

**Реализация приложения календаря на языке программирования Python***Кизянов Антон Олегович**Приамурский государственный университет имени Шолом-Алейхема**Студент***Аннотация**

В данной статье описан процесс создания приложения для построения календарей на конкретный год и месяц. Для создания используется язык программирования Python и его стандартная библиотека. Созданное приложение служит быстрым способом получения календаря на нужный год и месяц.

**Ключевые слова:** Python, календарь

**Implementing a Calendar Application in Python programming language***Kizyanov Anton Olegovich**Sholom-Aleichem Priamursky State University**student***Abstract**

This article describes the process of creating an application for building calendars for a specific year and month. For creation, the Python programming language and its standard library are used. The created application serves as a quick way to get the calendar for the desired year and month.

**Keywords:** Python, calendar

Программа генерирует печатаемые текстовые файлы ежемесячных календарей на месяц и год, которые вводит пользователь. Даты и календари - сложная тема в программировании, потому что существует очень много разных правил для определения количества дней в месяце, какие годы являются високосными и на какой день недели приходится конкретная дата.

Цель исследования – написать приложение для универсального календаря на языке программирования Python.

Ранее этим вопросом интересовались В.А. Слепов развивал тему «Проектирование и разработка программного продукта "личный блокнот для записи мыслей"» [1] в которой актуализируется необходимость разработки личного блокнота для записи мыслей. Определяются этапы разработки предложенного программного продукта: изучение предпочтений пользователей, определение требований к программе и проектирование, разработка графического интерфейса программы, разработка приложения, оформление необходимой документации. Для анализа предпочтений пользователей предложен вариант набора вопросов социологического

опроса для определения актуальности разработки. Полученные результаты опроса обработаны и продемонстрированы посредством диаграмм. Формирование общих требований к программному продукту показано с помощью диаграммы прецедентов. Диаграмма прецедентов определяет особенности программного продукта, отображает все способы взаимодействия пользователя: авторизация в приложении, создание, редактирование и удаление записей. Построена диаграмма активности, которая позволяет детализировать особенности реализации программным продуктом операций и прецедентов. Графический интерфейс создан средствами набора расширений графического фреймворка Qt для языка программирования Python. В.С. Чернова, Е.Б. Сидорова, Т.Г. Орешкина с темой «Разработка приложения «mydiary» на python» [2], а подробнее про проектирование и разработку приложения с графическим интерфейсом на языке программирования Python 3.8 с использованием набора библиотек PyQt. В качестве решения проблемы предлагается внедрение разработанной подсистемы планирования, реализованной на языке программирования Python, с применением свободного программного каркаса для веб-приложений Django, а также свободной объектно-реляционной системы управления базами данных PostgreSQL. С.А. Попков, С.А. Никитина опубликовали статью «Разработка календаря-планировщика» [3] в статье демонстрируется процесс создания приложения, которое будет информировать о событиях. Для этого использован функционал мессенджера Telegram, язык программирования Python. Был создан чат-бот, имитирующий органайзер-напоминку, который будет помогать людям и вовремя сообщать о предстоящих событиях в удобном формате.

Функция `getCalendarFor()` возвращает гигантскую многострочную строку календаря для данного месяца и года. В этой функции переменная `calText` хранит эту строку, которая добавляет к ней строки, пробелы и даты. Для отслеживания даты переменная `currentDate` содержит объект `datetime.date()`, который устанавливает следующую или предыдущую дату в путем добавления или вычитания объектов `datetime.timedelta()`.

```
import datetime

MONTHS = ('Январь', 'Февраль', 'Март', 'Апрель',
          'Май', 'Июнь', 'Июль', 'Август',
          'Сентябрь', 'Октябрь', 'Ноябрь', 'Декабрь')

while True:
    print('Введите год:')
    resp = input('> ')
    if resp.isdecimal() and int(resp) > 0:
        year = int(resp)
        break
    print('Пожалуйста, введите год как этот 2023.')
    continue

while True:
    print('Введите месяц, из этого диапазона, 1-12:')
    resp = input('> ')
```

```

if not resp.isdecimal():
    print('Введите номер месяца, например 2 для февраля.')
    continue

month = int(resp)
if 1 <= month <= 12:
    break

print('Пожалуйста, введите число от 1 до 12.')

def getCalFor(year, month):
    text = ''
    text += (' ' * 34) + MONTHS[month - 1] + ' ' + str(year) + '\n'
    text +=
    'Воскресенье..Понедельник..Вторник...Среда.....Четверг....Пятница....Суббота
    \n'

    weeksep = ('+-----' * 7) + '+\n'
    emptyrow = ('| ' * 7) + '|\n'
    DayNow = datetime.date(year, month, 1)

    while DayNow.weekday() != 6:
        DayNow -= datetime.timedelta(days=1)

    while True:
        text += weeksep
        dayNumberRow = ''
        for i in range(7):
            dayNumberLabel = str(DayNow.day).rjust(2)
            dayNumberRow += f'|{dayNumberLabel} '
            DayNow += datetime.timedelta(days=1)
        dayNumberRow += '|\n'
        text += dayNumberRow
        for i in range(3):
            text += emptyrow
        if DayNow.month != month:
            break
    text += weeksep
    return text

text = getCalFor(year, month)
print(text)

```

Результат работы приложения можно увидеть на рисунке 1.

Enter the year for the calendar:						
> 2022						
Enter the month for the calendar, 1-12:						
> 1						
January 2022						
Воскресенье...Понедельник...Вторник...Среда.....Четверг....Пятница....Суббота						
26	27	28	29	30	31	1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	31	1	2	3	4	5

Рис. 1 Результат вывода на 2022 год и 1 месяц

### Вывод

В этой статье было приведено описание приложения построения календаря. Приложение позволяет создать календарь на конкретный год и месяц. Благодаря этому приложению можно быстро создать нужный календарь за требуемый период.

**Библиографический список**

1. Слепов В.А. Проектирование и разработка программного продукта "личный блокнот для записи мыслей" // Научное обозрение. Технические науки. 2020. № 4. С. 58-63. URL: <https://elibrary.ru/item.asp?id=43932684> (Дата обращения: 05.01.2022)
2. Чернова В.С., Сидорова Е.Б., Орешкина Т.Г. Разработка приложения «mydiary» на python // В сборнике: Сборник научных трудов молодых ученых и специалистов. Сборник статей. В 2-х частях. Чебоксары, 2021. С. 408-413. URL: <https://elibrary.ru/item.asp?id=47359925> (Дата обращения: 05.01.2022)
3. Попков С.А., Никитина С.А. Разработка календаря-планировщика // В сборнике: Наука сегодня: опыт, традиции, инновации. Материалы международной научно-практической конференции. 2020. С. 15-16. URL: <https://elibrary.ru/item.asp?id=43779148> (Дата обращения: 05.01.2022)