Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего образования

"Российский экономический университет имени Г. В. Плеханова"

**МОСКОВСКИЙ ПРИБОРОСТРОИТЕЛЬНЫЙ ТЕХНИКУМ**

Специальность 09.02.07 Информационные системы и программирование

Квалификация: Программист

ОТЧЕТ

КОД БУДУЩЕГО

Листов: 18

|  |  |
| --- | --- |
| Выполнила студент | Проверил преподаватель |
| Группы П50-5-20 | \_\_\_\_\_\_\_\_\_\_Д.В. Серяк |
| Завьялов М.А. | " " \_\_\_\_\_\_\_\_2023 года |

Москва 2023

# Практическая работа №2

## Тема «Циклы условия».

Цель работы: изучить работу циклов и условий

Ход работы:

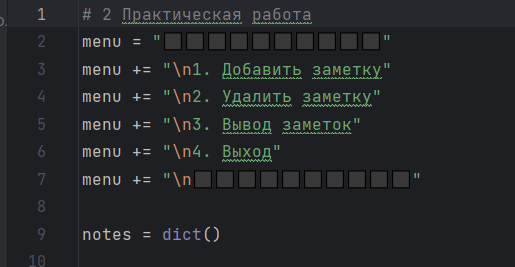


Рисунок 1 – Объявление переменных меню и массива

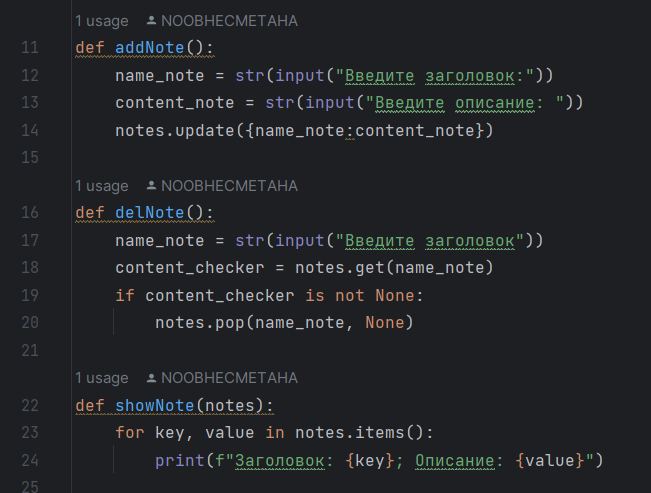


Рисунок 2 – Функции для блокнота



Рисунок 3 – Создания Функциональной части приложения

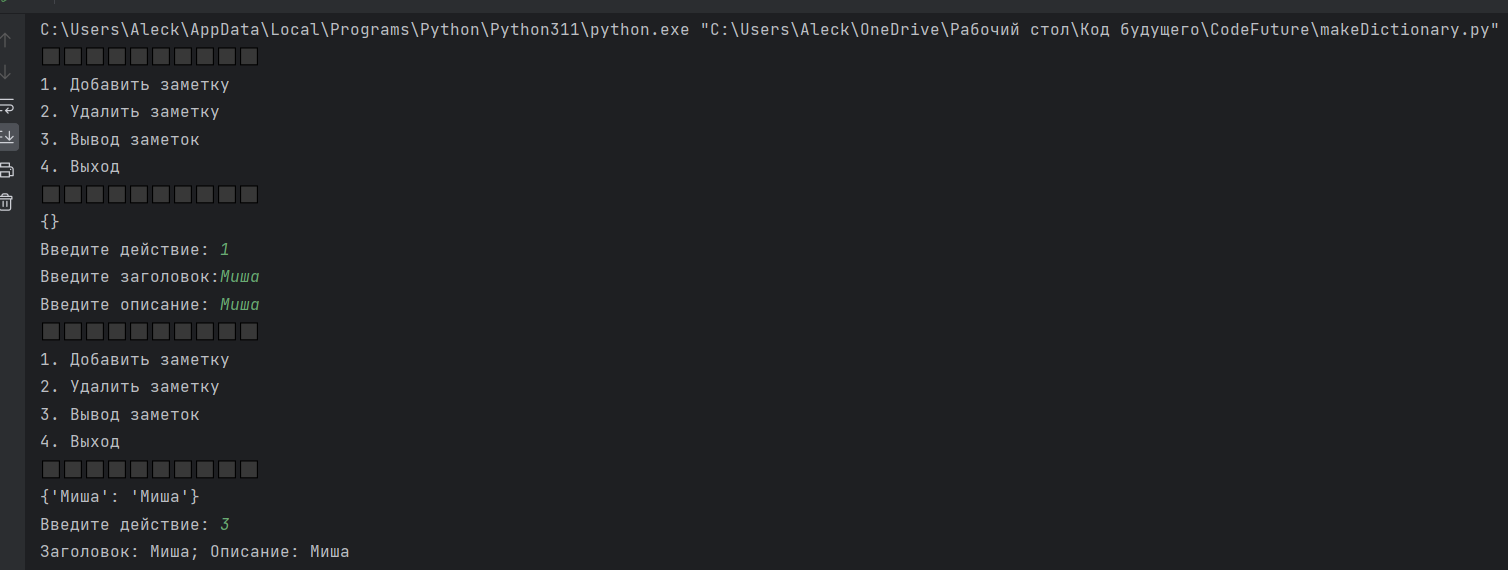


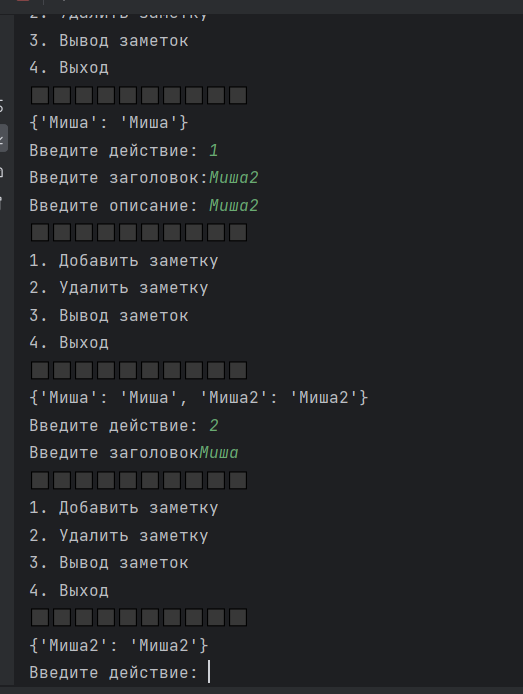
Рисунок 4 – Добавление и вывод заметки

Рисунок 5 – Добавление и удаление заметки

makeDirectionary.py

# 2 Практическая работа

menu = "⬛⬛⬛⬛⬛⬛⬛⬛⬛⬛"

menu += "\n1. Добавить заметку"

menu += "\n2. Удалить заметку"

menu += "\n3. Вывод заметок"

menu += "\n4. Выход"

menu += "\n⬛⬛⬛⬛⬛⬛⬛⬛⬛⬛"

notes = dict()

def addNote():

name\_note = str(input("Введите заголовок:"))

content\_note = str(input("Введите описание: "))

notes.update({name\_note:content\_note})

def delNote():

name\_note = str(input("Введите заголовок"))

content\_checker = notes.get(name\_note)

if content\_checker is not None:

notes.pop(name\_note, None)

def showNote(notes):

for key, value in notes.items():

print(f"Заголовок: {key}; Описание: {value}")

while True:

print(menu)

print(notes)

answer = int(input("Введите действие: "))

match answer:

case 1:

addNote()

case 2:

delNote()

case 3:

showNote(notes)

case 4:

break

case \_:

break

else:

print("Прощай")

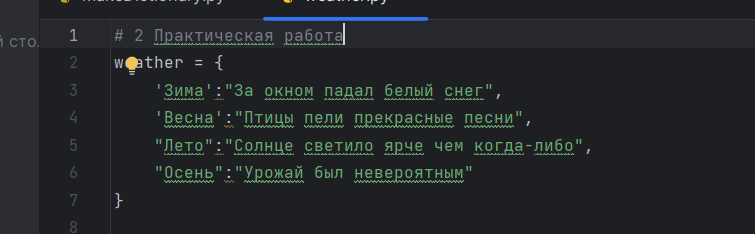


Рисунок 6 – Создание словаря для фраз

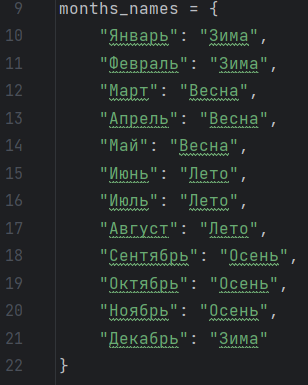


Рисунок 7 – Создания словаря на слова и сезоны

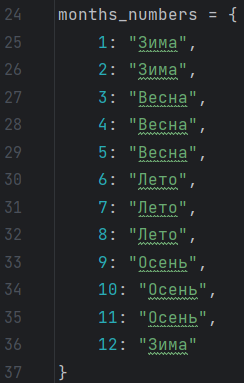


Рисунок 8 – Создание словаря на числа и время года

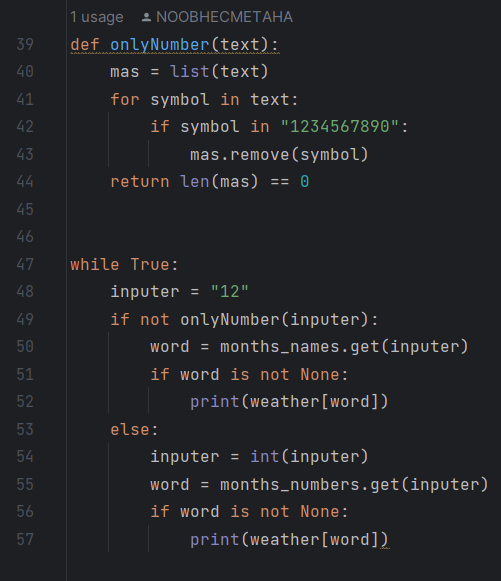


Рисунок 9 – Функция на проверку цифр и функциональная часть

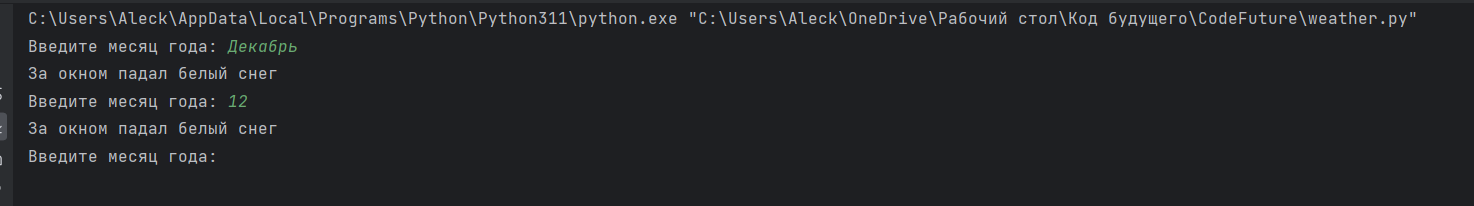


Рисунок 10 – Результат

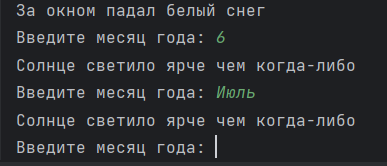


Рисунок 11 – Результат часть 2

Weather.py

# 2 Практическая работа

weather = {

'Зима':"За окном падал белый снег",

'Весна':"Птицы пели прекрасные песни",

"Лето":"Солнце светило ярче чем когда-либо",

"Осень":"Урожай был невероятным"

}

months\_names = {

"Январь": "Зима",

"Февраль": "Зима",

"Март": "Весна",

"Апрель": "Весна",

"Май": "Весна",

"Июнь": "Лето",

"Июль": "Лето",

"Август": "Лето",

"Сентябрь": "Осень",

"Октябрь": "Осень",

"Ноябрь": "Осень",

"Декабрь": "Зима"

}

months\_numbers = {

1: "Зима",

2: "Зима",

3: "Весна",

4: "Весна",

5: "Весна",

6: "Лето",

7: "Лето",

8: "Лето",

9: "Осень",

10: "Осень",

11: "Осень",

12: "Зима"

}

def onlyNumber(text):

mas = list(text)

for symbol in text:

if symbol in "1234567890":

mas.remove(symbol)

return len(mas) == 0

while True:

inputer = input("Введите месяц года: ")

if not onlyNumber(inputer):

word = months\_names.get(inputer)

if word is not None:

print(weather[word])

else:

inputer = int(inputer)

word = months\_numbers.get(inputer)

if word is not None:

print(weather[word])

Вывод: изучена работа с циклами и условиями

# Список иллюстраций

[Рисунок 1 – Создание контроллеров и моделей для API 2](#_Toc120729599)

[Рисунок 2 – Сгенерированный код контроллера «Материала» 3](#_Toc120729600)

[Рисунок 3 – Отредактированная модель «Продукта» 3](#_Toc120729601)

[Рисунок 4 – Сгенерированный сайт API 4](#_Toc120729602)

[Рисунок 5 - Сгенерированный сайт API часть 2 4](#_Toc120729603)

[Рисунок 6 – Просмотр объектов 5](#_Toc120729604)

[Рисунок 7 – Просмотр записей 5](#_Toc120729605)

[Рисунок 8 – Home Controller 6](#_Toc120729606)

[Рисунок 9 – Код для вывода таблицы «Пользователей» 6](#_Toc120729607)

[Рисунок 10 – Код для добавление записей в таблицу «Пользователей» 7](#_Toc120729608)

[Рисунок 11 – Код для изменение записи в таблице «Пользователей» 7](#_Toc120729609)

[Рисунок 12 – Удаление записей из таблице «Пользователей» 7](#_Toc120729610)

[Рисунок 13 – Разметка стартовой страницы для просмотра таблиц 8](#_Toc120729611)

[Рисунок 14 - Графическое отображение главной страницы 8](#_Toc120729612)

[Рисунок 15 – Разметка страница вывода таблицы «Пользователей» 9](#_Toc120729613)

[Рисунок 16 – Графическое отображение таблицы «Пользователей» 9](#_Toc120729614)

[Рисунок 17 – Разметка footer 10](#_Toc120729615)

[Рисунок 18 – Графическое отображение Footer 10](#_Toc120729616)

[Рисунок 19 – Разметка страницы определённой записи «Пользователя» 11](#_Toc120729617)

[Рисунок 20 – Графическое отображение страницы определённого пользователя 11](#_Toc120729618)

[Рисунок 21 – Разметка страницы определённой записи «Пользователя» 2 часть 12](#_Toc120729619)

[Рисунок 22 – Вывод таблицы определённого пользователя «Результат не найден» 12](#_Toc120729620)

[Рисунок 23 - Вывод таблицы определённого пользователя «Результат найден» 13](#_Toc120729621)

[Рисунок 24 – разметка добавление записи в таблицу «Пользователей» 13](#_Toc120729622)

[Рисунок 25 – Графическое отображение окна добавление «Пользователей» 14](#_Toc120729623)

[Рисунок 26 – Последний добавленный пользователь «Иван» 14](#_Toc120729624)

[Рисунок 27 – Разметка страницы обновления записи 14](#_Toc120729625)

[Рисунок 28 – Добавление записи 15](#_Toc120729626)

[Рисунок 29 – Запись изменение записи 15](#_Toc120729627)

[Рисунок 30 – Разметка метода удаления записи из таблицы «Пользователей» 16](#_Toc120729628)

[Рисунок 31 – удаление записи под ID 30 c именем «Челик» 16](#_Toc120729629)