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!**

Написать программу, которая:

1. Примет от пользователя формат времени и выведет на экран текущее время в заданном пользователем формате;
2. Примет от пользователя количество секунд, прошедшее с начала эпохи UNIX и выведет на экран время в формате текущей локализации;
3. Примет от пользователя некий формат даты и времени, а затем дату и время в заданном, на предыдущем шаге формате и выведет на экран количество секунд, прошедшее от начала эпохи UNIX;
4. После выполнения п.3, дополнительно, выведет на экран количество секунд прошедших от заданного в п.3 времени и даты до текущего времени;

**Программу декомпозировать на отдельные функции.**

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

import time

EXIT\_PROGRAM\_OPTION = 'exit'

FIRST\_EXERCISE\_OPTION = '1'

SECOND\_EXERCISE\_OPTION = '2'

THIRD\_EXERCISE\_OPTION = '3'

FOURTH\_EXERCISE\_OPTION = '4'

menu\_options = (FIRST\_EXERCISE\_OPTION,SECOND\_EXERCISE\_OPTION,THIRD\_EXERCISE\_OPTION,FOURTH\_EXERCISE\_OPTION,EXIT\_PROGRAM\_OPTION)

# Exercise 1

def start\_exercise\_one():

    print()

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

    print('от пользователя формат времени в котором он/она хочет увидеть ',end = "")

    print('текущее время')

    print('\nПеред тем как писать соответствующий формат, пишите %')

    user\_time\_format = input('Пожалуйста, введите желаемый формат времени: ')

    print('Время в желаемом вами формате:',time.strftime(user\_time\_format))

#Exercise 2

def start\_exercise\_two():

    print()

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

    print('принимает от пользователя количество секунд, прошедшее',end = " ")

    print('с начала эпохи UNIX, а затем выведет на экран время в ',end = "")

    print('формате текущей локализации.')

    user\_passed\_seconds = int(input('\nПожалуйста, введите пройденные секунды: '))

    print('Дата соответсвующая количеству секунд указанных вами:',time.ctime(user\_passed\_seconds))

#Execise 3

def get\_passed\_seconds\_to\_origin():

    user\_format = input('\nПожалуйста, введите формат времени: ')

    user\_day\_time = input('Пожалуйста, введите время соответствующее вашему формату: ')

    user\_day\_time\_struct = time.strptime(user\_day\_time,user\_format)

    passed\_seconds\_origin = time.mktime(user\_day\_time\_struct)

    return passed\_seconds\_origin

def start\_exercise\_three():

    print()

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

    print('примет от пользователя формат даты и времени,',end = " ")

    print('затем дату и время,в заданному пользователем формате.')

    print('Программа выведет на экран количество секунд, которое ',end = "")

    print('прошло с начала эпохи UNIX.')

    passed\_seconds = get\_passed\_seconds\_to\_origin()

    print('\nКоличество секунд, прошедшее от начала эпохи UNIX ',end = "")

    print('согласно вашим данным: ',passed\_seconds,'секунд')

#Exercise 4

def start\_exercise\_four():

    print()

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

    print('примет от пользователя формат даты и времени,', end = " ")

    print('затем дату и время, в заданном вами формате.')

    print('Программа выведет на экран количество секунд, прошедшее ',end = "" )

    print('с начала эпохи UNIX, а также время прошедшее от заданного ', end = "")

    print('времени до текущего.')

    passed\_seconds = get\_passed\_seconds\_to\_origin()

    print('\nКоличество секунд, прошедшее от начала эпохи UNIX ',end = "")

    print('согласно вашим данным: ',passed\_seconds,'секунд.')

    print('\nСекунды, прошедшие с указанного времени до текущего:',end = " ")

    print(int(time.mktime(time.gmtime()) - passed\_seconds),'секунд.')

def view\_available\_exercises():

    print('\nНапечатайте цифру одну из показанных ниже, или "exit",чтобы выйти:')

    print('\t\t\t 0) Напечатайте "exit", чтобы выйти из программы')

    print('\t\t\t 1) Напечатайте 1, чтобы запустить первое задание')

    print('\t\t\t 2) Напечатайте 2, чтобы запустить второе задание')

    print('\t\t\t 3) Напечатайте 3, чтобы запустить третье задание')

    print('\t\t\t 4) Напечатайте 4, чтобы запустить четвёртое задание')

def start\_main\_program():

    print('\nЗдравствуйте! Добро пожаловать в программу!')

    print('Вы можете запустить 1 из 4 домашних заданий на ваш выбор.',end = " ")

    print('Ниже будут представлены все инструкции...')

    while(True):

        view\_available\_exercises()

        user\_option = input('Введите необходимый номер задачи: ')

        if(user\_option not in menu\_options):

            print('Вы выбрали не ту опцию... Повторите ввод!')

            continue

        elif(user\_option == FIRST\_EXERCISE\_OPTION):

            start\_exercise\_one()

        elif(user\_option == SECOND\_EXERCISE\_OPTION):

            start\_exercise\_two()

        elif(user\_option == THIRD\_EXERCISE\_OPTION):

            start\_exercise\_three()

        elif(user\_option == FOURTH\_EXERCISE\_OPTION):

            start\_exercise\_four()

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

            break

    print('\nСпасибо за использование программы! До свидания!')

start\_main\_program()



