**Tallinna Tööstushariduskeskus**

**Lõpptöö  
 Teema: Kliendibaas Python**

**Õpetaja: Marina Oleinik  
Õpilane: Jekaterina Tsimiris**

**Tallinn 2020**

1. ВВЕДЕНИЕ

В настоящее время на рынке можно найти огромное количество разных программ, но маленьких простых программ, удобных и понятных пользователям – найти сложно. Поэтому автор данной работы создал небольшую программу – «Маленькая база данных клиентов», которая будет проста в использование для обычного пользователя. В программе можно хранить данные клиентов и счетов, добавлять / удалять клиентов, выводить на экран все детали по конкретному клиенту с возможностью озвучки, визуализируется отчет по счетам продаж и закупок, а также возможность добавлять счета по продажам и закупкам. Использовался язык написания программы – Питон. С блок - схемой работы приложения можно ознакомиться в приложении 1.

1. МОДУЛИ ПРИЛОЖЕНИЯ

В приложение были импортированы следующие модули:

1. ***from gtts import \****

*GTTS* (*Google Text to Speech*) - cоздает *mp3* файл с помощью модуля `gTTS`; активирует разговорную речь, т.е проговаривает текст.

1. ***import sys***

Модуль *sys* включен в стандартные библиотеки и содержит функции и другие данные, необходимые для нашего кода, чтобы выполнить самоанализ системы, в которой он работает.

1. ***import os***

Модуль *os* - это библиотека функций для работы с операционной системой.

1. ***import string***

*String* модуль предоставляет константы для операций, связанных с строкой.

1. ***import re***

Регулярные выражения – это последовательность символов, используемая для поиска и замены текста в строке или файле.

1. ***from tkinter import\****

***from tkinter import*** ***messagebox***

***from tkinter import Message***

***import tkinter as tk***

Модуль *Tkinter* является стандартным интерфейсом *Python* для графического пользовательского интерфейса *GUI (graphical user interface – GUI)* на основе широко распространенного на *X-Windows Tk*-интерфейса.

1. ***import datetime***

Модуль *datetime* для обработки времени и даты разными способами.

1. ***from collections import namedtuple***

«*namedtuples* ()» присутствует в модуле «*collection*», как и словари, они содержат ключи, которые хэшируются с определенным значением, поддерживает как доступ по значению ключа, так и итерацию, что не хватает словарям.

1. ***import csv***

Модуль *csv* реализует классы для чтения и записи табличных данных в формате *CSV*.

1. ***import*** ***pandas as pd***

***from pandas import ExcelWriter***

***from pandas import ExcelFile***

***from pandas import DataFrame***

*Pandas* модуль предназначен для анализа данных.

*Pandas ExcelWriter -* класс для записи объектов DataFrame в таблицы Excel

*Pandas ExcelFile – эквивалентно* ***«***read\_excel (ExcelFile,…)», чтение файла.

*Pandas DataFrame -*определяются как двумерные маркированные структуры данных со столбцами потенциально разных типов. DataFrame состоит из трех основных компонентов: данных, строк и столбцов.

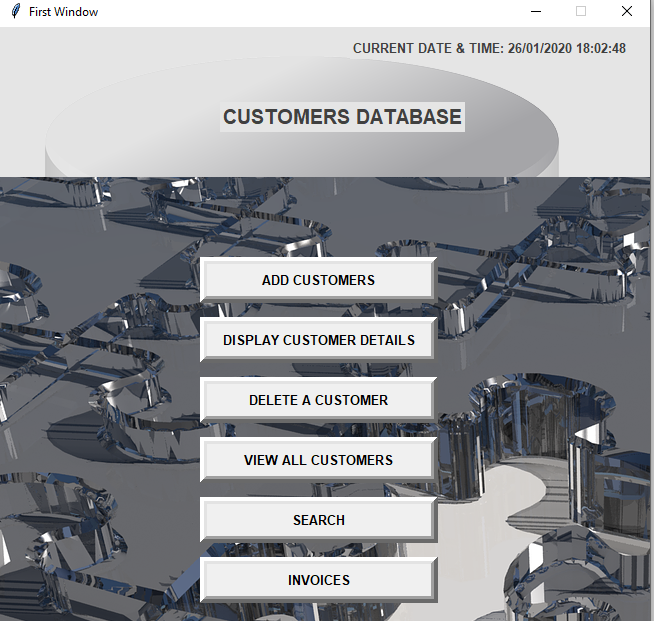
1. ***import*** ***matplotlib***

Модуль для построения графиков.

1. ***import numpy as np***

NumPy (Numeric Python) предоставляет базовые методы для манипуляции с большими массивами и матрицами.

1. FIRST WINDOW / ГЛАВНОЕ ОКНО

При запуске приложения появляется главное окно (*First Window*) с наименованием проекта «CUSTOMERS DATABASE».

Joonis . Main Window

Данное окно разбито на фреймы: верхний и нижний. В верхнем фрейме: текущая дата со временем, наименование проекта.

Для вывода в окне времени использовался модуль «*datetime»* и функция для вывода нужного формата:

*date=datetime.datetime.now()*

*date=str (date.strftime (" %d/%m/%Y %H:%M:%S ") )*

где *%d* – день месяца,

*%m* – месяц,

*%Y* – год,

*%H* – час,

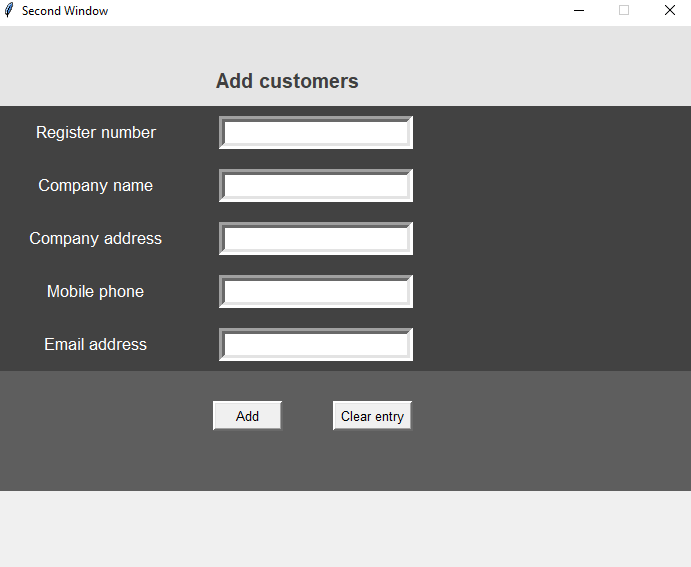
*%M* – минуты,

*%S* – секунды.

В нижнем фрейме расположены 6 кнопок, каждая из которых открывает дополнительно графическое окно с определенными функциями.

В двух фреймах использовались картинки в *png* формате для заднего фона.

1. *ADD CUSTOMERS /* ДОБАВЛЕНИЕ КЛИЕНТА

При нажатии в главном меню кнопки «*ADD CUSTOMERS*» открывается второе графическое окно с полями для заполнения данных. В данном окне использовалось 3 фрейма: верхний (расположен заголовок), средний (расположены поля для заполнения), нижний (расположены 2 кнопки).

После заполнения всех полей пользователем и нажатия кнопки «*ADD*», функция выполняет проверку – есть ли в *csv* файле такой же регистрационный номер компании или нет. Если регистрационный номер уже есть в *csv* файле, то выдает сообщение, что такой уже есть в системе! В противном случае все введенные данные пользователем сохраняются в *csv* файл и показывается сообщение, что новый клиент добавлен успешно.

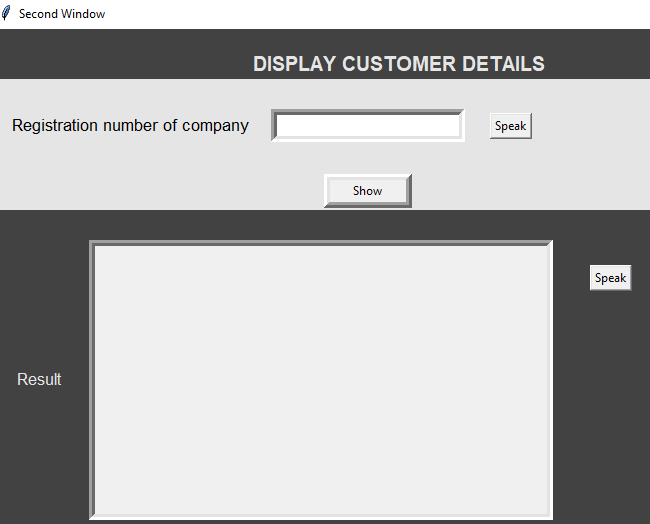
Joonis . Add customers

В функции использовались блоки: *try – except – else,* условные операторы *if – else, messagebox –* для вывода сообщения на экран.

*If – else –* логическое условие, она выбирает, какое действие следует выполнить, в зависимости от значения переменных в момент проверки условия.Блок *try* позволяет проверить блок кода на наличие ошибок.Блок *except* позволяет обработать ошибку. Блок *else* выполняется, если не было ошибок и не попал в *try.*

Кнопка «*CLEAR ENTRY*» очищает поля, если пользователь решил не сохранять данные.

Для выхода из данного окна пользователь нажимает крестик в верхнем правом углу.

1. DISPLAY CUSTOMER DETAILS / ПОКАЗАТЬ ДЕТАЛИ КЛИЕНТА

Joonis . Display customer details

При нажатии в главном меню кнопки «*DISPLAY CUSTOMER DETAILS*» открывается второе графическое окно с полем для ввода данных. В данном окне использовалось 3 фрейма: верхний (расположен заголовок), средний (расположен поле для ввода данных, кнопка «*Speak*», кнопка «*Show*»), нижний (расположены кнопка «*Speak*» и окно с результатом).

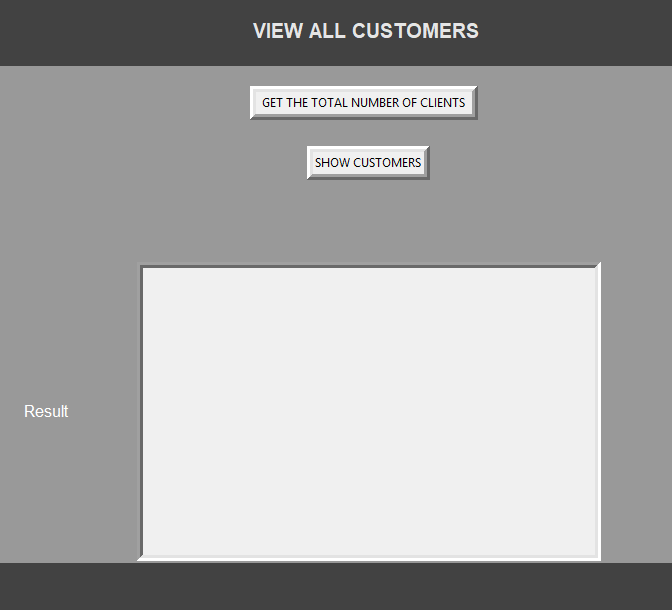
После введенного регистрационного номера пользователем, выполняется проверка есть ли такой в *csv* файле. Если есть клиент в системе, то выводится результат в окне со всеми данными, в противном случае выдается сообщение, что такого нет и приложение закрывается. Также есть возможность озвучки, то есть прослушать пользователю, как введенные им данные, так и полученный результат.

В функции использовались блоки: *try – except – else*, условные операторы *if – else, messagebox –* для вывода сообщения на экран. С помощью модуля *gtts* проговаривается текст.

1. DELETE A CUSTOMER / УДАЛИТЬ КЛИЕНТА

При нажатии в главном меню кнопки «*DELETE A CUSTOMER*» открывается второе графическое окно с полем для ввода данных. В данном окне использовалось 3 фрейма: верхний (расположен заголовок), средний (расположен поле для ввода данных, кнопка «*Delete*»), нижний (расположен окно с результатом). Пользователь вводит регистрационный номер, после чего удаляется из файла и в окне появляется результат- печатается, что клиент удален. Для выхода из данного окна пользователь нажимает крестик в верхнем правом углу.

1. *VIEW ALL CUSTOMERS /* СМОТРЕТЬ ВСЕХ КЛИЕНТОВ

При нажатии в главном меню кнопки «*VIEW ALL CUSTOMERS*» открывается второе графическое окно с кнопками и окном для вывода результата.

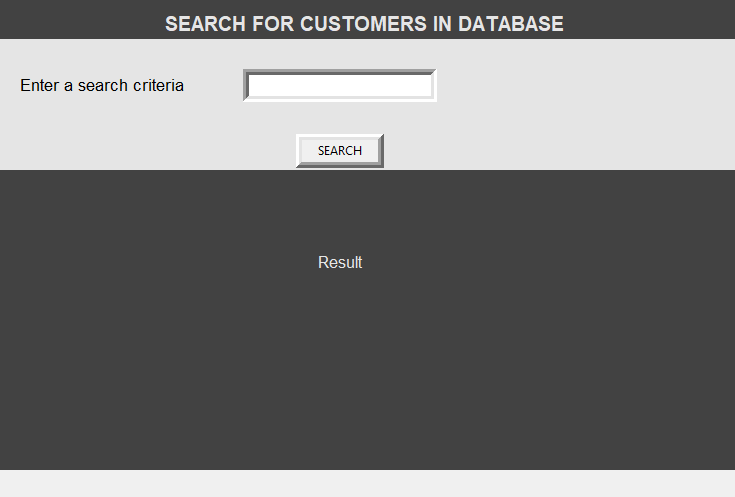
В данном окне использовалось 3 фрейма: верхний (расположен заголовок), средний (расположены кнопка «*Get the total number of clients*», кнопка «*Show customers*»), нижний – темная полоса.

После нажатия кнопки «*Get the total number of clients*», в окне для результата появляется информация о количестве клиентов в базе. После нажатия кнопки «*Show customers*» в окне для результата выводится вся информация клиентов.

Joonis . View all customers

Для чтения и вывода данных использовались модули *Pandas,Pandas DataFrame, numpy.*

Для выхода из данного окна пользователь нажимает крестик в верхнем правом углу.

1. SEARCH / ПОИСК

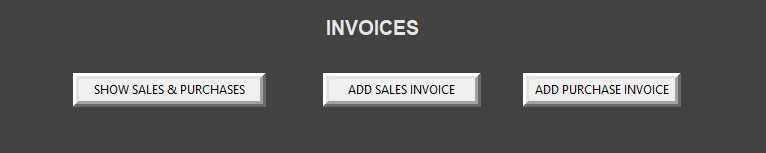
Joonis . Search

При нажатии в главном меню кнопки «*SEARCH*» открывается второе графическое окно с полем для ввода данных. В данном окне использовалось 3 фрейма: верхний (расположен заголовок), средний (расположены поле для ввода данных и кнопка «*Search*»), нижний (выводится результат).

После введенных данных пользователем, выполняется построчная проверка в *csv* файле. Если данные есть в базе, то выводится результат в окне, в противном случае выдается сообщение, что такого нет и приложение закрывается.

В функции использовались блоки: *try – except – else*, условные операторы *if – else, messagebox –* для вывода сообщения на экран.

1. INVOICES / СЧЕТА

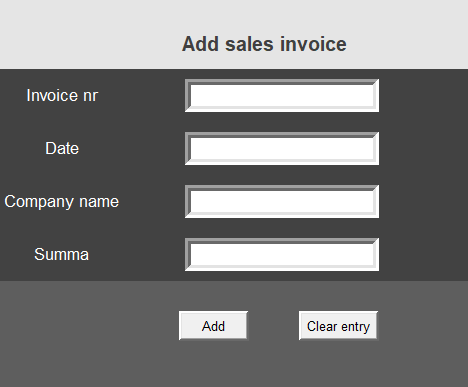
При нажатии в главном меню кнопки «*INVOICES*» открывается второе графическое окно с 3 кнопками.

Joonis . Invoices

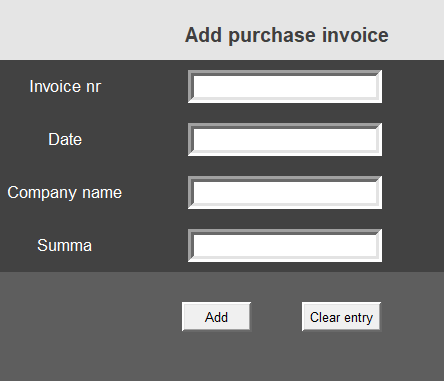
В данном окне использовалось 2 фрейма: верхний (расположен заголовок и кнопки: «*SHOW SALES & PURCHASES*», «*ADD SALES INVOICE*», «*ADD PURCHASE INVOICE*»), нижний – заполнен цветом.

При нажатии кнопки «*SHOW SALES & PURCHASES*» открывается график на основании таблицы данных в xlsx файле, где хранятся данные счетов продаж и закупок.

Для чтения и вывода данных использовались модули *Pandas,Pandas DataFrame, Matplotlib, numpy.*

При нажатии кнопки «*ADD SALES INVOICE*» открывается графическое окно с полями для заполнения данных. В данном окне использовалось 3 фрейма: верхний (расположен заголовок), средний (расположены поля для заполнения), нижний (расположены 2 кнопки). После заполнения всех полей пользователем и нажатия кнопки «*ADD*», функция выполняет проверку – есть ли в *xlsx* файле такой же номер счета или нет. Если есть, то выдается сообщение, что есть уже такой, в противном случае данные записываются в файл. Кнопка «*CLEAR ENTRY*» очищает поля, если пользователь решил не сохранять данные. В функции использовались блоки: *try – except – else*, условные операторы *if – else, messagebox –* для вывода сообщения на экран. Для выхода из данного окна пользователь нажимает крестик в верхнем правом углу.

Joonis . Add sales invoice

При нажатии кнопки «*ADD PURCHASE INVOICE*» открывается графическое окно с полями для заполнения данных. В данном окне использовалось 3 фрейма: верхний (расположен заголовок), средний (расположены поля для заполнения), нижний (расположены 2 кнопки). После заполнения всех полей пользователем и нажатия кнопки «*ADD*», функция выполняет проверку – есть ли в *xlsx* файле такой же номер счета или нет. Если есть, то выдается сообщение, что есть уже такой, в противном случае данные записываются в файл. Кнопка «*CLEAR ENTRY*» очищает поля, если пользователь решил не сохранять данные. В функции использовались блоки: *try – except – else*, условные операторы *if – else, messagebox –* для вывода сообщения на экран. Для выхода из данного окна пользователь нажимает крестик в верхнем правом углу.

Joonis . Add purchase invoice

1. ЗАКЛЮЧЕНИЕ

Данное приложение имеет потенциал для развития до стадии конечного продукта, для предложения пользователям вспомогательного управленческого инструмента.

1. ИСПОЛЬЗОВАННЫЕ ИСТОЧНИКИ ЛИТЕРАТУРЫ

Hoffman, C. 1994. A Python Quick Reference - [WWW] URL https://www.python.org/ftp/python/doc/1.1/quick-ref.1.1.html (02.10.2019)

Pandas - [WWW] URL <https://thispointer.com/pandas-check-if-a-value-exists-in-a-dataframe-using-in-not-in-operator-isin/> (11.01.2020)

Python – [WWW] URL https://ru.wikipedia.org/wiki/Python (02.10.2019)

Python Tutorial - [WWW] URL <https://www.tutorialspoint.com/python/index.htm> (12.01.2020)

Python Pandas – DataFrame - [WWW] URL <https://www.tutorialspoint.com/python_pandas/python_pandas_dataframe.htm> (12.01.2020)

Python Pandas – DataFrame - [WWW] URL <https://www.geeksforgeeks.org/python-pandas-dataframe/> (12.01.2020)

Rossum, G. 2001. Python Reference Manual - [WWW] URL https://docs.python.org/2.0/ref/ref.html (01.10.2019)

Styles with Matplotlib - [WWW] URL <https://pythonprogramming.net/styles-matplotlib-tutorial/> (10.01.2019)

1. СПИСОК ИЛЛЮСТРАЦИЙ

[Joonis 1. Main Window 4](file:///C:\Users\37255\Desktop\algoritmi,%20python\kliendi_baas\python_project.docx#_Toc31035426)

[Joonis 2. Add customers 5](file:///C:\Users\37255\Desktop\algoritmi,%20python\kliendi_baas\python_project.docx#_Toc31035427)

[Joonis 3. Display customer details 6](file:///C:\Users\37255\Desktop\algoritmi,%20python\kliendi_baas\python_project.docx#_Toc31035428)

[Joonis 4. View all customers 7](file:///C:\Users\37255\Desktop\algoritmi,%20python\kliendi_baas\python_project.docx#_Toc31035429)

[Joonis 5. Search 8](file:///C:\Users\37255\Desktop\algoritmi,%20python\kliendi_baas\python_project.docx#_Toc31035430)

[Joonis 6. Invoices 9](file:///C:\Users\37255\Desktop\algoritmi,%20python\kliendi_baas\python_project.docx#_Toc31035431)

[Joonis 7. Add sales invoice 9](file:///C:\Users\37255\Desktop\algoritmi,%20python\kliendi_baas\python_project.docx#_Toc31035432)

[Joonis 8. Add purchase invoice 10](file:///C:\Users\37255\Desktop\algoritmi,%20python\kliendi_baas\python_project.docx#_Toc31035433)

1. ПРИЛОЖЕНИЕ 1

Блок – схема работы приложения прикреплена в png формате.

СОДЕРЖАНИЕ

[I. ВВЕДЕНИЕ 2](#_Toc31043959)

[II. МОДУЛИ ПРИЛОЖЕНИЯ 3](#_Toc31043960)

[III. FIRST WINDOW / ГЛАВНОЕ ОКНО 5](#_Toc31043961)

[IV. *ADD CUSTOMERS /* ДОБАВЛЕНИЕ КЛИЕНТА 6](#_Toc31043962)

[V. DISPLAY CUSTOMER DETAILS / ПОКАЗАТЬ ДЕТАЛИ КЛИЕНТА 7](#_Toc31043963)

[VI. DELETE A CUSTOMER / УДАЛИТЬ КЛИЕНТА 8](#_Toc31043964)

[VII. *VIEW ALL CUSTOMERS /* СМОТРЕТЬ ВСЕХ КЛИЕНТОВ 8](#_Toc31043965)

[VIII. SEARCH / ПОИСК 9](#_Toc31043966)

[IX. INVOICES / СЧЕТА 10](#_Toc31043967)

[X. ЗАКЛЮЧЕНИЕ 12](#_Toc31043968)

[XI. ИСПОЛЬЗОВАННЫЕ ИСТОЧНИКИ ЛИТЕРАТУРЫ 13](#_Toc31043969)

[XII. СПИСОК ИЛЛЮСТРАЦИЙ 14](#_Toc31043970)

[XIII. ПРИЛОЖЕНИЕ 1 15](#_Toc31043971)