Домашнее задание.

Маратулы Темирболат

**Замечания по выполнению домашнего задания.**

1. **Вверху этого документа напишите свою Фамилию и Имя.**
2. Исходные коды Ваших программ сохраняйте прямо здесь, в тексте задания, строго под текстом каждого задания.
3. В дополнение к исходным кодам вставляйте в документ скриншот результата выполнения программ.
4. Не надо документ архивировать и вкладывать скриншоты в архив. Скриншоты сразу вставляете в документ, как картинку.
5. **Скриншоты удобно делать программой https://app.prntscr.com/ru/privacy.html**
6. Переменные начинаются с маленькими буквами, слова разделяются символом «\_» нижняя черта.
7. Константы пишутся заглавными буквами, слова разделяются символом «\_» нижняя черта.
8. **Не забывайте КРАСИВО оформлять интерфейс программ, за плохой, ЛЕНИВЫЙ интерфейс я тоже СНИЖАЮ ОЦЕНКУ!**
9. Написать программу-секретарь. Пользователь вводит планируемые события, дату и время наступления события. Программа, по запросу соответствующего пункта меню, выводит сколько времени (дней, часов, минут, секунд) осталось до того или иного события.

**Программу декомпозировать на отдельные функции.**

**Обязательно оформить с удобным интерфейсом пользователя (меню пользователя)**

import datetime

import time

TIME\_FORMAT = '%d.%m.%Y %H:%M'

CHANGED\_TIME\_FORMAT\_DATE = '%d.%m.%Y'

CHANGED\_TIME\_FORMAT = '%H:%M'

FIND\_BY\_NAME\_OPTION = '1'

FIND\_BY\_DATE\_TIME\_OPTIONS = '2'

find\_options = (FIND\_BY\_NAME\_OPTION,FIND\_BY\_DATE\_TIME\_OPTIONS)

DELETE\_EVENT\_OPTION = '1'

CHANGE\_EVENT\_OPTION = '2'

event\_otions = (DELETE\_EVENT\_OPTION,CHANGE\_EVENT\_OPTION)

CHANGE\_DATE\_OPTION = '1'

CHANGE\_TIME\_OPTION = '2'

change\_options = (CHANGE\_DATE\_OPTION,CHANGE\_TIME\_OPTION)

events = {

}

GO\_OUT\_OPTION = "exit"

days = [ "Понедельник", "Вторник", "Среда", "Четверг", "Пятница","Суббота", "Воскресенье" ]

months = [ "", "января", "февраля", "марта", "апреля", "мая", "июня",

"июля", "августа", "сентября", "октября", "ноября", "декабря" ]

def get\_todays\_time\_gate():

    return time.localtime()

def view\_todays\_time\_date():

    time\_date = get\_todays\_time\_gate()

    week\_day\_index = time\_date[6]

    month\_index = time\_date[1]

    day\_number = time\_date[2]

    year = time\_date[0]

    print()

    print('Сегодня: {} {} {} {} года '.format(days[week\_day\_index],day\_number,months[month\_index],year))

    print('Текущее время: {}:{}:{}'.format(time\_date[3],time\_date[4],time\_date[5]))

    print()

def get\_left\_time\_event(event\_date\_time):

    todays\_passed\_seconds = int(time.mktime(time.localtime()))

    event\_passed\_seconds = int(time.mktime(event\_date\_time))

    time\_difference\_seconds = event\_passed\_seconds - todays\_passed\_seconds

    return time\_difference\_seconds

def get\_formated\_time(event\_time\_seconds):

    left\_time = str(datetime.timedelta(seconds = event\_time\_seconds))

    if('days' in left\_time):

        left\_time = left\_time.replace('days', 'дней/дня')

    return left\_time

def view\_all\_set\_events():

    if(len(events) != 0):

        print('\nВсе ваши назначенные события:')

        for event in events:

            print('Событие: "{}" с датой {} {} {} {} года'.format(event,days[events[event][6]],events[event][2],months[events[event][1]],events[event][0]),end = ' ')

            print('и временем: {}:{}'.format(events[event][3],events[event][4]),'Описание:',end = "")

            left\_time\_event = get\_left\_time\_event(events[event])

            if(left\_time\_event > 0):

                print('\tосталось времени до события: ' + get\_formated\_time(left\_time\_event))

            else:

                print('\tсобытие прошло')

    else:

        print('\nИзвините, ваш список событий пуст')

def view\_change\_event\_options():

    print('Все возможные опции:')

    print('\t\t\t 1) Поменять Дату события')

    print('\t\t\t 2) Поменять Время события')

def change\_time\_event(event\_name):

    print('\nВы решили изменить ВРЕМЯ события "{}"'.format(event\_name))

    print('Пожалуйста введите новое ВРЕМЯ для события "{}"'.format(event\_name))

    new\_time = input('Введите новое время для события в формате "HH:MM": ')

    new\_time = '{}.{}.{}'.format(events[event\_name][2],events[event\_name][1],events[event\_name][0]) + ' ' + new\_time

    new\_time = time.strptime(new\_time,TIME\_FORMAT)

    events[event\_name] = new\_time

    print('Время события "{}" успешно изменено!'.format(event\_name))

def change\_date\_event(event\_name):

    print('\nВы решили изменить ДАТУ события "{}"'.format(event\_name))

    print('Пожалуйста введите новую дату для события "{}"'.format(event\_name))

    new\_date = input('Введите новую дату события в формате "dd.mm.YYYY": ')

    new\_date = new\_date + ' ' + '{}:{}'.format(events[event\_name][3],events[event\_name][4])

    new\_date = time.strptime(new\_date,TIME\_FORMAT)

    events[event\_name] = new\_date

    print('Ваша дата успешно изменена!')

def delete\_event(event\_name):

    print('\nВы решили удалить событие')

    del events[event\_name]

    print('Событие "{}" успешно удалено!'.format(event\_name))

    print()

def view\_find\_options():

    print('\nВсе возможные опции по поиску:')

    print('\t\t\t 1) Найти событие по Имени')

    print('\t\t\t 2) Найти событие по Дате и Времени')

def view\_event\_options():

    print('\nВсе возможные опции: ')

    print('\t\t\t 1) Удалить событие')

    print('\t\t\t 2) Изменить событие')

def modify\_event(event\_name):

    print('\nЧто вы хотите сделать с событием "{}"?'.format(event\_name))

    view\_event\_options()

    user\_option = input('Ваш выбор: ')

    while(user\_option not in event\_otions):

        print('Извините, но такой опции нет. Повторите пожалуйста ввод')

        view\_event\_options()

        user\_option = input('Ваш выбор: ')

    if(user\_option == DELETE\_EVENT\_OPTION):

        delete\_event(event\_name)

    else:

        print('Что вы хотите изменить?')

        view\_change\_event\_options()

        user\_choice = input('Ваш выбор: ')

        while(user\_choice not in change\_options):

            print('Извините, но такой опции нет. Повторите попытку')

            view\_change\_event\_options()

            user\_choice = input('Ваш выбор: ')

        if(user\_choice == CHANGE\_DATE\_OPTION):

            change\_date\_event(event\_name)

        else:

            change\_time\_event(event\_name)

def fint\_event\_by\_date():

    print('\nВы решили найти событие по указанной вами дате и времени')

    user\_find\_time = ask\_time\_event()

    all\_dates = list(events.values())

    if(user\_find\_time in all\_dates):

        event\_index = all\_dates.index(user\_find\_time)

        counter = 0

        found\_event = ''

        for event in events:

            if(counter == event\_index):

                found\_event = event

                break

            counter += 1

        modify\_event(found\_event)

    else:

        print('Извините, но такого события нет')

def find\_event\_by\_name():

    print('\nПожалуйста, укажите название события, которое вы хотите найти')

    event\_name = input('Укажите название события либо "exit", чтобы выйти в главное меню: ')

    while(event\_name not in events and event\_name != GO\_OUT\_OPTION):

        print('Извините, такого события нет...Повторите попытку или',end = " ")

        print('напечатайте "exit" чтобы вернуться в главное меню')

        event\_name = input('Укажите название события либо "exit", чтобы выйти в главное меню: ')

    if(event\_name == GO\_OUT\_OPTION):

        print('Возвращаемся в главное меню...')

        return

    modify\_event(event\_name)

def ask\_time\_event():

    print('\nУкажите интересующее вас время и дату в формате dd.mm.YYYY HH:MM',end = ', ')

    print('где dd - номер дня, mm - номер месяца, YYYY - номер года',end = " ")

    print('HH - часы, MM - число минут')

    user\_given\_time = input('Укажите время и дату в формате "dd.mm.YYYY HH:MM": ')

    user\_time = time.strptime(user\_given\_time,TIME\_FORMAT)

    return user\_time

def add\_new\_event():

    print('\nВы решили добавить новое событие')

    print('Пожалуйста, укажите название для нового события или напишите',end = ' ')

    print('"exit", чтобы вернуться в главное меню')

    new\_event\_name = input('Название для нового события: ')

    while(new\_event\_name in events and new\_event\_name != GO\_OUT\_OPTION):

        print('Извините, такое событие уже существует. Придумайте ',end = "")

        print('пожалуйста другое Название для события. Либо ',end = '')

        print('напишите "exit", чтобы вернуться в главное меню')

        new\_event\_name = input('Название для нового события: ')

    if(new\_event\_name == GO\_OUT\_OPTION):

        print('\nВозвращаемся в главное меню...')

        return

    print('Пожалуйста, укажите дату и время для предстоящего события')

    new\_event\_date\_time = ask\_time\_event()

    events.setdefault(new\_event\_name,new\_event\_date\_time)

    print('Ваше событие "{}" успешно добавлено!'.format(new\_event\_name))

def find\_event():

    print('\nВы решили найти событие. Пожалуйста, выберите метод поиска события')

    print('Если вы хотите вернуться в главное меню, то напишите "exit"')

    view\_find\_options()

    user\_option = input('Ваш выбор: ')

    while(user\_option not in find\_options and user\_option != GO\_OUT\_OPTION):

        print('Извините, но такой опции нет. Пожалуйста повторите ',end = "")

        print('запрос или напишите "exit" для выхода в главное меню')

        view\_find\_options()

        user\_option = input('Ваш выбор: ')

    is\_found\_event = False

    found\_event\_index = -1

    if(user\_option == GO\_OUT\_OPTION):

        print('Возвращаемся в главное меню...')

        return

    elif(user\_option == FIND\_BY\_NAME\_OPTION):

        is\_found\_event = find\_event\_by\_name()

    else:

        found\_event\_index = fint\_event\_by\_date()

EXIT\_OPTION = '0'

ADD\_NEW\_EVENT\_OPTION = '1'

FIND\_EVENT = '2'

VIEW\_ALL\_EVENTS\_OPTION = '3'

VIEW\_TODAY\_DATE\_TIME\_OPTION = '4'

main\_program\_options = (EXIT\_OPTION,ADD\_NEW\_EVENT\_OPTION,FIND\_EVENT,VIEW\_ALL\_EVENTS\_OPTION,VIEW\_TODAY\_DATE\_TIME\_OPTION)

def view\_main\_options():

    print('\nВсе доступные опции:')

    print('\t\t\t 0) Выйти из программы')

    print('\t\t\t 1) Добавить новое событие')

    print('\t\t\t 2) Найти событие')

    print('\t\t\t 3) Просмотреть все события')

    print('\t\t\t 4) Просмотреть сегодняшнюю Дату и Время')

def start\_main\_program():

    print('Здравствуйте! Добро пожаловать в программу, которая выводит все',end = " ")

    print('предстоящии события, дату и время наступления события. Программа',end = " ")

    print('выводит сколько времени (дней, часов, минут, секунд) осталось до',end = " ")

    print('того или иного события.')

    is\_start\_program = True

    print('\nНачнем программу...')

    while(is\_start\_program):

        print('\nПожалуйста, выберите одну из доступных опций')

        view\_main\_options()

        user\_option = input('Ваш выбор: ')

        if(user\_option not in main\_program\_options):

            print('Извините, такой опции нет. Повторите попытку')

            continue

        elif(user\_option == EXIT\_OPTION):

            print('Спасибо за использование программы.До свидания')

            break

        elif(user\_option == ADD\_NEW\_EVENT\_OPTION):

            add\_new\_event()

        elif(user\_option == VIEW\_ALL\_EVENTS\_OPTION):

            view\_all\_set\_events()

        elif(user\_option == VIEW\_TODAY\_DATE\_TIME\_OPTION):

            view\_todays\_time\_date()

        else:

            find\_event()

start\_main\_program()











