

Практическая работа №12.

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

Цель: закрепить усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрести навыки составления программ с использованием GUI Tkinter в IDE PyCharm Community.

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).

Анкета Web-разработчика	
Регистрационное имя	<input type="text"/>
Пароль	<input type="password"/> <input type="password"/> : подтвердите пароль
Ваша специализация	Web-мастер ▾
Пол	М <input checked="" type="radio"/> Ж <input type="radio"/>
Ваши навыки	<input type="checkbox"/> знание HTML и CSS <input type="checkbox"/> знание Perl <input type="checkbox"/> знание ASP <input type="checkbox"/> знание Adobe Photoshop <input type="checkbox"/> знание JAVA <input type="checkbox"/> знание JavaScript <input type="checkbox"/> знание Flash
Дополнительные сведения о себе	<input type="text"/>
<input type="button" value="зарегистрировать"/> <input type="button" value="очистить форму"/>	

Задание 2. Разработать программу с применением пакета tk, взяв в качестве условия одну любую задачу из ПЗ №№ 3 – 8.

Тип алгоритма: циклический и ветвление.

Текст программы:

Задача 1:

```
# Реализовать прототип в IDE PyCharm Community с применением пакета tk.  
# Получить интерфейс максимально приближенный к оригиналу.
```

```
from tkinter import *  
from tkinter import ttk
```

```
special_list = ['Front-End', 'Back-End', 'Full-Stack'] # Создание списка  
специальностей для combobox.
```

```
def ClearForm(): # Функция-обработчик очистки формы.  
    name.delete('0', END)  
    password.delete('0', END)  
    double_password.delete('0', END)  
    special.delete('0', END)  
    skill_bar1.deselect()  
    skill_bar2.deselect()  
    skill_bar3.deselect()  
    skill_bar4.deselect()  
    skill_bar5.deselect()  
    skill_bar6.deselect()  
    skill_bar7.deselect()  
    genderM.deselect()  
    genderF.deselect()  
    inf_txt.delete("1.0", END)
```

```
root = Tk() # Инициализация корневого окна приложения.  
root.title("Анкета Web-Разработчика")  
root.geometry('690x540')
```

```
label = Label(text='Анкета Web-Разработчика', font=('Times New Roman', 20))  
label.grid(row=0, column=0, sticky=N, pady=7)  
var = IntVar()
```

```
# Создание фреймов для возможности стилизовать сетку между разделами  
приложения.
```

```
frame1 = LabelFrame(master=root, relief=RAISED, borderwidth=1,  
background='#B0E0E6')  
frame1.grid(row=1, column=0, sticky=W + E + N + S)  
frame2 = LabelFrame(master=root, relief=RAISED, borderwidth=1,  
background='#C0C0C0')  
frame2.grid(row=1, column=1, sticky=W + E + N + S)  
frame3 = LabelFrame(master=root, relief=RAISED, borderwidth=1,  
background='#B0E0E6')  
frame3.grid(row=2, column=0, sticky=W + E + N + S)
```

```

frame4 = LabelFrame(master=root, relief=RAISED, borderwidth=1,
background='#C0C0C0')
frame4.grid(row=2, column=1, sticky=N + S + W)
frame5 = LabelFrame(master=root, relief=RAISED, borderwidth=1,
background='#B0E0E6')
frame5.grid(row=3, column=0, sticky=W + E + N + S)
frame6 = LabelFrame(master=root, relief=RAISED, borderwidth=1,
background='#C0C0C0')
frame6.grid(row=3, column=1, sticky=W + E + N + S)
frame7 = LabelFrame(master=root, relief=RAISED, borderwidth=1,
background='#B0E0E6')
frame7.grid(row=4, column=0, sticky=W + E + N + S)
frame8 = LabelFrame(master=root, relief=RAISED, borderwidth=1,
background='#C0C0C0')
frame8.grid(row=4, column=1, sticky=W + E + N + S)
frame9 = LabelFrame(master=root, relief=RAISED, borderwidth=1,
background='#B0E0E6')
frame9.grid(row=5, column=0, sticky=W + E + N + S)
frame10 = LabelFrame(master=root, relief=RAISED, borderwidth=1,
background='#C0C0C0')
frame10.grid(row=5, column=1, sticky=W + E + N + S)
frame11 = LabelFrame(master=root, relief=RAISED, borderwidth=1,
background='#B0E0E6')
frame11.grid(row=6, column=0, sticky=W + E + N + S)
frame12 = LabelFrame(master=root, relief=RAISED, borderwidth=1,
background='#C0C0C0')
frame12.grid(row=6, column=1, sticky=W + E + N + S)

# Добавление поля ввода регистрационного имени пользователя и подписи к нему.
label1 = Label(master=frame1, text='Регистрационное имя', font=('Times New
Roman', 15), background='#B0E0E6')
label1.grid(row=1, column=0, sticky=W + E)

name = Entry(master=frame2, width=40, relief=RAISED, borderwidth=1,
background='#C0C0C0')
name.grid(row=1, column=1, sticky=W + E + N + S, pady=5, padx=5)

# Добавление поля ввода пароля и его подтверждения.

label2 = Label(master=frame3, text='Пароль', font=('Times New Roman', 15),
background='#B0E0E6')
label2.grid(row=2, column=0, sticky=W + N + S + E)

password = Entry(master=frame4, width=40, relief=RAISED, borderwidth=1,
background='#C0C0C0')
password.grid(row=3, column=1, sticky=W, padx=5, pady=5)
double_password = Entry(master=frame4, width=40, relief=RAISED,
background='#C0C0C0')
double_password.grid(rowspan=1, column=1, pady=5, padx=5, sticky=W)

lable_pass = Label(master=frame4, text=':Подтвердите пароль', font=('Times New
Roman', 10), background='#C0C0C0')
lable_pass.grid(row=4, column=2, sticky=E)

# Добавление виджета выбора с выплывающим меню combobox (выбор специализации).

```

```

spec_label = Label(master=frame5, text='Ваша специализация', font=('Times New
Roman', 15), background='#B0E0E6')
spec_label.grid(row=3, column=0, sticky=W)

special = ttk.Combobox(values=special_list, background='#C0C0C0')
special.grid(row=3, column=1, sticky=W, padx=5)

# Добавление виджета RadioButton для выбора пола.

gender_lab = Label(master=frame7, text='Пол', font=('Times New Roman', 15),
background='#B0E0E6')
gender_lab.grid(row=4, column=0, sticky=W)

genderM = Radiobutton(frame8, text='М', variable=var, value='m',
background='#C0C0C0')
genderF = Radiobutton(frame8, text='Ж', variable=var, value='f',
background='#C0C0C0')

genderM.grid(row=4, column=1)
genderF.grid(row=4, column=2)

# Добавление виджета CheckButton для указания личных навыков.

skills = Label(master=frame9, text='Ваши навыки', font=('Times New Roman', 15),
background='#B0E0E6')
skills.grid(row=5, column=0, sticky=W)

skill_bar1 = Checkbutton(text='Знание HTML и CSS', background='#C0C0C0')
skill_bar1.grid(in_=frame10, row=5, column=1, sticky=W, pady=1)
skill_bar2 = Checkbutton(text='Знание Perl', background='#C0C0C0')
skill_bar2.grid(in_=frame10, row=6, column=1, sticky=W, pady=1)
skill_bar3 = Checkbutton(text='Знание ASP', background='#C0C0C0')
skill_bar3.grid(in_=frame10, row=7, column=1, sticky=W, pady=1)
skill_bar4 = Checkbutton(text='Знание Adobe Photoshop', background='#C0C0C0')
skill_bar4.grid(in_=frame10, row=8, column=1, sticky=W, pady=1)
skill_bar5 = Checkbutton(text='Знание JAVA', background='#C0C0C0')
skill_bar5.grid(in_=frame10, row=9, column=1, sticky=W, pady=1)
skill_bar6 = Checkbutton(text='Знание JavaScript', background='#C0C0C0')
skill_bar6.grid(in_=frame10, row=10, column=1, sticky=W, pady=1)
skill_bar7 = Checkbutton(text='Знание Flash', background='#C0C0C0')
skill_bar7.grid(in_=frame10, row=11, column=1, sticky=W, pady=1)

# Добавление большого поля ввода с возможностью прокрутки по оси Y.

more_info = Label(master=frame11, text='Дополнительные сведения о себе',
font=('Times New Roman', 15),
background='#B0E0E6')
more_info.grid(row=6, column=0, sticky=W)

inf_txt = Text(height=5, width=43, font=('Times New Roman', 12), wrap=WORD,
background='#C0C0C0')
inf_txt.grid(pady=5, padx=5, row=6, column=1, sticky=W)

scroll = Scrollbar(command=inf_txt.yview, background='#C0C0C0')
scroll.grid(row=6, column=1, sticky=E + N + S, pady=5, padx=5)
inf_txt.config(yscrollcommand=scroll.set)

```

```
# Добавление кнопок завершения регистрации и очистки формы.

log_button = Button(text='Зарегистрировать')
clear_button = Button(text='Очистить форму', command=ClearForm)

log_button.grid(pady=7, row=7, column=0, sticky=W)
clear_button.grid(pady=7, row=7, column=0, sticky=E)

root.mainloop()
```

Задача 2:

```
# Размер скидки на продукты определен следующим образом: при покупке до 500 р.
скидка
# составит 2%; при покупке от 500 р. до 1000 р. скидка составит 3%; при покупке
от 1000 р.
# до 1500 р. скидка составит 4%; при покупке от 1500 р. до 2000 р. скидка
составит 5%.
# Составить программу определяющую размер скидки в зависимости от потраченной
суммы.
# Реализовать данную задачу приложением с применением библиотеки tkinter.
```

```
from tkinter import *
from tkinter import ttk
```

```
# Написание функции-обработчика: вывод полученных данных производится путём
изменения скрытого текстового поля.
```

```
def disk_t():
    a = summa.get()
# Обработка исключений
# (возникла ошибка ввиду невозможности корректного прерывания функции в случае
ввода неправильных данных)
# Решение данной проблемы будет представлено ниже.
    try:
        a = int(a)
    except ValueError:
        sale_label.configure(text='Некорректная форма записи', fg='red')
        accept_button.configure(command=disk_t)

def sale(k, s): # Составление функции для решения задачи.
    p = s / 100
    t = k * p
    sale_label.configure(text=f'Конечная стоимость: {k - t} рублей.',
fg='black')
# Условием повторно проверяется правильность введённых данных, в противном
случае в поле вывода выбивается ошибка.
    if type(a) != int:
        sale_label.configure(text=f'Некорректная форма записи', fg='red')
```

```

elif a < 0:
    sale_label.configure(text=f'Число не должно быть отрицательным',
fg='red')

elif 0 <= a < 500:
    sale(a, 2)

elif 500 <= a < 1000:
    sale(a, 3)

elif 1000 <= a < 1500:
    sale(a, 4)

elif 1500 <= a < 2000:
    sale(a, 5)

else:
    sale(a, 10)

# Инициализация корневого окна приложения.

root = Tk()
root.title('Калькулятор скидки')
root.geometry('390x190')
root.configure(bg='#FFF8DC')

main_frame = LabelFrame(root, text='Калькулятор скидки', bd=5, bg='#FFF8DC')
main_frame.grid(pady=5, padx=5)

# Добавление подписи к полю ввода с указанием единиц измерения.

main_label = Label(master=main_frame, text='Введите стоимость покупки:',
font=('Arial', 20), bg='#FFF8DC')
main_label.grid(row=0, columnspan=2, sticky=W + E, pady=5)

meas_system = Label(master=main_frame, text='Руб.', font=('Arial', 14),
bg='#FFF8DC')
meas_system.grid(row=1, column=1, sticky=W)

# Добавление кнопки отправления данных в функцию-обработчик.

accept_button = Button(master=main_frame, text='Принять', font=('Arial', 10),
command=disk_t, bg='F4A460', bd=1,
relief=FLAT, activebackground="#8B4513")
accept_button.grid(row=2, columnspan=2, sticky=E + W, pady=7)

# Добавление поля ввода необходимых данных

summa = Entry(master=main_frame, width='20', font=('Arial', 14), bg='FFDEAD')
summa.grid(row=1, column=0, sticky=E)

# Добавление изменяемого текстового поля-вывода.

sale_label = Label(main_frame, font=('Arial', 15), bg='FFDEAD')
sale_label.grid(row=3, columnspan=2, sticky=E + W)

```

```
root.mainloop()
```

Протокол работы программы:

Задача 1:

Анкета Web-Разработчика

Регистрационное имя

Пароль

Подтвердите пароль

Ваша специализация

Пол

☐ М ☐ Ж

Ваши навыки

☐ Знание HTML и CSS

☐ Знание Perl

☐ Знание ASP

☐ Знание Adobe Photoshop

☐ Знание JAVA

☐ Знание JavaScript

☐ Знание Flash

Дополнительные сведения о себе

Зарегистрировать

Очистить форму

Задача 2:

Калькулятор скидки

Калькулятор скидки

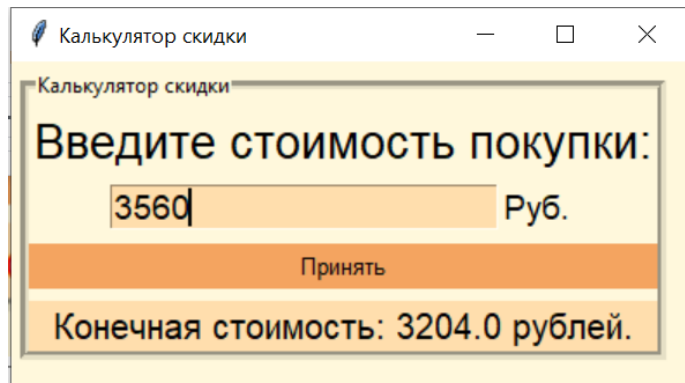
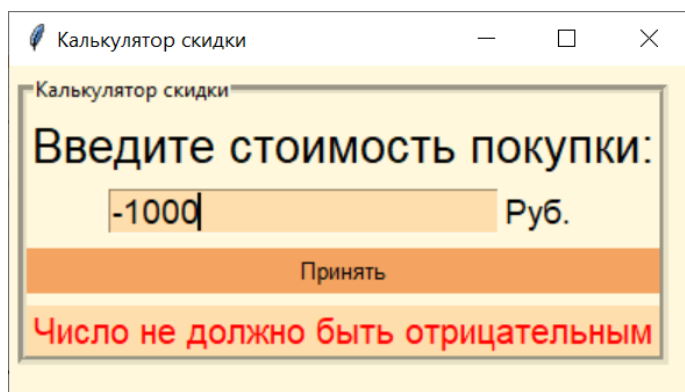
Введите стоимость покупки:

asdsadas

Руб.

Принять

Некорректная форма записи



Вывод: в процессе выполнения практического занятия закрепил усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрел навыки составления программ с использованием GUI Tkinter в IDE PyCharm Community.