

Практическое занятие №17

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

2.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.

3.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).

4.Тип алгоритма : циклический

5.Текст программы :

```
import tkinter as tk
from tkinter import ttk

def submit():
    # Здесь можно добавить логику для обработки введенных данных
    print("Данные подтверждены")

def cancel():
    # Здесь можно добавить логику для отмены ввода
    print("Отмена ввода")

root = tk.Tk()
root.title("Обработка формы")

# Метки и поля ввода
tk.Label(root, text="Форма регистрации пользователя", font=("Helvetica",
16)).grid(row=0, column=0, columnspan=2, pady=10)

tk.Label(root, text="Ваше имя:").grid(row=1, column=0, sticky="e", padx=5, pady=5)
entry_name = tk.Entry(root)
entry_name.grid(row=1, column=1, padx=5, pady=5)

tk.Label(root, text="Пароль:").grid(row=2, column=0, sticky="e", padx=5, pady=5)
entry_password = tk.Entry(root, show="*")
entry_password.grid(row=2, column=1, padx=5, pady=5)

tk.Label(root, text="Возраст:").grid(row=3, column=0, sticky="e", padx=5, pady=5)
```

```
entry_age = tk.Entry(root)
entry_age.grid(row=3, column=1, padx=5, pady=5)
```

```
tk.Label(root, text="Пол:").grid(row=4, column=0, sticky="e", padx=5, pady=5)
gender_var = tk.StringVar()
tk.Radiobutton(root, text="Мужской", variable=gender_var,
value="Мужской").grid(row=4, column=1, sticky="w")
tk.Radiobutton(root, text="Женский", variable=gender_var,
value="Женский").grid(row=4, column=1)
```

```
tk.Label(root, text="Ваши увлечения:").grid(row=5, column=0, sticky="e", padx=5,
pady=5)
hobbies_music = tk.BooleanVar()
hobbies_video = tk.BooleanVar()
hobbies_drawing = tk.BooleanVar()
tk.Checkbutton(root, text="Музыка", variable=hobbies_music).grid(row=5,
column=1, sticky="w")
tk.Checkbutton(root, text="Видео", variable=hobbies_video).grid(row=5, column=1)
tk.Checkbutton(root, text="Рисование", variable=hobbies_drawing).grid(row=5,
column=1, sticky="e")
```

```
tk.Label(root, text="Ваша страна:").grid(row=6, column=0, sticky="e", padx=5,
pady=5)
country_var = tk.StringVar()
country_combobox = ttk.Combobox(root, textvariable=country_var)
country_combobox['values'] = ("Россия", "Украина", "Беларусь", "Казахстан",
"Другие")
country_combobox.grid(row=6, column=1, padx=5, pady=5)
```

```
tk.Label(root, text="Ваш город:").grid(row=7, column=0, sticky="e", padx=5, pady=5)
city_var = tk.StringVar()
city_combobox = ttk.Combobox(root, textvariable=city_var)
city_combobox['values'] = ("Москва", "Санкт-Петербург", "Киев", "Минск",
"Алматы", "Другие")
city_combobox.grid(row=7, column=1, padx=5, pady=5)
```

```
tk.Label(root, text="Кратко о себе:").grid(row=8, column=0, sticky="e", padx=5,
pady=5)
entry_about = tk.Text(root, height=4, width=30)
entry_about.grid(row=8, column=1, padx=5, pady=5)
```

```
tk.Label(root, text="Решите пример, запишите результат в поле
ниже:").grid(row=9, column=0, columnspan=2, padx=5, pady=5)
entry_captcha = tk.Entry(root)
entry_captcha.grid(row=10, column=0, columnspan=2, padx=5, pady=5)
```

```
# Кнопки
```

```
cancel_button = tk.Button(root, text="Отменить ввод", command=cancel)
cancel_button.grid(row=11, column=0, pady=10)
```

```
submit_button = tk.Button(root, text="Данные подтверждаю", command=submit)
submit_button.grid(row=11, column=1, pady=10)
```

```
root.mainloop()
```

6. Протокол работы программы:

```
root, text="Отменить ввод", command=cancel)
col
root
col
```

7. Постановка задачи (2) : Разработать программу с применением пакета tk, взяв в качестве условия одну любую задачу из ПЗ №№ 2 – 9.

8. Текст программы:

```
import tkinter as tk
```

```
def check_distinct_digits(number):
    digits = set(str(number))
    return len(digits) == 3
```

```

def main():
    # Создаем главное окно
    root = tk.Tk()
    root.title("Проверка различных цифр")

    # Создаем поле ввода для номера
    number_label = tk.Label(root, text="Введите трехзначное число:")
    number_entry = tk.Entry(root)

    # Создаем кнопку для проверки
    check_button = tk.Button(root, text="Проверить", command=lambda:
check_number(number_entry.get()))

    # Создаем поле вывода для результата
    result_label = tk.Label(root, text="")

    # Размещаем элементы на окне
    number_label.pack()
    number_entry.pack()
    check_button.pack()
    result_label.pack()

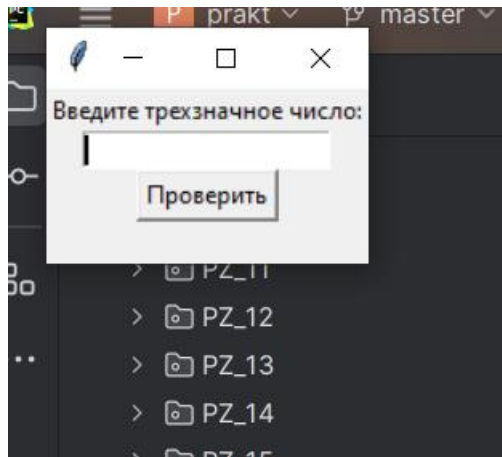
    def check_number(number):
        if not number.isdigit() or len(number) != 3:
            result_label["text"] = "Введите корректное трехзначное число."
        else:
            number = int(number)
            result = check_distinct_digits(number)
            if result:
                result_label["text"] = "Все цифры различны."
            else:
                result_label["text"] = "Не все цифры различны."

    # Запускаем главное окно
    root.mainloop()

if __name__ == "__main__":
    main()

```

9.Протокол работы программы:



10. Постановка задачи (3) : перейдите в каталог PZ11. Выведите список всех файлов в этом каталоге. Имена вложенных подкаталогов выводить не нужно.

11. Текст программы :

```
import os
```

```
os.chdir("../PZ_11")
files_in_pz11 = [f for f in os.listdir() if os.path.isfile(f)]
print("Список файлов в каталоге PZ_11:", files_in_pz11)
```

12. Протокол работы программы:

Список файлов в каталоге PZ_11: ['file1.txt', 'file2.txt', 'file3.txt', 'PZ_11.1.py', 'PZ_11.py', 'text18-21.txt', 'vtoroi_file.txt', '__init__.py']

Process finished with exit code 0

13. Постановка задачи (4) : 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.

14. Текст программы:

```
import os
import shutil
```

```
os.chdir('../..')
os.makedirs('test/test1', exist_ok=True)
```

```
file_from_pz6_1 = 'PZ_6/PZ_6.1.py'
file_from_pz6_2 = 'PZ_6/PZ_6.2.py'
file_from_pz7 = 'PZ_7/PZ_7.py'
```

```
shutil.move(file_from_pz6_1, 'test/')
shutil.move(file_from_pz6_2, 'test/')
shutil.move(file_from_pz7, 'test/test1/')
```

```
shutil.move(file_from_pz6_2, 'test/')
shutil.move(file_from_pz7, 'test/test1/PZ_7.1.txt')
```

```
files_in_test = [f for f in os.listdir('test') if os.path.isfile(os.path.join('test', f))]
for file in files_in_test:
    file_size = os.path.getsize(os.path.join('test', file))
    print(f"Размер файла {file} в папке test: {file_size} байт")
```

15.Протокол работы программы:

Размер файла PZ_6.1.py в папке test: 675 байт

Размер файла PZ_6.2.py в папке test: 1093 байт

Process finished with exit code 0

16.Постановка задачи (5) : перейти в папку с PZ11, найти там файл с самым коротким именем, имя вывести в консоль. Использовать функцию basename () (os.path.basename()).

17.Текст программы:

```
import os
```

```
os.chdir('../PZ_11')
shortest_name_file = min((f for f in os.listdir() if os.path.isfile(f)), key=len)
print("Файл с самым коротким именем:", os.path.basename(shortest_name_file))
```

18.Протокол работы программы:

Файл с самым коротким именем: PZ_11.py

Process finished with exit code 0

19.Постановка задачи (6) : перейти в любую папку где есть отчет в формате .pdf и «запустить» файл в привязанной к нему программе. Использовать функцию os.startfile().

20.Текст программы:

```
import os
```

```
pdf_report_path = 'D:/PycharmProjects/prakt/Reports/PZ_10.pdf'
os.startfile(pdf_report_path)
```

21.Протокол работы программы:

Практическое задание №10

Тема: Составление программ с применением множеств в IDE PyCharm Community.

Цель: Закрепить усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрести навыки составления программ с применением множеств в IDE PyCharm Community.

Постановка задачи: В озере водятся несколько видов рыб. Три рыбака поймали рыб некоторых их имеющихся в озере видов. Определить, рыб каких видов поймал каждый рыбак и рыб каких видов, имеющихся в озере, не выловил ни один из рыбаков.

Тип алгоритма: циклический

Текст программ:

```
fish_in_lake = {'лещ', 'капн', 'окунь', 'щука', 'голец', 'вобла', 'толстолобик'}
fisherman1 = {'лещ', 'капн', 'щука'}
fisherman2 = {'окунь', 'щука', 'голец'}
fisherman3 = {'лещ', 'окунь', 'голец'}

print("1 рыбак не поймал", fish_in_lake - fisherman1)
print("2 рыбак не поймал", fish_in_lake - fisherman2)
print("3 рыбак не поймал", fish_in_lake - fisherman3)
print()
print("1 рыбак поймал", fish_in_lake & fisherman1)
print("2 рыбак поймал", fish_in_lake & fisherman2)
print("3 рыбак поймал", fish_in_lake & fisherman3)
```

Протокол работы программы:

```
1 рыбак не поймал {'вобла', 'голец', 'толстолобик', 'окунь'}
2 рыбак не поймал {'капн', 'лещ', 'вобла', 'толстолобик'}
3 рыбак не поймал {'капн', 'щука', 'вобла', 'толстолобик'}
```

22. Постановка задачи (7) : удалить файл test.txt

23. Текст программы:

```
import os
```

```
os.remove('D:\\PycharmProjects\\prakt\\test\\test1\\test.txt')
print("Файл test.txt удален")
```

24. Протокол работы программы:

Файл test.txt удален

Process finished with exit code 0

Вывод: Сегодня я составлял программы с использованием GUI Tkinter в IDE PyCharm Community, изучал возможности модуля OS. Также 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

