М", variable=gender_var,
value="М").grid(row=4, column=1, sticky='w', padx=3,
pady=3)

tk.Radiobutton(root, text="Ж", variable=gender_var,
value="Ж").grid(row=4, column=1, padx=3, pady=3)


tk.Label(root, text="Ваши навыки",
background="#BFBFBF").grid(row=5, column=0, sticky='w',
padx=3, pady=3)

skills = ["знание HTML и CSS", "знание Perl", "знание
ASP", "знание Adobe Photoshop", "знание JAVA",
          "знание JavaScript", "знание Flash"]

skills_vars = [tk.IntVar() for _ in skills]

for i, skill in enumerate(skills):

    tk.Checkbutton(root, text=skill,
variable=skills_vars[i]).grid(row=5 + i, column=1,
sticky='w', padx=3, pady=3)


for i in range(6 + len(skills)):

    if i == 0 or i in range(3, 5) or i == 12:

        red_frame_0 = tk.Frame(root, background="#D1DAC9",
relief="solid", bd=1)

    else:

        red_frame_0 = tk.Frame(root, background="#D1DAC9")

    red_frame_0.grid(row=i, column=0, sticky='nsew')


    green_frame = tk.Frame(root, background="#BFBFBF",
relief="solid", bd=1)

    green_frame.grid(row=i, column=1, sticky='nsew')

```

```
frame_3 = tk.Frame(root, background="#BFBFBF")
frame_3.grid(row=i, column=3, sticky='nsew')

tk.Label(root, text="Регистрационное имя",
background="#D1DAC9").grid(row=0, column=0, sticky='w',
padx=3, pady=3)

username_entry = tk.Entry(root)
username_entry.grid(row=0, column=1, sticky='w', padx=5)

tk.Label(root, text="Пароль",
background="#D1DAC9").grid(row=1, column=0, sticky='w')
password_entry = tk.Entry(root, show='*')
password_entry.grid(row=1, column=1, sticky='w', padx=5)

tk.Label(root, text=":подтвердите пароль",
background="#BFBFBF").grid(row=2, column=1, sticky='e',
padx=(130, 3), pady=3)
confirm_password_entry = tk.Entry(root, show='*')
confirm_password_entry.grid(row=2, column=1, sticky='w',
padx=5)

tk.Label(root, text="Ваша специализация",
background="#D1DAC9").grid(row=3, column=0, sticky='w',
padx=3, pady=3)
specialization_combobox = ttk.Combobox(root,
values=["Web-мастер", "Программист", "Дизайнер"])
specialization_combobox.grid(row=3, column=1, padx=3,
pady=3, sticky='w')

tk.Label(root, text="Пол",
background="#D1DAC9").grid(row=4, column=0, sticky='w',
padx=3, pady=3)
```

```

gender_var = tk.StringVar()

tk.Radiobutton(root, text="М", variable=gender_var,
value="М", bg='#BFBFBF').grid(row=4, column=1,
sticky='w', padx=3)

tk.Radiobutton(root, text="Ж", variable=gender_var,
value="Ж", bg='#BFBFBF').grid(row=4, column=1, padx=(0,
130))

tk.Label(root, text="Ваши навыки",
background="#D1DAC9").grid(row=5, column=0, sticky='w',
padx=3, pady=3)

for i, skill in enumerate(skills):

    tk.Checkbutton(root, text=skill,
variable=skills_vars[i], bg="#BFBFBF").grid(row=5 + i,
column=1, sticky='w', padx=3, pady=3)

tk.Label(root, text="Дополнительные сведения\но себе",
background="#D1DAC9").grid(row=5 + len(skills), column=0,

sticky='w', padx=3, pady=3)

additional_info_text = tk.Text(root, width=30, height=4)
additional_info_text.grid(row=5 + len(skills), column=1,
columnspan=3, padx=3, pady=3, sticky='w')

button_frame = tk.Frame(root,
highlightbackground="black", highlightthickness=0)

button_frame.grid(row=6 + len(skills), column=0,
columnspan=1, sticky='ew')

submit_button = tk.Button(button_frame,
text="зарегистрировать", command=submit_form)

submit_button.grid(row=1, column=0, sticky='w', padx=5,
pady=5)

```

```

clear_button = tk.Button(button_frame, text="ОЧИСТИТЬ
форму", command=clear_form)

clear_button.grid(row=1, column=1, sticky='w', padx=5,
pady=5)

root.grid_columnconfigure(0, weight=1)
root.grid_columnconfigure(1, weight=1)
root.grid_columnconfigure(2, weight=1)
root.grid_columnconfigure(3, weight=1)
button_frame.grid_columnconfigure(0, weight=1)
button_frame.grid_columnconfigure(1, weight=1)
button_frame.grid_columnconfigure(2, weight=1)
button_frame.grid_columnconfigure(3, weight=1)

root.mainloop()

```

Протокол работы программы:

Process finished with exit code 0

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

Тип алгоритма: линейный

Текст программы:

```

"""
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_even_odd():
    try:
        A = int(entry.get())
        if A % 2 == 0:
            result = f"Число {A} является чётным"
        else:
            result = f"Число {A} является нечётным"
        messagebox.showinfo("Результат", result)
    except ValueError:
        messagebox.showerror("Ошибка", "Ошибка ввода. Введите целое число")

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

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

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

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

root.mainloop()
```

Протокол работы программы:

Process finished with exit code 0

Постановка задачи №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.

Тип алгоритма: циклический

Текст программы:

```
"""
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 PЗ6, а в папку test1 - один файл из PЗ7.

Файл из PЗ7 переименовать в test.txt. Вывести в консоль информацию о размере

файлов в папке test.

перейти в папку с PЗ11, 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.

"""

```
import os
```

```
import random
```

```
# I
```

```
os.chdir('../PЗ_11')
```

```
files = [f for f in os.listdir() if os.path.isfile(f)]
```

```
print("Список файлов в каталоге PЗ11:", files)
```

```
# II
```

```
os.chdir('../')
```

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

```
os.rename('PЗ_6/file.txt', 'test/file.txt')
```

```
os.rename('PЗ_6/file2.txt', 'test/file2.txt')
```

```
os.rename('PZ_7/file3.txt', 'test/test1/test.txt')

test_files = [f for f in os.listdir('test') if
os.path.isfile(os.path.join('test', f))]

for file in test_files:

    size = os.path.getsize(os.path.join('test', file))

    print(f"Размер файла {file}: {size} байт")

# III

os.chdir('PZ_11')

shortest_name = min([f for f in os.listdir() if
os.path.isfile(f)], key=len)

print("Файл с самым коротким именем:",
os.path.basename(shortest_name))

# IV

os.chdir('../reports')

pdf_file = [f for f in os.listdir() if
f.endswith('.pdf')]

os.startfile(pdf_file[random.randint(0,
len(pdf_file)-1)])

# V

os.chdir('../test/test1')

os.remove('test.txt')

print("Файл test.txt был удален")
```

Протокол работы программы:

Список файлов в каталоге PZ11: ['PZ_11.1.py', 'PZ_11.2.py', 'text18-3.txt']

Размер файла file.txt: 0 байт

Размер файла file2.txt: 0 байт

Файл с самым коротким именем: PZ_11.1.py

Файл test.txt был удален

Process finished with exit code 0

Вывод:

В процессе выполнения практического занятия №17, я закрепил усвоенные знания, понятия, алгоритмы, основные принципы составления программ с использованием GUI Tkinter в IDE PyCharm Community, изучение возможностей модуля OS.