МИНИСТЕРСТВО ТРАНСПОРТА РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное учреждение высшего образования «Российский университет транспорта»

Институт экономики и финансов

Кафедра «Информационные системы цифровой экономики»

**Курсовая работа**

по дисциплине «Программная инженерия»

на тему «Python. Использование словарей»

Выполнил студент

группы ЭМС-111

Дроздов А. Д.

Научный руководитель:

Логинова Людмила Николаевна

Кандидат технических наук, доцент

Москва, 2022 год

ОГЛАВЛЕНИЕ

[ВВЕДЕНИЕ 3](#_Toc103340933)

[1. ЯЗЫК PYTHON: ОПИСАНИЕ, ХАРАКТЕРИСТИКА, ВОЗМОЖНОСТИ 4](#_Toc103340934)

[1.1 Расширяемость Python. 5](#_Toc103340935)

[1.2.1 Графический интерфейс 6](#_Toc103340936)

[1.2.2 Базы данных 6](#_Toc103340937)

[1.2.3 Веб-разработка 7](#_Toc103340938)

[1.2.4 Манипуляции с изображениями и видео 8](#_Toc103340939)

[2. ИНДИВИДУАЛЬНОЕ ЗАДАНИЕ 10](#_Toc103340940)

[2.1 Условие задачи 10](#_Toc103340941)

[2.2 Решение 10](#_Toc103340942)

[2.3 Тестирование 12](#_Toc103340943)

[ЗАКЛЮЧЕНИЕ 14](#_Toc103340944)

[СПИСОК ЛИТЕРАТУРЫ 15](#_Toc103340945)

# ВВЕДЕНИЕ

На данный момент высокоуровневый язык программирования Python является очень актуальным при разработке приложений. Большой набор библиотек, инструментов, шаблонов, компактность кода и структурированность – вот причина популярности данного языка программирования.

Целью настоящей курсовой работы является изучение инструментов Python и реализация программы, указанной в задании №6.

Задачи курсовой работы:

1. Описание языка программирования Python;
2. Изучение инструментов языка программирования Python;
3. Реализация программы, описанной в задании №5, с использованием инструментов языка программирования Python.

Курсовая работа состоит из двух глав, введения, заключения, списка литературы и приложения (файл Дроздов\_Курсовая\_работа.ipynb).

В первой главе описан язык Python, история его создания, основные характеристики, возможности.

Во второй глава описан принцип работы реализованной программы.

1. ЯЗЫК PYTHON: ОПИСАНИЕ, ХАРАКТЕРИСТИКА, ВОЗМОЖНОСТИ

Python — это высокоуровневый язык программирования общего назначения, который используется в том числе и для разработки веб-приложений. Язык ориентирован на повышение производительности разработчика и читаемости кода.

Python поддерживает несколько парадигм программирования: структурное, объектно-ориентированное, функциональное, императивное и аспектно-ориентированное. В языке присутствует динамическая типизация, автоматическое управление памятью, полная интроспекция, механизм обработки исключений, поддержка многопоточных вычислений и удобные высокоуровневые структуры данных. Программный код на Python организовывается в функции и классы, которые могут объединяться в модули, а они в свою очередь могут быть объединены в пакеты. Python обычно используется как интерпретируемый, но может быть скомпилирован в байт-код Java и в MSIL (в рамках платфоры .NET).

По производительности интерпретируемый Python похож на все остальные подобные языки, но возможность компиляции в байт-код позволяет добиться большей производительности.

По сравнению с [Ruby](https://web-creator.ru/articles/ruby) и некоторыми другими языками, в Python отсутствует возможность модифицировать встроенные классы, такие, как int, str, float, list и другие.

В Python присутствует глобальная блокировка интерпретатора (GIL) — при своей работе основной интерпретатор постоянно использует большое количество потоко-небезопасных данных. В основном это словари, в которых хранятся атрибуты объектов, и обращения к внешнему коду, поэтому, во избежание разрушения этих данных при совместной модификации из разных потоков, перед началом исполнения нескольких инструкций (обычно ста) поток интерпретатора захватывает GIL, а по окончании освобождает. [1]

* 1. Расширяемость Python.

Одна из отличительных черт Python — его невероятная расширяемость – существует много готовых библиотек (как встроенных, так и сторонних) и фреймворков для решения задач.

Библиотеками в программировании называют набор готовых методов и классов для решения конкретных типов задач.

Фреймворк же – это программная платформа, определяющая структуру программной системы; программное обеспечение, облегчающее разработку и объединение разных компонентов большого программного проекта.

* Pygame. Библиотека для создания небольших игр и мультимедийных приложений;
* NumPy. Библиотека для работы с искусственным интеллектом и машинным обучением. Используется для сложных математических вычислений;
* Pandas. Библиотека для работы с большими данными;
* SQLAlchemy. Библиотека для работы с базами данных;
* Django, Flask. Фреймворки для разработки серверной части приложений;

Наличие библиотек значит, что под каждую задачу есть свой инструмент. Придумывать что-то сложное с нуля не придется. [2]

Далее представлен список одних из самых популярных и чаще всего используемых при работе с Python инструментов.

* + 1. Графический интерфейс

В стандартной библиотеке Python есть графическая библиотека интерфейсов Tkinter. Но при помощи внешних модулей вы можете поднять GUI ваших приложений на новый уровень.

* wxPython. Создает по-настоящему нативные пользовательские интерфейсы для Python-приложений, которые могут запускаться на Windows, Mac и Linux, а также прочих Unix-подобных системах практически без каких-либо модификаций.
* PyGObject. Пакет Python, предоставляющий привязки для библиотек на базе Gobject, таких как GTK, GStreamer, WebKitGTK, GLib, GIO и др.
* Pmw. Набор инструментов для создания на Python высокоуровневых составных виджетов (с использованием модуля Tkinter).
* WCK. Расширение API, позволяющее реализовывать на чистом Python всевозможные пользовательские виджеты.
* Tix. Мощный набор компонентов пользовательского интерфейса, при помощи которого можно расширить возможности ваших Tcl/Tk и Python-приложений. Использование Tix вместе с Tk очень улучшает внешний вид и функционал приложений.
  + 1. Базы данных

Модули, перечисленные в этом разделе, имеют открытый исходный код. С их помощью можно существенно облегчить доступ к базам данных.

* MySQLdb. Python DB API-2.0-совместимый интерфейс для доступа к базам данных MySQL.
* PyGreSQL. Модуль, который служит интерфейсом для базы данных PostgreSQL. Он дает возможность работать с PostgreSQL-фичами из Python-скрипта.
* Gadfly. Простая реляционная СУБД, реализованная на Python (на основе SQL).
* SQLAlchemy. Библиотека Python для работы с базами данных с применением ORM. Позволяет разработчикам приложений использовать всю мощь и гибкость SQL.
* KinterbasDB. Расширение Python, реализующее поддержку совместимости Python Database API 2.0 для опенсорсной реляционной базы данных Firebird и некоторых версий ее проприетарной кузины — Borland Interbase.
  + 1. Веб-разработка

Python это популярный язык для веб-разработки. Все пакеты Python, перечисленные в этом разделе, делают жизнь веб-разработчика гораздо проще.

* Beautiful Soup. Библиотека Python для быстрого парсинга HTML- и XML-файлов.
* scrape. Модуль Python для веб-браузинга и скрапинга.
* mechanize. Очень полезный модуль, благодаря которому вы получаете эмулятор браузера для взаимодействия с веб-страницами.
* libgmail. Обеспечивает доступ к гугловскому сервису Gmail.
* Google Maps. Благодаря этому модулю можно использовать веб-сервисы платформы Google Maps в своем приложении на Python.
* Requests позволяет невероятно легко отсылать HTTP/1.1-запросы.
* Selenium. При помощи этого модуля разработчики могут программными методами открывать веб-страницы, заполнять поля, кликать по кнопкам и отсылать формы.
* pyquery позволяет делать jQuery-запросы в XML-документах. API этой библиотеки максимально приближен к jQuery. Для быстрой манипуляции с XML и HTML pyquery использует LXML.
  + 1. Манипуляции с изображениями и видео

Python это очень мощный язык; с его помощью можно решать многие задачи, например, работать с изображениями. Стандартная библиотека не имеет встроенного функционала для этих целей, но следующие модули вам помогут.

* Python Imaging Library (PIL). PIL добавляет в интерпретатор Python возможность обработки изображений. Эта библиотека поддерживает много форматов файлов и предоставляет хороший функционал для работы с графикой.
* Gdmodule. Это интерфейс для библиотеки GD, созданной Томасом Баутелом.
* VideoCapture. Win32-расширение для доступа к таким видеоустройствам, как веб-камеры (USB) и ТВ-карты.
* MoviePy. Библиотека Python для редактирования видео: обрезка, склейка, вставка надписей, нелинейное редактирование, обработка видео и создание пользовательских эффектов.
* pyscreenshot. Кроссплатформенный модуль, позволяющий делать скриншоты без установки сторонних библиотек.
* Data Science и математика.

Для Python есть много расширений, которые можно использовать в научных целях, например, в математике, науке о данных и инженерии.

* SciPy. SciPy (произносится как «сай пай») это экосистема программного обеспечения с открытым кодом для математики, науки и инженерии.
* Matplotlib. Библиотека Python для создания 2D-графиков. Генерирует пригодные для печати изображения в разных форматах. Matplotlib может использоваться в скриптах, интерпретаторе Python, в Jupyter Notebook, на серверах веб-приложений и в GUI-инструментарии.
* Pandas. Быстрый, мощный, гибкий и легкий в использовании инструмент для анализа данных и работы с данными. Имеет открытый исходный код.
* Numpy. Библиотека, обеспечивающая поддержку больших, многомерных массивов и матриц. Для работы с этими массивами Numpy имеет обширную коллекцию высокоуровневых математических функций.

1. ИНДИВИДУАЛЬНОЕ ЗАДАНИЕ
   1. Условие задачи

Азбука Морзе была изобретена в XIX веке для передачи информации посредством телеграфа. Она широко используется и сегодня, более чем через 160 лет после ее создания.

Написать программу, в которой соответствие символов из азбуки Морзе будет храниться в виде словаря. В табл. 6.3 приведена та часть азбуки, которая вам понадобится при решении этого задания. В основной программе вам необходимо запросить у пользователя строку. После этого программа должна преобразовать его в соответствующую последовательность точек и тире, вставляя пробелы между отдельными символами. Символы, не представленные в таблице, можно игнорировать.

Например, сообщение Hello, World! может быть представлено следующей последовательностью:



Рис. 2.1.1 – Словарь для шифровки и дешифровки

*Источник: Индивидуальное задание*

* 1. Решение

В качестве среды разработки был выбран JupyterHub, а именно средство ноутбук. Jupyter-ноутбук — это среда разработки, где сразу можно видеть результат выполнения кода и его отдельных фрагментов. Отличие от традиционной среды разработки в том, что код можно разбить на куски и выполнять их в произвольном порядке.

В настоящей работе рассматривается реализация программы, указанной в задании №5, вариант №4. Первым делом в коде был прописан словарь для последующей шифровки сообщений (Рис.2.2.1):

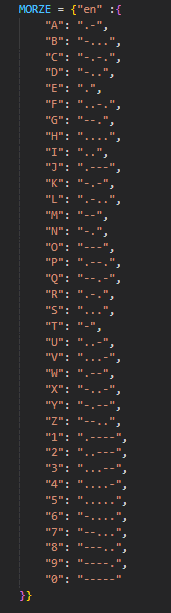


Рис. 2.2.1 – Словарь для шифровки и дешифровки

*Источник: Составлено автором*

Далее были описаны две функции - morze\_en\_encode и morze\_en\_decode – для шифровки и дешифровки (соответственно) текстовых сообщений. Первая функция (Рис. 2.2.2) является реализацией указанного выше задания.

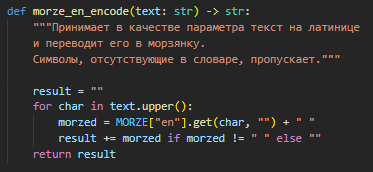


Рис. 2.2.2 – Код функции шифрования

*Источник: Составлено автором*

Реализация второй функции (Рис. 2.2.3) же необходима для проведения проверки работы первой функции.

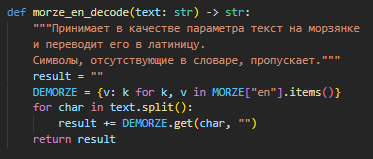


Рис. 2.2.3 – Код функции дешифрования

*Источник: Составлено автором*

* 1. Тестирование

Для тестирования составлен отдельный модуль (Рис. 2.3.1), состоящий из:

* функции тестирования (шифрование и последующее дешифрование текстовой строки)
* списка тестовых текстовых строк
* цикла, инициализирующего тестироване

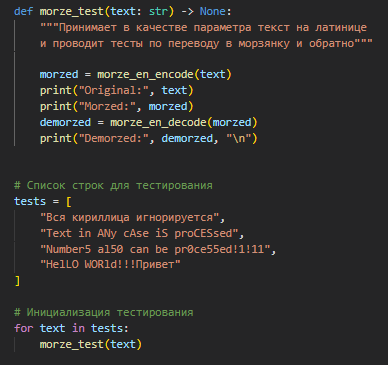


Рис. 2.3.1 – Код тестирования функции

*Источник: Составлено автором*

После запуска кода тестирования, был получен следующий результат (Рис 2.3.2):

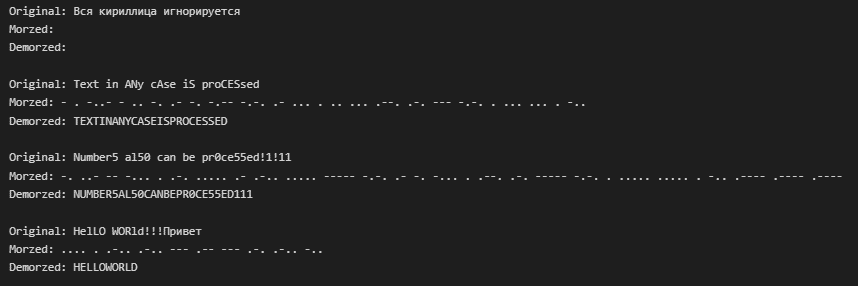


Рис. 2.3.2 – Результат тестирования

*Источник: Составлено автором*

# ЗАКЛЮЧЕНИЕ

В ходе настоящей работы была реализована программа на языке программирования Python. Были использованы различные методы работы: создание словарей, работа с условиями, ввод и вывод информации, работа с числами и функциями.

Язык программирования Python применяется во многих областях, а также подходит для реализации небольших, но полезных программ. Простой синтаксис делает работу удобной, не позволяет разработчику допускать много ошибок. Наличие большого количества встроенных функций и методов позволяют не тратить время на их написания с нуля.

Настоящая работа была выполнена при обращении к различным источникам: Интернет-источникам, научным статьям, а также учебникам.

# СПИСОК ЛИТЕРАТУРЫ

Нетологии, Р. (2016). *Что такое Python: чем он хорош, где пригодится и как его выучить*. Получено 2022, из Медия Нетологии: https://netology.ru/blog/python

*Основные модули Python*. (2020). Получено 2022, из Pythonist: https://pythonist.ru/osnovnye-moduli-python/

*Язык программирования Python*. (2021). Получено 2022, из Сложные IT-проекты: https://web-creator.ru/articles/python