

## Практическое занятие № 17

**Тема:** Работа с GUI Tkinter в IDE PyCharm Community.

**Цель:** закрепить усвоенные знания работы с GUI Tkinter, основными принципами составления программ, приобрести навыки составления программ с GUI Tkinter в IDE PyCharm Community.

### 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  
'''
```

The image shows a web form titled "EVENT REGISTRATION FORM". The form contains the following fields and elements:

- Name:** Two input fields labeled "First Name" and "Last Name".
- Company:** A single wide input field.
- Email:** A single wide input field.
- Phone:** Two input fields labeled "Area Code" and "Phone Number".
- Subject:** A dropdown menu with the text "Choose option" and a downward arrow.
- Are you an existing customer?:** Two radio buttons, one labeled "Yes" (which is selected) and one labeled "No".
- REGISTER:** A red button with white text.

### Текст программы:

```
import tkinter as tk
from tkinter import ttk

def register():
    print("Registration submitted")

root = tk.Tk()
root.title("Registration Form")
root.geometry("600x300") # Устанавливаем размер окна

# Name Section
name_label = ttk.Label(root, text="Name")
name_label.grid(row=0, column=0, padx=5, pady=5, sticky="W")
last_name_entry = ttk.Entry(root)
last_name_entry.grid(row=0, column=1, padx=5, pady=5)
first_name_entry = ttk.Entry(root)
first_name_entry.grid(row=0, column=2, padx=5, pady=5)
label = ttk.Label(root, text="first name", font=("Arial", 7))
label.grid(row=1, column=1, padx=15, pady=1, sticky="W")
label = ttk.Label(root, text="last name", font=("Arial", 7))
label.grid(row=1, column=2, padx=10, pady=1, sticky="W")

# Company Section
company_label = ttk.Label(root, text="Company")
company_label.grid(row=2, column=0, padx=5, pady=5, sticky="W")
company_entry = ttk.Entry(root)
company_entry.grid(row=2, column=1, padx=5, pady=5)

# Email Section
email_label = ttk.Label(root, text="Email")
email_label.grid(row=3, column=0, padx=5, pady=5, sticky="W")
email_entry = ttk.Entry(root)
email_entry.grid(row=3, column=1, padx=5, pady=5)

# Phone Section
phone_label = ttk.Label(root, text="Phone")
phone_label.grid(row=4, column=0, padx=5, pady=5, sticky="W")
area_code_entry = ttk.Entry(root,)
area_code_entry.grid(row=4, column=1, padx=5, pady=5)
phone_number_entry = ttk.Entry(root)
phone_number_entry.grid(row=4, column=2, padx=5, pady=5)
label = ttk.Label(root, text="Area Code", font=("Arial", 7))
label.grid(row=5, column=1, padx=15, pady=1, sticky="W")
label = ttk.Label(root, text="Phone Number", font=("Arial", 7))
label.grid(row=5, column=2, padx=10, pady=1, sticky="W")

# Subject Section
subject_label = ttk.Label(root, text="Subject")
subject_label.grid(row=6, column=0, padx=5, pady=5, sticky="W")
```

```

subject_combobox = ttk.Combobox(root, values=["Choose option",
"Feedback", "Support"])
subject_combobox.set('Choose option')
subject_combobox.grid(row=6, column=1, padx=5, pady=5)

# Survey Section
survey_label = ttk.Label(root, text="Are you an existing
customer?")
survey_label.grid(row=7, column=0, sticky="W")

survey_var = tk.StringVar()
survey_var.set("Yes")

yes_radio = ttk.Radiobutton(root, text="Yes", variable=survey_var,
value="Yes")
yes_radio.grid(row=8, column=0, padx=5, pady=5, sticky="W")

no_radio = ttk.Radiobutton(root, text="No", variable=survey_var,
value="No")
no_radio.grid(row=8, column=0, padx=100, pady=5, sticky="W")

# Register Button
style = ttk.Style()
style.configure("Red.TButton", foreground="black",
background="red")

register_button = ttk.Button(root, text="Register",
command=register, style="Red.TButton")
register_button.grid(row=9, column=0, padx=40, pady=5, sticky="W")

root.mainloop()

```

## Протокол работы программы:

**Registration Form**

Name:  first name  last name

Company:

Email:

Phone:  Area Code  Phone Number

Subject:

Are you an existing customer?

☒ Yes ☐ No

## Постановка задачи 2.

```
'''
Дана масса M в килограммах. Используя операцию деления нацело,
найти количество полных тонн в ней (1 тонна = 1000 кг). Решить
задачу с использованием tkinter
'''
```

### Текст программы:

```
import tkinter as tk
from tkinter import messagebox

def calculate_tonnes():
    try:
        mass_kg = float(entry.get())
        tonnes = mass_kg // 1000 # Используем операцию деления
нацело
        result_label.config(text=f"Полных тонн: {tonnes}")
    except ValueError:
        messagebox.showerror("Ошибка", "Пожалуйста, введите
корректное значение для массы")

# Создаем графический интерфейс
root = tk.Tk()
root.title("Конвертер массы в тонны")

input_label = tk.Label(root, text="Введите массу в килограммах:")
input_label.pack()

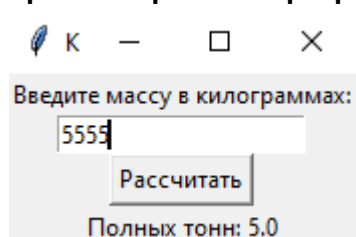
entry = tk.Entry(root)
entry.pack()

calculate_button = tk.Button(root, text="Рассчитать",
command=calculate_tonnes)
calculate_button.pack()

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

root.mainloop()
```

### Протокол работы программы:



Вывод: в процессе выполнения практического занятия выработал навыки работы с GUI Tkinter в IDE PyCharm Community.

Выполнены разработка кода, отладка, тестирование, оптимизация программного кода. Готовые программные коды выложены на GitHub.