

Практическое занятие № 17

Тема: составление программ с использованием GUI Tkinter в IDE PyCharm Community, изучение возможностей модуля OS.

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:

```
from tkinter import *
from tkinter import ttk

def delete():
    regname_entry.delete(0, 'end')
    passw_entry.delete(0, 'end')
    passww_entry.delete(0, 'end')
    spec_entry.delete(0, 'end')
    skills2.delete(1.0, 2.0)
    HTMSCSS.deselect()
    Perl.deselect()
    ASP.deselect()
    Photoshop.deselect()
    JAVA.deselect()
    JavaScript.deselect()
    Flash.deselect()

def reg():
    g = 'Ж' if r_var.get() else 'М'

    with open('user.txt', 'w+') as f:
        f.write('Name: '+regname_entry.get()+'\n')
        f.write('Password: '+passw_entry.get()+'\n')
        f.write('Specialization: '+spec_entry.get()+'\n')
        f.write('Gender: '+g+'\n')
        f.write('HTML & CSS: '+str(var1.get())+'\n')
        f.write('Perl: '+str(var2.get())+'\n')
        f.write('ASP: '+str(var3.get())+'\n')
        f.write('Photoshop: '+str(var4.get())+'\n')
        f.write('JAVA: '+str(var5.get())+'\n')
```

```

f.write('JavaScript: '+str(var6.get())+'\n')
f.write('Flash: '+str(var7.get())+'\n')
f.write('Additional info: '+skills2.get(1.0, 'end')+'\n')

root = Tk()

root.title('Анкета Web-разработчика')
root.geometry('510x450')
root.resizable(width=False, height=False)

regname = Label(root, text='Регистрационное имя', font=('Times New Roman',
10)).grid(row=1, column=0, sticky='w')
regname_entry = Entry(root)
regname_entry.grid(row=1, column=1, padx=5, pady=5, sticky='e')

passw = Label(root, text='Пароль', font=('Times New Roman', 10)).grid(row=2,
column=0, sticky='w')
passww = Label(root, text='подтвердите пароль', font=('Times New Roman',
10)).grid(row=3, column=2, sticky='w')
passw_entry = Entry(root, show="•")
passw_entry.grid(row=2, column=1, padx=5, sticky='e')
passww_entry = Entry(root, show="•")
passww_entry.grid(row=3, column=1, padx=5, sticky='e')

spec = Label(root, text='Ваша специализация', font=('Times New Roman',
10)).grid(row=4, column=0, sticky='w')
www = ['Web-мастер', 'Web-ломастер']
spec_entry = ttk.Combobox(values=www)
spec_entry.grid(row=4, column=1, padx=5, pady=5, sticky='e')

gender = Label(root, text='Пол', font=('Times New Roman', 10)).grid(row=5, column=0,
sticky='w')
r_var = BooleanVar()
r_var.set(False)
radioM = Radiobutton(root, text='Ж', variable=r_var, value=True)
radioM.grid(row=5, column=1, padx=51, pady=5, sticky='w')
radioW = Radiobutton(root, text='М', variable=r_var, value=False)
radioW.grid(row=5, column=1, padx=5, pady=5, sticky='w')

var1, var2, var3, var4, var5, var6, var7 = BooleanVar(), BooleanVar(), BooleanVar(),
BooleanVar(), BooleanVar(), BooleanVar(), BooleanVar()
skills = Label(root, text='Ваши навыки', font=('Times New Roman', 10))
skills.grid(row=6, column=0, sticky='w')
HTMSSCS = Checkbutton(root, text='знание HTML и CSS', font=('Times New Roman', 10),
variable=var1)
HTMSSCS.grid(row=6, column=1, padx=5, sticky='w')
Perl = Checkbutton(root, text='знание Perl', font=('Times New Roman', 10),
variable=var2)
Perl.grid(row=7, column=1, padx=5, sticky='w')
ASP = Checkbutton(root, text='знание ASP', font=('Times New Roman', 10),
variable=var3)
ASP.grid(row=8, column=1, padx=5, sticky='w')
Photoshop = Checkbutton(root, text='знание Adobe Photoshop', font=('Times New Roman',
10), variable=var4)

```

```
Photoshop.grid(row=9, column=1, padx=5, sticky='w')
JAVA = Checkbutton(root, text='знание JAVA', font=('Times New Roman', 10),
variable=var5)
JAVA.grid(row=10, column=1, padx=5, sticky='w')
JavaScript = Checkbutton(root, text='знание JavaScript', font=('Times New Roman',
10), variable=var6)
JavaScript.grid(row=11, column=1, padx=5, sticky='w')
Flash = Checkbutton(root, text='знание Flash', font=('Times New Roman', 10),
variable=var7)
Flash.grid(row=12, column=1, padx=5, sticky='w')

skills = Label(root, text='Дополнительные', font=('Times New Roman', 10))
skills.grid(row=13, column=0, sticky='nw')
skills = Label(root, text='сведения о себе', font=('Times New Roman', 10))
skills.grid(row=13, column=0, sticky='w')

skills2 = Text(root, height=4, width=25)
skills2.grid(row=13, column=1, sticky='w', padx=5)

zareg = Button(text='зарегистрировать', command=reg)
zareg.grid(row=14, column=0, padx=10, pady=10, sticky='ns')
clear = Button(text='очистить форму', command=delete)
clear.grid(row=14, column=1, padx=10, pady=10, sticky='w')

if __name__ == '__main__':
    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:

```
from tkinter import *

def check_string():
    stroka = entry.get().lower()
    try:
        for i in range(len(stroka)-1):
            if ord(stroka[i]) > ord(stroka[i+1]):
                result.set('Символ, нарушающий порядок: '+str(i+1))
                return
        result.set('0')
    except ValueError:
        result.set('Неправильный ввод!')

root = Tk()
root.title("Порядок")
root.geometry('400x300')

prog = Label(root, text='Программа для проверки алфавитного порядка латиницы')
prog.pack()

entry = Entry(root)
entry.pack()

result = StringVar()
result_label = Label(root, textvariable=result)
result_label.pack()

button = Button(root, text='Проверить', command=check_string)
button.pack()

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

# Задание 1
os.chdir('..\PZ_11')
files = [f for f in os.listdir('.') if os.path.isfile(f)]
print(files)

# Задание 2
os.chdir('..')
os.makedirs('test/test1')
os.rename('PZ_6/PZ_6_3_1.py', 'test/PZ_6_3_1.py')
os.rename('PZ_6/PZ_6_3_2.py', 'test/PZ_6_3_2.py')
os.rename('PZ_7/PZ_7_3_1.py', 'test/test1/PZ_7_3_1.py')
os.rename('test/test1/PZ_7_3_1.py', 'test/test1/test.txt')

for file in os.listdir('test'):
    print(f"{file}: {os.path.getsize(os.path.join('test', file))} bytes")

# Задание 3
os.chdir('PZ_11')
files = os.listdir('.')
shortest_file = min(files, key=len)
print(os.path.basename(shortest_file))
os.chdir('..')
```

```
# Задание 4
os.chdir("..\reports")
os.startfile('PZ_17.pdf')
os.chdir('..')

# Задание 5
# os.remove("..\test\test1\test.txt")
```

Протокол работы программы:

['file.txt', 'itog.txt', 'PZ_11_3_1.py', 'PZ_11_3_2.py', 'text18-3.txt', 'text2.txt']
PZ_6_3_1.py: 532 bytes
PZ_6_3_2.py: 700 bytes
test1: 0 bytes
file.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.