

Практическое занятие № 17

Тема: составление программ с использованием GUI Tkinter в IDE PyCharm Community

Цели практического занятия: Цели практического занятия: закрепить усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрести навыки составления программ с 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: Линейный

Текст программы:

```
import tkinter as tk

root = tk.Tk()
root.title("Персональная информация")
text_info = tk.Label(root, text='Форма заявки на работу в зоопарке')
text_info.config(font=("Courier", 17))
text_info.grid(row=0, column=0)
text_sub_info = tk.Label(root, text='Пожалуйста, заполните форму. Обязательные поля помечены *')
text_sub_info.config(font=("Courier", 8))
text_sub_info.grid(row=1, column=0)
personal_frame = tk.LabelFrame(root, text="Персональная информация")
personal_frame.grid(row=2, column=0, padx=7, pady=7, sticky="nsew")
contact_frame = tk.LabelFrame(root, text="Контактная информация")
contact_frame.grid(row=3, column=0, padx=7, pady=7, sticky="nsew")
preferences_frame = tk.LabelFrame(root, text="Выберите ваших любимых животных:")
preferences_frame.grid(row=4, column=0, padx=7, pady=7, sticky="nsew")
name_label = tk.Label(personal_frame, text="Имя *")
name_label.grid(row=0, column=0, padx=7, pady=7)
name_entry = tk.Entry(personal_frame)
name_entry.grid(row=0, column=1, padx=7, pady=7)
age_label = tk.Label(personal_frame, text="Телефон")
age_label.grid(row=1, column=0, padx=7, pady=7)
age_entry = tk.Entry(personal_frame)
age_entry.grid(row=1, column=1, padx=7, pady=7)
email_label = tk.Label(personal_frame, text="Email *")
email_label.grid(row=2, column=0, padx=7, pady=7)
email_entry = tk.Entry(personal_frame)
email_entry.grid(row=2, column=1, padx=7, pady=7)
phone_label = tk.Label(contact_frame, text="Возврат *")
phone_label.grid(row=0, column=0, padx=7, pady=7, sticky="w")
phone_entry = tk.Entry(contact_frame)
phone_entry.grid(row=0, column=1, padx=7, pady=7, sticky="w")
gender_label = tk.Label(contact_frame, text="Пол")
gender_label.grid(row=1, column=0, padx=7, pady=7, sticky="w")
gender_options = "Мужской"
gender_options1 = "Женский"
gender_var = tk.StringVar()
gender_menu = tk.OptionMenu(contact_frame, gender_var, gender_options, gender_options1)
gender_menu.grid(row=1, column=1)
phone_label1 = tk.Label(contact_frame, text="Перечислите личные качества")
phone_label1.grid(row=2, column=0, padx=7, pady=7, sticky="w")
phone_entry = tk.Text(contact_frame, height=4, width=25)
phone_entry.grid(row=3, column=0, padx=7, pady=7)
```

```
animals_frame = tk.Frame(preferences_frame)
animals_frame.grid(row=0, column=0)
zeb_check = tk.Checkbutton(animals_frame, text="Зебра")
slon_check = tk.Checkbutton(animals_frame, text="Слон")
cat_check = tk.Checkbutton(animals_frame, text="Кошак")
anti_check = tk.Checkbutton(animals_frame, text="Антилопа")
anak_check = tk.Checkbutton(animals_frame, text="Анаконда")
bird_check = tk.Checkbutton(animals_frame, text="Голубь")
hum_check = tk.Checkbutton(animals_frame, text="Человек")
crab_check = tk.Checkbutton(animals_frame, text="Краб")
zeb_check.grid(row=0, column=0)
slon_check.grid(row=1, column=0)
cat_check.grid(row=0, column=1)
anti_check.grid(row=1, column=1)
anak_check.grid(row=0, column=2)
bird_check.grid(row=1, column=2)
hum_check.grid(row=0, column=3)
crab_check.grid(row=1, column=3)
submit_button = tk.Button(root, text="Отправить информацию")
submit_button.grid(row=5, column=0, pady=1, sticky='w')
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:

```
import tkinter as tk
from random import randint

symbols_list = ['a', 'b', 'c', 'd', 'e', 'f', 'j']
def any40s():
    try:
        word = len(symbols_list) - 1
        result = []
        for i in range(40):
            a = randint(0, word)
            res = symbols_list[a]
            result.append(res)
        return result
    except:
        return "Произошла ошибка!"
def show_result():
    result_list = any40s()
    result_text.delete("1.0", "end")
    result_text.insert("1.0", ''.join(result_list))
root = tk.Tk()
root.title("40 любых символов")
personal_frame = tk.LabelFrame(root, text="40 рандомных символов")
personal_frame.grid(row=0, column=0, padx=5, pady=5, sticky="nsew")
result_text = tk.Text(personal_frame, height=3, width=30)
result_text.grid(row=0, column=0)
button = tk.Button(root, text="Сгенерировать", command=show_result)
button.grid(row=1, column=0)
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: Линейный

Текст программы:

```
import os
# task 1
print(os.listdir('./PZ_11'))
# task 2
try:
    os.mkdir('./test')
    os.mkdir('./test/test1')
except:
    pass

# task 3
os.rename('./PZ_6/1.txt', './test/1.txt')
os.rename('./PZ_6/2.txt', './test/2.txt')
os.rename('./PZ_7/3.txt', './test/test1/4.txt')
# task 4
path = os.listdir('./test')
for i in path:
    print((os.path.getsize(f'./test/{i}')), 'bytes')
# task 5
path = os.listdir('./PZ_11')
path = min(path, key=len)
print(os.path.basename(f'./PZ_11/{path}'))
# task 6
os.startfile('Reports\Report_2.pdf')

# task 7
open('PZ_17/123.txt', 'w')
os.remove('PZ_17/123.txt')
```

Протокол работы программы:

```
['1.txt', '11.txt', 'PZ_11_1.py', 'PZ_11_2.py', 'text18-2.txt', '__init__.py']
```

```
0 bytes
```

```
1.txt
```

```
Process finished with exit code 0
```

Вывод: в процессе выполнения практического занятия выработал навыки составления программ с GUI Tkinter в IDE PyCharm Community, изучил возможности модуля OS. Были использованы языковые конструкции open, print, for.

Выполнены разработка кода, отладка, тестирование, оптимизация программного кода.

Готовые программные коды выложены на GitHub.