

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 1 задачи:

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

Тип алгоритма: линейный

Текст программы:

```
import tkinter as tk
from tkinter import ttk, Button
```

```
def create_account():
    print("Account Created")
```

```
app = tk.Tk()
app.title("Buat Akun Google Anda")
```

```
def create_labeled_entry(frame, text, row, column, show=None, span=1):
    label = tk.Label(frame, text=text)
    label.grid(row=row, column=column, sticky=tk.W, pady=2)
    entry = tk.Entry(frame, show=show)
    entry.grid(row=row, column=column + 1, pady=2, columnspan=span, sticky=(tk.W, tk.E))
    return entry
```

```
frame = ttk.Frame(app, padding="20 20 20 20")
frame.grid(row=0, column=0, sticky=(tk.W, tk.E, tk.N, tk.S))
```

```
first_name_entry = create_labeled_entry(frame, "Nama Depan", 0, 0)
last_name_entry = create_labeled_entry(frame, "Nama Belakang", 0, 2)
```

```
username_label = tk.Label(frame, text="Pilih nama pengguna Anda")
username_label.grid(row=1, column=0, sticky=tk.W, pady=2)
username_entry = tk.Entry(frame)
username_entry.grid(row=1, column=1, pady=2, columnspan=3, sticky=(tk.W, tk.E))
```

```
email_pref_label = tk.Label(frame, text="Saya lebih suka menggunakan alamat email saya yang sekarang", fg="blue")
email_pref_label.grid(row=2, column=1, pady=2, columnspan=3, sticky=tk.W)
```

```

password_entry = create_labeled_entry(frame, "Buat sandi", 3, 0, show="*")
confirm_password_entry = create_labeled_entry(frame, "Konfirmasi sandi Anda", 4, 0, show="*")

dob_label = tk.Label(frame, text="Tanggal lahir")
dob_label.grid(row=5, column=0, sticky=tk.W, pady=2)

dob_year = ttk.Combobox(frame, values=list(range(1900, 2024)))
dob_year.grid(row=5, column=1, pady=2)

dob_month = ttk.Combobox(frame, values=[str(i).zfill(2) for i in range(1, 13)])
dob_month.grid(row=5, column=2, pady=2)

dob_day = ttk.Combobox(frame, values=[str(i).zfill(2) for i in range(1, 32)])
dob_day.grid(row=5, column=3, pady=2)

gender_label = tk.Label(frame, text="Gender")
gender_label.grid(row=6, column=0, sticky=tk.W, pady=2)

gender = ttk.Combobox(frame, values=["Saya...", "Pria", "Wanita", "Lainnya"])
gender.grid(row=6, column=1, pady=2, columnspan=3, sticky=(tk.W, tk.E))

phone_label = tk.Label(frame, text="Ponsel")
phone_label.grid(row=7, column=0, sticky=tk.W, pady=2)

phone_code_label = tk.Label(frame, text="+62")
phone_code_label.grid(row=7, column=1, sticky=tk.W, pady=2)
phone_entry = tk.Entry(frame)
phone_entry.grid(row=7, column=2, pady=2, columnspan=2, sticky=(tk.W, tk.E))

current_email_label = tk.Label(frame, text="Alamat email Anda saat ini")
current_email_label.grid(row=8, column=0, sticky=tk.W, pady=2)
current_email_entry = tk.Entry(frame)
current_email_entry.grid(row=8, column=1, pady=2, columnspan=3, sticky=(tk.W, tk.E))

location_label = tk.Label(frame, text="Lokasi")
location_label.grid(row=9, column=0, sticky=tk.W, pady=2)

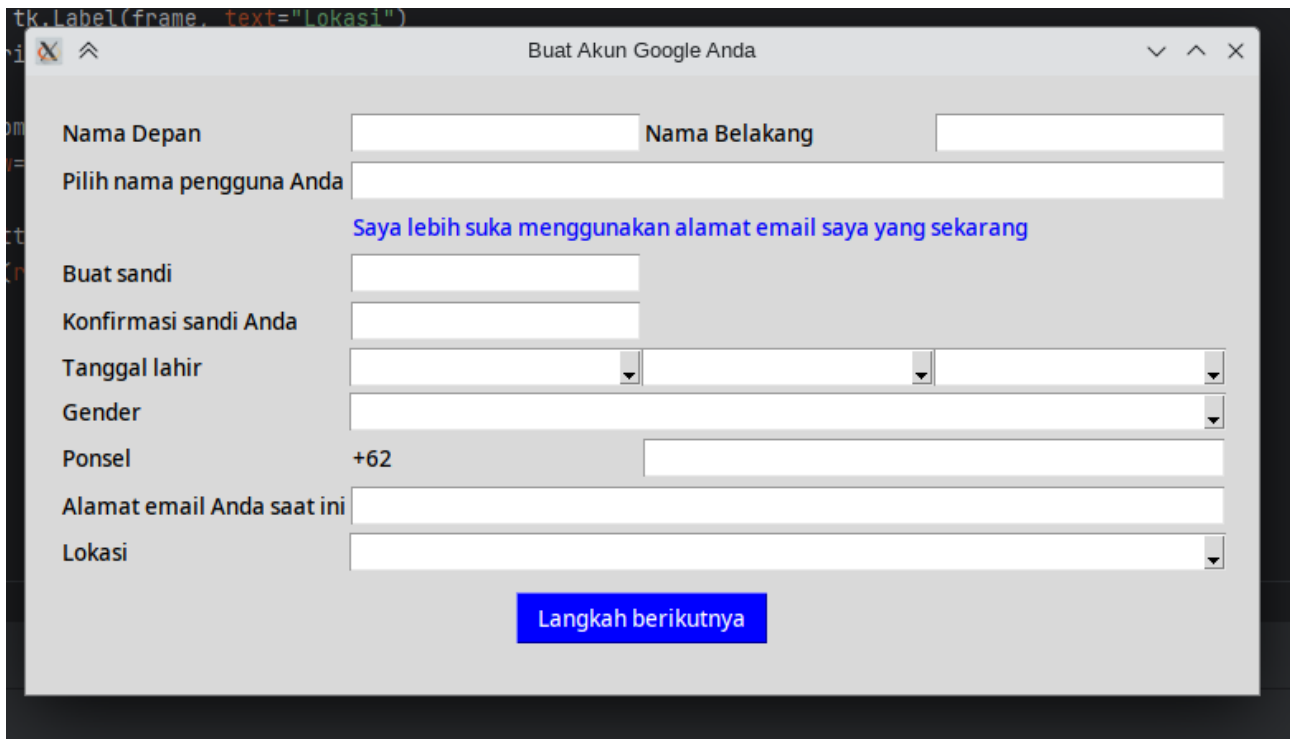
location = ttk.Combobox(frame, values=["Indonesia"])
location.grid(row=9, column=1, pady=2, columnspan=3, sticky=(tk.W, tk.E))

next_button = Button(frame, text="Langkah berikutnya", command=create_account, bg='blue',
fg='white')
next_button.grid(row=10, column=0, columnspan=4, pady=10)

app.mainloop()

```

Результат работы программы:



Постановка 2 задачи:

Вводится количество секунд с начала дня. Рассчитать количество полных минут с начал дня

Тип алгоритма: линейный

Текст программы:

```
import tkinter as tk
```

```
from tkinter import ttk
```

```
def calculate_minutes():
```

```
    try:
```

```
        seconds = int(entry_seconds.get())
```

```
        minutes = seconds // 60
```

```
        result_label.config(text=f"Количество полных минут: {minutes}")
```

```
    except ValueError:
```

```
        result_label.config(text="Введите корректное число секунд.")
```

```
app = tk.Tk()
```

```
app.title("Рассчитать полные минуты")
```

```
frame = ttk.Frame(app, padding="10 10 10 10")
```

```
frame.grid(row=0, column=0)
```

```
label_seconds = tk.Label(frame, text="Введите количество секунд:")
```

```
label_seconds.grid(row=0, column=0, sticky=tk.W, pady=2)
```

```
entry_seconds = tk.Entry(frame)
```

```
entry_seconds.grid(row=0, column=1, pady=2)
```

```

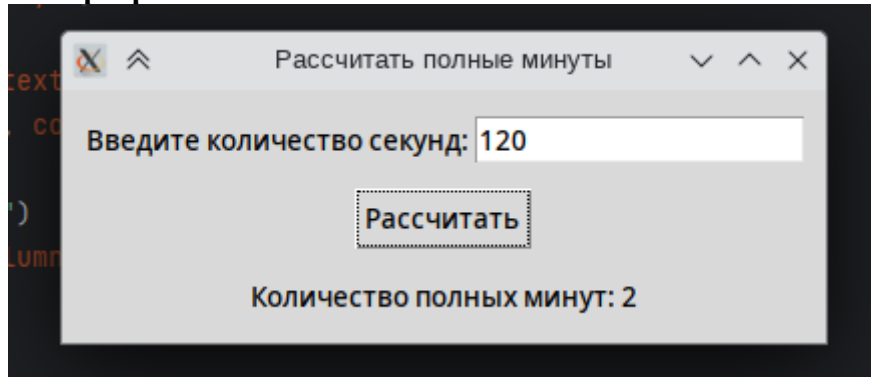
calculate_button = ttk.Button(frame, text="Рассчитать", command=calculate_minutes)
calculate_button.grid(row=1, column=0, columnspan=2, pady=10)

result_label = tk.Label(frame, text="")
result_label.grid(row=2, column=0, columnspan=2, pady=2)

app.mainloop()

```

Результат работы программы:



Постановка 3 задачи:

Задание предполагает, что у студента есть проект с практическими работами (№№ 2-13), оформленный согласно требованиям. Все задания выполняются с использованием модуля OS:

- 1) перейдите в каталог PZ11. Выведите список всех файлов в этом каталоге. Имена вложенных подкаталогов выводить не нужно.
- 2) 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.
- 3) перейти в папку с PZ11, найти там файл с самым коротким именем, имя вывести в консоль. Использовать функцию basename () (os.path.basename()).
- 4) перейти в любую папку где есть отчет в формате .pdf и «запустите» файл в привязанной к нему программе. Использовать функцию os.startfile().
- 5) удалить файл test.txt.

Тип алгоритма: линейный

Текст программы:

```

import os

# task 1
os.chdir('../PZ_11')
print(*filter(lambda x: not os.path.isdir(x), os.listdir()))

# task 2
os.chdir('.')
if not os.path.exists('test'):
    os.makedirs('test/test1')
os.replace('PZ_6/1.txt', 'test/1.txt')
os.replace('PZ_6/2.txt', 'test/2.txt')
os.replace('PZ_7/1.txt', 'test/test1/1.txt')

```

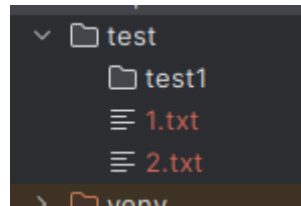
```
os.rename('test/test1/1.txt', 'test/test1/test.txt')
print(f'Размер файла test/1.txt {os.path.getsize('test/1.txt')}')
print(f'Размер файла test/2.txt {os.path.getsize('test/2.txt')}')

# task 3
os.chdir('PZ_11')
print(min([os.path.basename(i) for i in os.listdir()], key=len))

# task 4
os.chdir('../reports')
os.startfile('PZ_2.pdf')

# task 5
os.chdir('.')
os.remove('test/test1/test.txt')
```

Результат работы программы:



Вывод:

В процессе выполнения практического задания я закрепил усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрел навыки составления программ с использованием GUI Tkinter в IDE PyCharm Community, изучил возможности модуля OS.